Broadband Traffic Modeling: Simple Solutions to Hard Problems

Ronald G. Addie, University of Southern Queensland

Moshe Zukerman and Timothy D. Neame, University of Melbourne

ABSTRACT A much clearer picture of the progress toward an integrated high-speed multiservice network is now emerging. Such networks were anticipated over 20 years ago, at a time when packet switching was just another way to transmit data. Now the technology is so mature that media barons are organizing their investments in order to take advantage of the profit. Many of the technical problems are now solved, and the fundamental protocols required for these networks are sufficiently well-defined to support a rapidly expanding industry. However, consensus on how to talk about the statistics of the data transmitted around these networks has not been readily forthcoming. Nevertheless, there now exists a family of models with sufficient richness to describe real traffic fairly well, which can be parameterized conveniently, and which degenerates to a readily analyzable Gaussian model in the situation of very large networks. This analysis leads to important architectural conclusions which accord with common sense, in particular the conclusion that integrated networks of the future should be able to provide better and better service with efficiency tending toward perfect. This is a rather happy conclusion which thoroughly confirms the current rapid drive toward an all-encompassing integrated multiservice network. Rather than the increase in traffic and diversity of service types leading to greater and greater complexity, it seems that the flow of traffic in our networks may become steadily more manageable.

It is predicted that the exponential growth of store-and-forward (packet switching) networks such as the Internet and/or asynchronous transfer mode (ATM) will continue for some time, and that their traffic volume will soon exceed that of traditional telephone networks. As soon as this happens, even before it happens, there will be very strong pressure for all communication services to make use of the cheapest available communication service; in other words, there is pressure for all services to use one integrated multiservice network. Two significant reasons for this pressure are:

- One network is cheaper than two (or more) for the users.
- Services often need access to other services, which is impossible if they are on a different network.

Indeed, we can see this already beginning to happen, and the communication, computer, and entertainment industries have been reorganizing themselves in anticipation of this development, which is sometimes referred to as *convergence*.

In such a context of rapid ongoing change it may seem impossible to make much in the way of a definite statement about the future. However, perhaps the old saying, “Change is the only constant,” will come to our aid. In particular, as networks get larger, the impact of one traffic event, or one class of traffic, or one group of users, in the whole spectrum of factors which make up the network, will become less and less significant, and as a consequence, the overall behavior of the network may become somewhat more predictable.

In packet switching networks, traffic is segmented into blocks of data. These blocks may be Internet packets or ATM cells. For the purpose of this article the specifics of traffic segmentation are not important; we shall use the word *cell* to refer to the basic unit of traffic load, and for simplicity we shall assume that all cells are of the same size (as in ATM). Such cells are switched between switches according to information contained in their headers. When cells arrive at a switch, or when they are ready to leave a switch, if an excess number of cells all need to use the same link at the same time, cells will have to be stored in a buffer awaiting transmission.

This causes random delays for the cells which may degrade performance of services based on these cells. Some services may be badly affected if the delay caused in this way varies excessively. If the rate of cells heading down a certain link remains greater than the link capacity for a sufficient period of time, the buffer will overflow and loss occurs. For data services, such loss leads to retransmissions which delay messages even more, or may even cause network congestion collapse. In video services, cell loss reduces picture quality. Virtually all aspects of performance over which we have any control when designing and managing networks are related to the levels of stored data in the buffers which sit between the switches and transmission links of the network.

Store-and-forward networks can be viewed as a network of queues, and the most fundamental component in such networks is the single-server queue. Accordingly, performance analysis of a single-server queue has been a focal point of communications research for several years. The important questions often raised by network planners and designers are: how much bandwidth is required to serve a bursty traffic stream such that certain quality of service (QoS) requirements are met? Or equivalently, what is the proper average utilization of a given link serving a bursty traffic stream? How is this appropriate utilization level affected when the traffic is composed of many traffic streams? In other words, what is the *multiplexing gain*? Is there an accurate and useful traffic model in the form of a simple stochastic process which can be described by a small number of parameters and, when fed into a single server queue, gives the same queuing performance as a real traffic stream? Such a traffic model will be very useful in network design tools or in tools supporting real-time traffic management. Despite very extensive research during the last 20 years [1], there is no consensus on such a model. The wish of many practitioners for an “Erlang formula” for broadband traffic has not yet been granted. Nevertheless, significant progress has been made over the years in understanding the characteristics of such traffic. We intend here to provide the reader with insights into these characteristics, and to make a contribution toward a consensus on certain key aspects of broadband traffic modeling.

Although the significance of the effect of traffic correlation on queuing performance was recognized, during the ’80s and early ’90s, the focus of traffic modeling research was mainly on processes exhibiting only short-range dependence (SRD). (See [1] for a comprehensive review on traffic models.)

0163-6804/98/$10.00 © 1998 IEEE

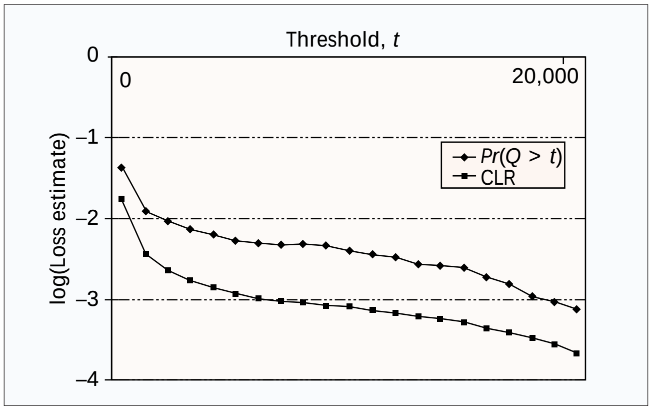

Figure 1. *Overflow probability versus cell loss ratio.*

The discovery of long-range dependence (LRD) in Ethernet traffic in Bellcore [2] is considered to be one the most significant in the area of traffic modeling. Further studies confirm that metropolitan area network (MAN) traffic [3], wide area network (WAN) traffic, and variable bit rate (VBR) video traffic [1] also exhibit LRD (or self-similarity or fractal) characteristics. This discovery provided insight into traffic behavior which has brought researchers closer to achieving the goal of obtaining useful traffic models supporting efficient network dimensioning procedures and traffic management functions.

The rapid and ongoing aggregation of more and more traffic onto the Internet, or whatever it becomes, is a factor which cannot be ignored in traffic modeling. Fortunately, we can apply a well and truly established mathematical theory to this process of aggregation to discover how this growth might affect traffic and the performance it experiences: the *central limit theorem*. This theorem applies not just to the sample distribution of the amount of traffic arriving in a certain interval of time, but actually to the entire joint distribution of traffic in all streams arriving in a whole range of different time intervals [4]. As a consequence, we can expect traffic to become more and more like Gaussian processes, and we can expect networks to behave more closely, as time passes, to the way they might behave if the traffic were Gaussian.

At the moment, network traffic does not appear to be all that close to Gaussian. In many networks the degree of aggregation is not sufficient to inhibit the bad behavior of one traffic stream from impinging negatively on another, and the only way to guarantee good performance (low delay) is to operate at relatively low occupancies. However, once the degree of aggregation is sufficient that the approximation by Gaussian traffic is good, a rather different regime seems to emerge. Whenever independent quantities are added, their mean grows linearly in the number added, while their standard deviation grows in proportion to the square root of the number added; for this reason, we often see a sort of *smoothing* taking place. It appears that the same should be true of network traffic, but only from that point in time when a Gaussian approximation is valid.

If networks were growing slowly, considering how unpleasant some network traffic is, it would be unreasonable to hold out hope for the central limit theorem to come to our rescue in this manner. However, traffic growth rates are very large and not slackening off, and the central links of today's networks carry traffic from millions of independent sources.

At the edge of this desert of bursty traffic which we have been traversing, while the communication infrastructure of the third millenium is put in place, there sits, just on the horizon, a land of milk and honey — the realm of integrated multiservice networks, in which all services receive good service, despite the high utilization levels on all the links, and there is no need to set up elaborate schemes to protect one service from another. And the reason things are so good in this realm is that the traffic there is Gaussian! However, before traffic becomes Gaussian, a simple model like the M/Pareto, if properly used, can potentially provide accurate prediction of queuing performance and dimensioning for bursty LRD traffic.

BASIC CONCEPTS

In this section we shall explain the basic concepts in the art of traffic modeling. In particular, we consider a single-server queue fed by a traffic stream and explain the effect of key statistical characteristics of the traffic process on queuing performance.

QUEUING PERFORMANCE MEASURES

The most important queuing performance measures for networks are delay, jitter (standard error, or variance, of delay), loss, and overflow. Since these are statistical quantities, we need to talk about their mean and variance, or their probability distribution. Jitter can be important, because some services can cope perfectly well with a consistent delay, but not very well with fluctuating delays.

Although these measures may not always represent subjective QoS perceived by users, they are relatively simple to measure, monitor, and accommodate, and hence they are used in practice. Delay is the time from the moment a cell arrives until its service is complete. Loss ratio is the ratio between the total number of cells lost and the total number of cells arriving. The underlying assumption in the definition of loss ratio is that the buffer is finite, and cells arriving when the buffer is full are lost. On the other hand, to define overflow probability we usually consider an infinite buffer queue, and define overflow probability as the probability (or the proportion of time) that the number of cells in the buffer exceeds a certain threshold. In queuing theory, overflow probability is closely related to the concepts of virtual waiting time or unfinished work distribution.

Out of the four, the overflow probability is the most amenable to mathematical analysis, and in many publications it is used as an approximation for loss ratio. Normally, this is not a bad approximation; however, one should keep in mind that we can easily create a situation whereby the overflow probability is very small (almost zero) and the loss ratio is close to one. Consider a case where the input process is characterized by a very bursty process which starts with a large burst that arrives at once, and then for a very long time (much longer than the time it takes to process the burst) there are no arrivals at all; then another large burst arrives and the process repeats itself. Clearly, most of the time there is no traffic, and hence the overflow probability is very small. On the other hand, in the case of finite buffer, most of the traffic is lost because each burst is much bigger than the amount of work that can be buffered; therefore, the loss ratio is almost one. Of course, this example is not a typical case. In Fig. 1 we compare loss ratio and overflow probability in a queue fed by an Ethernet trace, and demonstrate that the two are closely related in a more realistic situation. Overflow probability will be used henceforth to characterize performance because it is readily amenable to analysis and hence provides us with the insight we need.

TRAFFIC CHARACTERISTICS

By a link we mean any shared transmission resource, not necessarily a physical link. It can be a virtual path or a logical link dedicated to a certain service where many connections of the same type are multiplexed together. Link dimensioning is not only done at the network design stage. Dimensioning of logical links and virtual paths can take place during operation

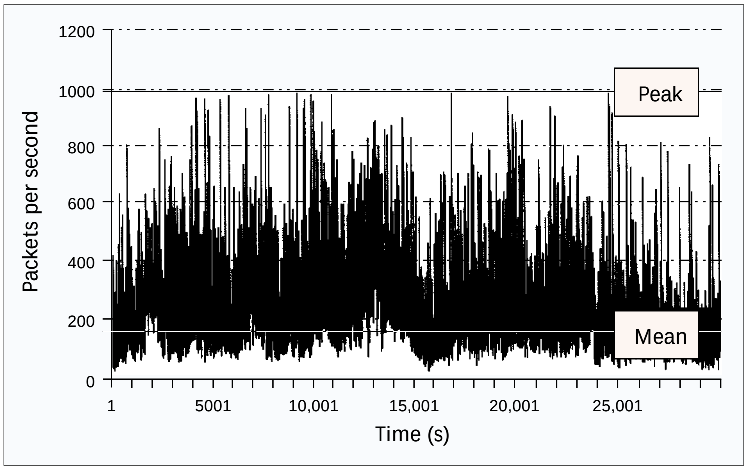

Figure 2. *Mean and peak for an Ethernet trace.*

when bandwidth is switched between different logical links to optimize network utilization and to guarantee QoS to users. Such bandwidth switching can rely on real-time traffic measurements and optimal use of these measurements for efficient network operation. An understanding of important traffic characteristics is essential for optimal use of traffic measurements. We begin our discussion with the two simple characteristics, peak and mean, and then discuss characteristics related to burstiness.

Peak and Mean — The mean is simply the average number of cells (amount of work) generated in a unit of time. The peak represents the highest rate generated, and in the context of ATM it is defined as the reciprocal of the minimal intercell time.

Clearly, if the service rate is equal to or higher than the peak rate, no cells will ever be buffered, and the cell loss ratio (CLR) will be equal to zero. On the other hand, if the service rate is lower than the mean arrival rate, the queue will lead to unacceptable levels of delay, CLR, and overflow probability. Setting the service rate equal to the peak is called *peak rate allocation* (PRA). For typical bursty streams, PRA is too wasteful because very rarely does the source transmit at its peak, and normally an optimal service rate will be somewhere between the peak and the mean. The concept of optimal service rate, known also as *effective bandwidth*, is the minimal service rate which can serve a bursty stream such that QoS requirements are met. If the difference between peak and mean is very large and the traffic is very rarely at its peak, PRA may be wasteful; nevertheless, PRA is an important option and should be used for services requiring very high QoS, or for cases where the peak cell rate (PCR) is used for long periods; a special case of the latter is where the peak is equal to the mean. This is the case of constant bit rate (CBR) traffic, in which case PRA must be used.

Normally, a link will be used by many bursty sources, and rarely, they all transmit at their peak at the same time; and since most bursty services do not require very stringent QoS, optimal decisions on effective bandwidth will have to be made to operate the network economically. In many cases, the peak is many times the mean (Fig. 2) and the range of possible effective bandwidth levels very wide. The decision of how to set the effective bandwidth is very much affected by the burstiness of the traffic.

Burstiness and Correlation — It has been known for many years that if sources are more *bursty*, queuing performance is degraded, and to meet QoS we must increase the service rate. However, what exactly the term *burstiness* means in terms of measurable traffic statistics has been a critical problem. In other words, the problem of identifying the critical traffic statistics that cause congestion, and their effect on congestion, has been an open problem for many years.

Clearly, the variance of the amount of work arriving in an interval, that of the marginal distribution, or the *short-term variance* affects queuing performance. Assuming a single-server queue (SSQ) with a fixed service rate, which is greater than the mean arrival rate to ensure stability, if the mean arrival rate stays fixed and the variance increases, the frequency of congestion will increase and with it average queue size levels, overflow probabilities, CLR, and delay. In fact, for every arbitrarily small mean, it is easy to demonstrate that large variance can lead to arbitrarily high CLR. Consider the case where a very large batch arrives (the batch is much larger than the service rate and the buffer size) and almost all of it is lost. Following the arrival of the very large batch there are no arrivals for a long period of time — long enough to keep the overall mean arrival rate below any given level.

The mean, peak, and short-term variance are not the only critical statistical characteristics. It has been known for many years that the traffic autocorrelation function has significant effect on queuing performance. It is easy to demonstrate a situation wherein a traffic stream with a given mean, peak, and variance is fed into an SSQ with a finite buffer and no loss occurs, but as the correlation increases (without changing the mean, peak, and short-term variance, or even the entire marginal distribution) loss increases significantly. A high level of correlation means long periods of heavy traffic (typically with the arrival rate higher than the service rate) during which the queue builds up and significant losses occur. These long periods of heavy traffic will typically be followed by long periods of light traffic, but the damage (the loss) is already done. On the other hand, low correlation means that short periods of high traffic are quickly followed by low-traffic periods, alleviating the danger of buffer overflow.

In Figs. 3 and 4 we demonstrate the adverse effect of correlation on performance and link utilization. The LRD curve in Fig. 3 represents the overflow probability as a function of the buffer size of an SSQ fed by measured Ethernet traffic. The "shuffled" curve represents the overflow probability versus buffer size for the same traffic, but in random order (obtained by shuffling the original measured traffic, thus retaining the marginal distribution). The SRD curve represents traffic where groups of 20 s traffic streams each are shuffled while the traffic within each group is unshuffled. It is

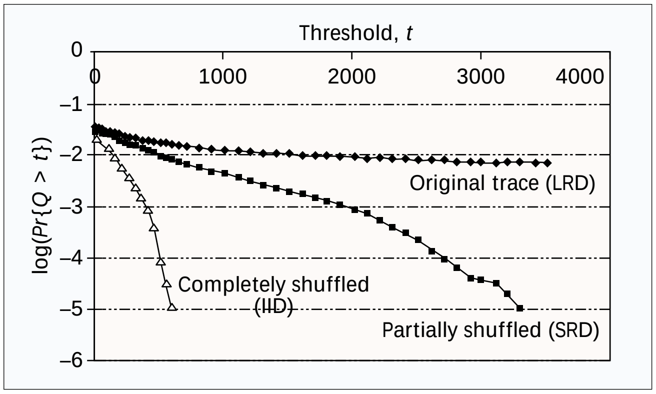

Figure 3. *The effect of correlation on overflow probability.*

clear that correlation has an adverse effect on performance, even if the marginal distribution remains intact.

In Fig. 4 we demonstrate the effect of correlation on link efficiency by determining the minimal bandwidth required to serve a traffic stream such that the CLR will be kept below a certain value. The results are presented for a wide range of buffer sizes. It is clear from the figure that the LRD traffic requires much more bandwidth than the shuffled traffic.

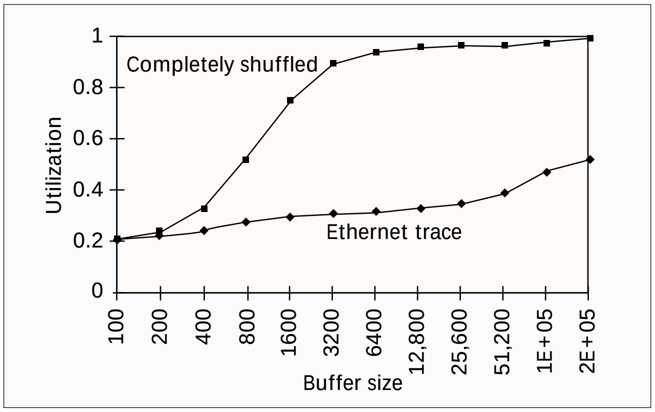

Figure 4. *The effect of correlation on utilization.*

Discrete Time Modeling

In this article we motivate the use of the discrete time fractal Gaussian model as an accurate and flexible model for aggregate traffic. For the purpose of comparability we shall discretize also the M/Pareto model. We shall now provide some basic definitions related to a generic discrete time traffic (or queuing) model. Defining the discrete time queuing model mathematically gives us an opportunity to mathematically define the traffic characteristics discussed earlier.

Consider a FIFO single server queue. Let time be divided into fixed-length sampling intervals. The model allows arbitrary choice of interval length. Let A_n be a continuous random variable representing the amount of work entering the system during the nth sampling interval. The process $\{A_n\}$ is assumed to be stationary and ergodic. Let τ be a fixed number representing the amount of work which can be processed by the server per sampling interval. We assume here for simplicity that the service takes place at the end of the interval. The mean of the amount of work arriving in an interval n is denoted $E(A_n)$. The variance of A_n is denoted by σ^2.

Let the sequence of continuous random variables Y_n be the net input process defined as

$$Y_n = A_n - \tau, \quad n \geq 0.$$

Let V_n be the unfinished work at the beginning of the nth sampling interval. Using the above notation, the system unfinished work process, for the case of infinite buffer, satisfies Lindley's recurrence equation:

$$V_{n+1} = (V_n + Y_n)^+, \qquad n \geq 0, \qquad (1)$$

where $V_0 = 0$ and where $X^+ = \max(0, X)$. The choice of sampling interval length should be made in a way that justifies the other assumptions of the model. For further discussion of the choice of sampling interval, see [5].

Let m be the mean of the net input, that is, $m = E(A_n) - \tau$. A necessary and sufficient condition for queuing stability is $m < 0$. Since τ is constant, $\sigma^2 = \mathrm{Var}(A_n) = \mathrm{Var}(Y_n)$.

We can now talk about the long-term variance, the variance of the amount of work that arrives over a long period of time (many time intervals). In other words, we consider the variance of the amount of work arriving in n consecutive intervals: $V(n) = \mathrm{Var}\ (\Sigma_{k=1}^{n} A_k)$.

Of particular interest is the limit: $v = \lim_{n\to\infty} = V(n)/n$. We call this limit the asymptotic variance rate (AVR) [3]. The AVR is also equal to the autocovariance sum [5]. That is, $v = \Sigma_{k=-\infty}^{\infty} \mathrm{Cov}(A_n, A_{n+k})$.

If the AVR is finite the process is SRD; otherwise, it is LRD. In other words, when this limit is infinite, the autocovariance (as well as the autocorrelation function) has a heavy tail which characterizes LRD traffic.

We have been measuring work in continuously varying units (real numbers) and time in discrete units (integers). Since work arrives in cells, and time can be subdivided arbitrarily, a strong case can be put that we should measure work in discrete units and time continuously.

The use of a continuous variable for work is justifiable, however, because the quantity of data in one ATM cell is very small and, as networks grow in all the ways previously discussed, this fundamental quantum of traffic will become less and less significant. Use of a continuous quantity for work also allows us to use a continuous probability distribution — such as the Gaussian distribution — to describe it. This becomes quite important when we consider very large aggregates of traffic in which, because of the effect of the central limit theorem, the best model is Gaussian.

The basic queuing equation which describes the buffering in the switches of a network is Eq. 1, which presupposes the use of discrete time. Switches do operate in this sort of fashion. On the other hand, as networks get larger, the number of basic cycles per second of a switch will be increasing steadily, so a continuous-time version of Eq. 1 might be a better model. A continuous time version can be written thus:

$$V_t = \max_{s\in(p,t)} \int_s^t Y_\tau d\tau, \qquad t \geq 0. \qquad (2)$$

If the sampling interval used in Eq. 1 is sufficiently small and the arriving work process, $\{A_n\}_{n\geq 0}$ is positive, Eqs. 1 and 2 should be virtually equivalent. However, some continuous-time models, in particular *fractional Brownian noise*, have the property that at small timescales more and more of the arriving work appears to be negative — virtually half of all arriving work at sufficiently small timescales. In this situation the continuous and discrete models appear to behave quite differently. This particular model is quite important because it is the simplest Gaussian model which captures the essential features of today's network traffic; however, the anomalous behavior at small timescales does not correspond to the real world, so we use discrete time models.

Fractal Traffic

Fractal (self-similar or LRD) traffic is characterized by burstiness on many timescales (hence the expression *self-similarity*). This is demonstrated in Fig. 5. We can see that as we increase the timescale the shuffled traffic is smoothed out, while the fractal traffic retains much of its burstiness.

To explain the traffic parameters related to fractal (LRD) traffic, we shall reuse notation defined above for the discrete time models. Suppose $V(n) = C_1 \text{ x } n^{\beta}$. If $\beta = 1$, we have a linear relationship between $V(n)$ and n, so v is finite, and the process is SRD. If $1 < \beta < 2$, $V(n)/n$ approaches infinity as n grows, and the process is LRD. Stationarity of the sequence ensures that $\beta \leq 2$, and the only case where $\beta = 2$ is where successive traffic values are identical.

Since $\log V(n) = C_2 + \beta \text{ x } \log n$, β is the slope on a log-log graph of $V(n)$ versus n. This slope is related to the *Hurst parameter*, $H = \beta/2$. (The name of Hurst derives from work on water flow in which long-range dependence of time series was first identified). The higher the value of H (or β), the higher the level of autocorrelation, and consequently the worse the queuing performance.

The M/Pareto Model

Fractal traffic is characterized by significant long bursts [1]. Such bursts are caused by downloading large files, or long periods of high levels of VBR video activities, intensive bursts of database activity, and so on. It is therefore appealing to model fractal traffic by a model involving such long bursts. The M/Pareto model was used in [6] and is basically a Poisson process with rate λ of Pareto-distributed overlapping bursts. For more details of the statistics of this model, see [7] and the references cited therein.

Each burst, from the time of its arrival, will continue for a random period. During that random period, the cell arrival process during a burst is constant. Let r be its rate. The period that represents the length of the burst has a Pareto distribution. The probability that a Pareto-distributed random variable X exceeds threshold x is

$$P\{X > x\} = \begin{cases} \left(\dfrac{x}{\delta}\right)^{-\gamma}, & x \ge \delta, \\ 1, & \text{otherwise,} \end{cases}$$

$1 < \gamma < 2$, $\delta > 0$. The mean of X is $\delta/(\gamma - 1)$ and its variance is infinite. Thus, the mean number of cells within one burst is $r\delta/(\gamma - 1)$. The mean amount of work arriving within an interval of length t in the M/Pareto traffic model is $\lambda t r\delta/(\gamma - 1)$.

By [7], the variance of the work arriving in an interval of length t for this M/Pareto model is

$$\sigma^2(t) = \begin{cases} r^2\lambda t^2\left(1 - \dfrac{(\gamma-1)t}{\gamma\delta}\right), & 0 \le t \le \delta \\ r^2\lambda\delta^2\left\{\dfrac{\delta^{-2H}t^{2H}}{H(2H-1)(3-2H)} - \dfrac{2(1-H)t}{(2H-1)\delta} + \dfrac{1-H}{3H}\right\}, & t > \delta \end{cases}$$

in which $H = (3 - \gamma)/2$. For large t, the dominant term here is

$$r^2\lambda\delta^2\frac{\delta^{-2H}t^{2H}}{H(2H-1)(3-2H)}.$$

Since the growth of this function is proportional to t^{2H}, this model is described as asymptotically self-similar with Hurst parameter $H = (3 - \gamma)/2$.

Gaussian Models

Due to the central limit theorem, the Gaussian model accurately represents aggregation of many traffic streams. This has been proved in [4] and is illustrated by simulation later in this article. Fractional Brownian noise traces used in the simulations described in this article were generated by the technique

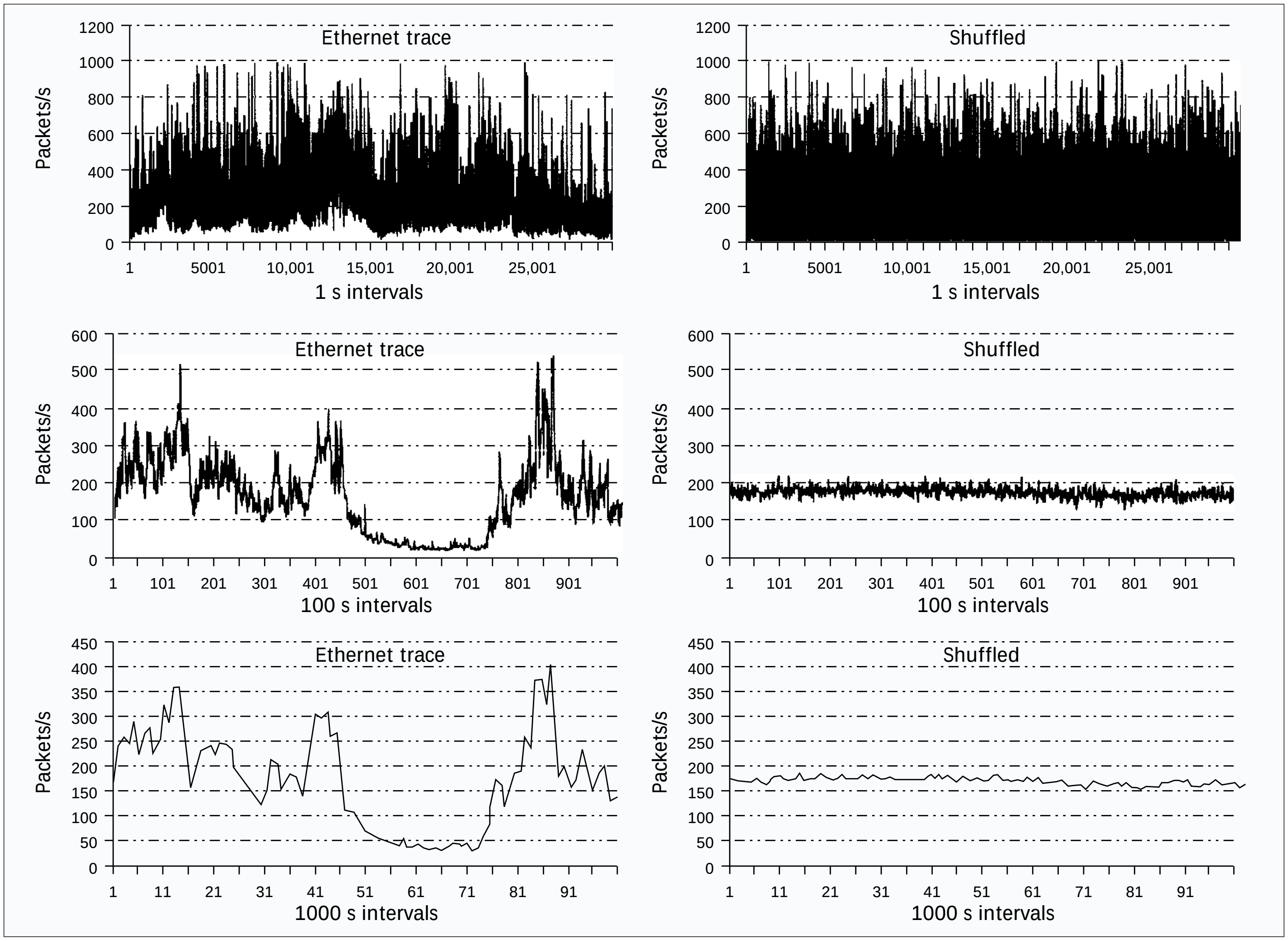

Figure 5. *Self-similarity.*

described in [8].

The Discrete Gaussian Model

Consider a FIFO single server queue, let time be divided into fixed-length sampling intervals, and consider all the definitions and notation of the discrete time queuing model described earlier. We now suppose that the arrival process, namely the $\{A_n\}$ process, is not only stationary and ergodic, but also Gaussian. We allow for any autocorrelation function for the arrival process; that is, we allow the AVR (namely v) to be finite (the SRD case) or infinite (the LRD case). We shall now present performance results for the overflow probabilities for these two cases. Notice that for the SRD case the tail of the overflow probability function is negative exponential, while this is not true for the LRD case.

The Short-Range Dependent Case — For the SRD case, the tail of the overflow probability (or the unfinished work distribution) is exponential. By [5], it can be approximated by

$$P\{V\infty > t\} \approx \tilde{c}e^{s^*t},\ t > 0, \tag{3}$$

where

$$s^* = \frac{2m}{v},\ \tilde{c} = -s^*\psi(-m,\sigma)/erf\left(-m\sigma/(v\sqrt{2})\right),$$

and

$$\psi(x,\sigma) = \frac{\sigma}{\sqrt{2\pi}}e^{-\frac{x^2}{2\sigma^2}} - \frac{x}{2}erf\,c\left(\frac{x}{\sigma\sqrt{2}}\right). \tag{4}$$

The function $\psi(m, \sigma)$ represents the mean of the random variable X^+, where X is a normally distributed random variable with mean $-m$ and standard deviation σ. The result obtained for the rate of the tail s^* is exact, while that obtained for weight of the tail is an approximation.

Because the service rate τ is fixed and independent of the traffic, the critical three traffic parameters are $E(A_n)$, σ^2, and v. These three parameters are additive in the sense that the parameters for traffic formed by aggregating two streams can be obtained by adding the parameters of those streams.

The Long-Range Dependent Case — As mentioned above, in many applications traffic is LRD, and in this case the results from the previous subsection do not apply. Under the LRD case v is not finite, and the unfinished work distribution does not have a dominant exponential tail. Fortunately, as shown in [3] (although a blunder introduced an incorrect factor in that article), we can apply the SRD results to the LRD case and obtain the following approximation for the overflow probability:

$$P\{V_\infty > t\} \approx \frac{\sqrt{2\pi}}{\sigma}\psi(-m,\sigma)e^{s^*(t)t} \tag{5}$$

in which

$$s^*(t) = -\frac{1}{2\sigma^2}|1-H|^{-2}\left(\frac{H}{|(1-H)m|}\right)^{-2H}t^{1-2H}.$$

Equation 5 has recently been validated by simulation experiments [9] and was shown to be quite accurate as long as the Hurst parameter takes on values larger than 0.5, which is usually the case. Figure 6, from [9], illustrates the accuracy of Eq. 5.

Fractional Brownian Noise vs. Discrete Time Models

Interest in Gaussian models for communication networks has been growing steadily over the last few years, and an increasing collection of results have accumulated, particularly for continuous time models. The continuous time framework is more attractive because it avoids the need to select a somewhat arbitrary sampling interval. The primary candidate for a Gaussian continuous time model is fractional Brownian noise (FBN), which may be thought of as the first derivative of fractional Brownian motion (FBM). An FBM process $\{Z_t\}$ is characterized by the property

$$\mathrm{Var}(Z_t) = \sigma^2 t^{2H},$$

in which H is the Hurst parameter, and σ^2 is the variance of the amount of traffic arriving in a unit time.

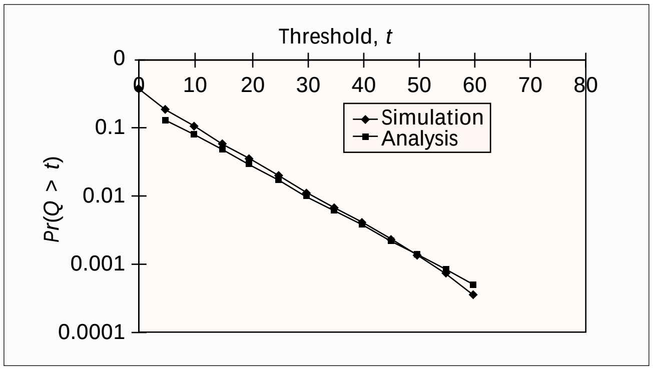

Figure 6. *Comparison of analytical and simulation results for overflow probability in a Gaussian fractal queue.*

Recently, rather precise results have become available for this model [10, 11]. Somewhat surprisingly, these present quite a different picture from that implied by Eq. 5. One of the very clear implications of Eq. 5 is that at low traffic levels, the probability that there will be any queuing at all in switch buffers will be insignificant, whereas the continuous time model seems to imply that the probability that there will be some traffic queued will be not substantially different from 1 [10].

This contrast forces us to rethink the apparent similarity between the discrete time and continuous time models. In the discrete time model, within a sampling interval no queuing takes place. Instead, all the work arriving during a sampling interval is simply added together (positive and negative work, if you will, according to the somewhat nonphysical framework of Gaussian models) and processed at the end of the sampling interval. For this reason, it is important that the sampling interval chosen in a discrete time Gaussian model not be too long. On the other hand, if the chosen sampling interval is too short, the frequency of intervals during which "negative work" arrives becomes quite high, which results in the model becoming less realistic. In effect, the sampling interval is another parameter of the model, which must be chosen appropriately.

Traffic Aggregation

Convergence to Gaussian

The central limit theorem applies to traffic processes, and as a consequence, as more traffic is aggregated together, by the sharing of a link by more and more traffic streams, traffic becomes more Gaussian, or, more formally, weakly converges to a Gaussian process [4]. This relies on the assumption that the traffic arriving in a given time interval has finite variance, a fact which seems to be confirmed by measurements. Moreover, this is the right type of convergence to apply to network traffic; it is appropriate to say that if sufficient traffic is aggregated, a Gaussian model behaves much the same as the aggregated traffic [4].

Thus, at some stage in the future, as more and more sources contribute their load to our shared public networks, Gaussian models should become a good approximation to their traffic. Please note that the central limit theorem does

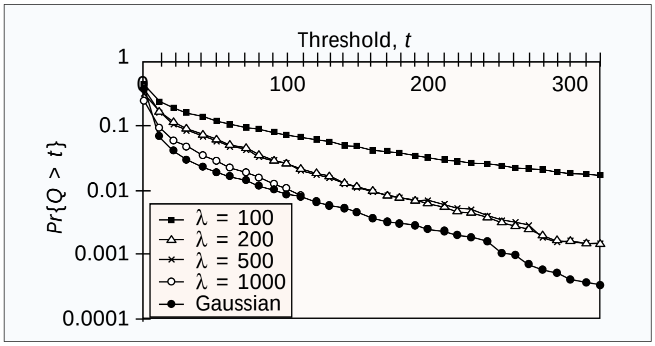

Figure 7. *Traffic aggregation toward a Gaussian process.*

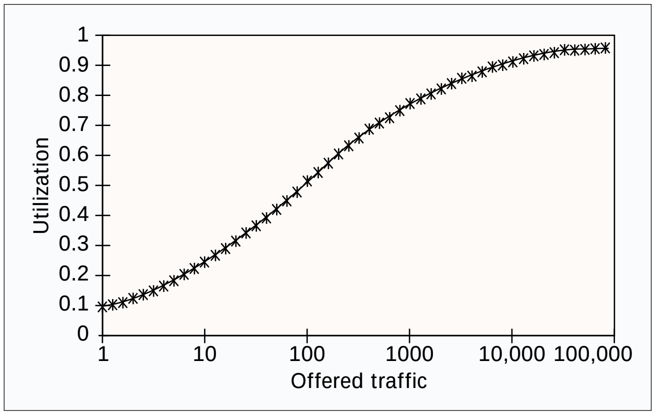

Figure 8. *Demonstration of multiplexing gain.*

not show that traffic will tend, in the limit, to a continuous time Gaussian model, such as FBN. The behavior of such models at very small timescales (events very close together) is not necessarily similar to the behavior of real systems — unless we make further assumptions.

Figure 7 shows a particular case where the traffic processes being aggregated together are all M/Pareto. Under these circumstances, the aggregate traffic is also M/Pareto (because the M/Pareto model is closed under superposition) as long as all the bursts are identically (Pareto) distributed. Observe that as the level of aggregation increases, the performance curves move closer to the curve for the Gaussian case, confirming the results of [4].

Multiplexing Gain

Theoretical and practical studies assuming a variety of models, including the Gaussian traffic model, have often come to the conclusion that the additional margin of transmission capacity required by a traffic stream to be carried with adequate performance is not lessened (when measured as a percentage of total carried traffic) by the aggregation of several traffic streams together. This conclusion follows, for example, if we use a strategy for guaranteeing adequate performance based on the constraining the performance parameter s^*, as in Eq. 3.

However, the parameter s^* is not the whole story, and Eq. 3 is only an approximation. Here is an alternative analysis of multiplexing gain which comes to a very different conclusion.

Let us now assume that performance is, at least for sufficiently large networks, determined solely by first and second order (mean, variance and autocovariance) characteristics of the carried traffic. Whether the traffic is short or long range dependent has no impact on the argument of this subsection.

With a suitable choice of server speed (i.e., we upgrade our transmission link appropriately), a system handling k times as much traffic (assuming aggregation of independent sources of traffic) is *similar* to the original system in the sense that the buffer contents distribution can be obtained by rescaling the original curve.

Let $\{X_n\}_{n \in \mathbf{Z}}$ denote the original traffic, with mean μ. Suppose the original service rate is τ, so the net mean is $m = \mu - \tau$ and suppose $\mathrm{Var}(\Sigma_{j=1}^{n} X_j) = V(n)$; that is, the variance-time curve is arbitrary.

Now consider traffic $\{Y_n^{(k)}\}$ which is obtained by aggregating together k independent traffic streams statistically identical to $\{X_n\}$; and for this traffic let us use a faster service time, $\tau_k = k\mu - \sqrt{k}m$. That is, the *performance margin* of this model, the extra capacity of the server above the rate at which traffic arrives, is $\sqrt{k}|m|$ (i.e., $\sqrt{k}$ times the original performance margin).

The variance-time function for $\{Y_n^{(k)}\}$ is $kV(n)$. Now let us rescale the aggregate model by measuring work in units $\sqrt{k}$ times as large as the original units. The variance-time curve of $\{Y_n^{(k)}\}$ in these units is therefore $V(n)$, exactly the same as the original traffic, and the mean net input into this system, in the chosen units, is

$$\frac{kE(X) - \tau_k}{\sqrt{k}} = m,$$

also the same as the original. This shows that when expressed in units x $\sqrt{k}$, the stationary queuing distribution of the system is the same as the original system.

In order to achieve a stationary distribution which is stable in units proportional to $\sqrt{k}$, which actually implies less and less buffering when measured in the time it takes to transmit the work in the buffer, it is adequate that the margin above $kE(X)$ increase at the somewhat slower rate of $\sqrt{k}$.

As a proportion of the total system capacity, the performance margin steadily reduces. This is a concrete representation of the *multiplexing gain* which can be expected as networks become larger and traffic is aggregated. This is illustrated in Fig. 8. This formula for multiplexing gain will not apply until a Gaussian approximation is valid.

M/Pareto Modeling of a Non-Gaussian Traffic Stream

The problem of how to accurately characterize real traffic has been considered unsolvable for many years. Given the insight of the previous section, the answer to fractal traffic modeling may be quite simple. There were many failed attempts to fit the mean, variance, and Hurst parameter of the real traffic to that of a model like Gaussian fractal or M/Pareto [6], and to match the queuing curve (overflow probability versus threshold). Observing Fig. 7, where we can see a family of different queuing curves, all with the same three key parameters — m, variance, and Hurst parameter — we might speculate that the missing link was the *level of multiplexing*, the λ of the M/Pareto model. In Fig. 9, we demonstrate that we can fit the

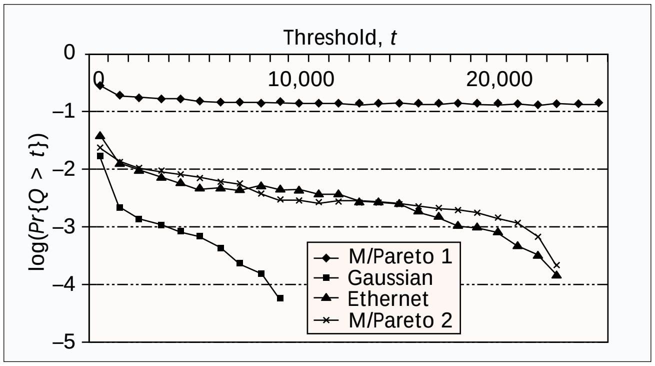

Figure 9. *Modeling an Ethernet trace.*

M/Pareto to an Ethernet trace in which insufficient multiplexing exists for the trace to be Gaussian.

In Fig. 9, the curve designated M/Pareto 1 represents an arbitrary fit of the three parameters: m, variance and Hurst parameter of the real traffic to an M/Pareto model. The curve designated M/Pareto 2 represents, in addition to the three parameters, a careful choice of λ and r.

Conclusions

In this article we attempt to look some distance into the future, using the best information we have about traffic in networks together with a full recognition of the rapid, and inevitable, growth of integrated multiservice networking. Fully accounting for this growth leads us to consider how to study the statistics of aggregation and, somewhat surprisingly, one of the oldest and most basic facts of statistics can be pressed into service very effectively — the central limit theorem applies to traffic in networks and leads us to conclude that as networks get larger, and carry traffic from more independent sources, inevitably this traffic will become closer to Gaussian, not just superficially but in the wholehearted sense that the performance experienced by such aggregated traffic will be similar to that of Gaussian traffic. This has been proved elsewhere, and is illustrated above.

We say that Gaussian traffic has the very attractive property that as a link grows, the *performance margin* required to provide good service only expands in proportion to the square root of the expanding traffic. This is a simple way to quantify multiplexing gain. This attractive feature of traffic will only show itself when aggregation levels are adequate.

The M/Pareto model has been used in this article for traffic which is not sufficiently aggregated for the Gaussian model to apply. A step has been made toward showing that real traffic can be modeled as M/Pareto. The M/Pareto model has the attractive feature of having sufficient parameters to replace the basic parameters of the Gaussian model, and to have one parameter still to spare, which can be used to allow for the level of aggregation.

References

[1] J. Roberts, U. Mocci, and J. Virtamo, *Broadband Network Teletraffic, Final Report of Action COST 242*, Springer, 1996.

[2] W. E. Leland *et al.*, "On the self-similar nature of ethernet traffic (extended version)," *IEEE/ACM Trans. Networking*, vol. 2, no. 1–15, 1994.

[3] R. G. Addie, M. Zukerman, and T. Neame, "Fractal traffic: Measurements, modeling and performance evaluation," *Proc., IEEE INFOCOM '95*, Apr. 1995.

[4] R. G. Addie, "On the weak convergence of long range dependent traffic processes," submitted to *J. Stat. Planning and Inference*.

[5] R. G. Addie and M. Zukerman, "An approximation for performance evaluation of stationary single server queues," *IEEE Trans. Commun.*, Dec. 1994.

[6] T. Neame *et al.*, "Investigation of traffic models for high speed data networks," *Proc. ATNAC '95*, 1995.

[7] P. J. Davy, "Self-similar processes with non-negative stationary increments," submitted to *J. Stat. Planning and Inference*.

[8] W. Lau *et al.*, Self-similar traffic generation: The random midpoint displacement algorithm and its properties," *Proc. ICC '95*, 1995.

[9] N. Lavaud, "The fractional Brownian motion: Traffic model and related studies," Master's thesis, Monash Univ., 1998.

[10] O. Narayan, "Exact asymptotic queue length distribution for fractional Brownian traffic," *Adv. in Perf. Anal.*, vol. 1, 1998.

[11] I. Norros *et al.*, "A benes formula for a buffer with fractional brownian input," *Proc. 9th ITC Specialists Sem.*, 1995.

Biographies

Ron Addie (addie@usq.edu.au) completed his B.Sc. and Ph.D. at Monash University in Melbourne. He worked for 20 years at Telstra Research Laboratories, on network performance analysis, design, and architecture. He has worked on traffic models for ATM networks for a number of years, focussing particularly on Gaussian models. For the last five years he has worked at the University of Southern Queensland, Toowoomba, Australia.

Moshe Zukerman [SM] (m.zukerman@ee.mu.oz.au) received his B.Sc. in industrial engineering and management and his M.Sc. in operation research from Technion, Israel Institute of Technology, and a Ph.D degree in electrical engineering from the University of California at Los Angeles in 1985. He was an independent consultant with IRI Corporation and a postdoctoral fellow at UCLA during 1985-1986. From 1986 to 1997 he served at Telstra Research Laboratories (TRL), first as a research engineer and from 1988 as a project leader managing a team of researchers who conducted research and provided advice to Telstra on design and traffic management and operation of its modern networks, and on traffic aspects of evolving telecommunications standards. He was the recipient of the Telstra Research Laboratories Outstanding Achievement Award in 1990. In 1997 he joined Melbourne University, where he is responsible for promoting and expanding telecommunications research and teaching in the Electrical and Electronic Engineering Department. Since 1990 he has also taught and supervised graduate students at Monash University. He has served as session chair and member of technical and organizing committees for numerous national and international conferences. He gave tutorials in several major international conferences like ICC '97 and GLOBECOM '97. He also served on the editorial board of the *Australian Telecommunications Research Journal* during 1991–1996. Presently, he is serving on the editorial board of the *International Journal of Communication Systems*, and as a guest editor of *IEEE JSAC* for an issue on future voice technologies. He has submitted contributions to and represented Australia in several ITU-T/CCITT standards meetings. He has published over 100 papers in scientific journals and conference proceedings, and has been awarded nine national and international patents.

Timothy D. Neame (t.neame@ee.mu.oz.au) received a B.E. degree with first class honors in electronic engineering and a B.Sc. degree in information technology from the University of Western Australia in 1993. Between 1994 and 1998 he worked as a research engineer at Telstra Research Laboratories, working on evaluation of MAN and WAN traffic models and development of interactive video services. He is currently a doctoral student in electronic engineering at