Performance Evaluation 107 (2017) 1–33

Contents lists available at ScienceDirect

Performance Evaluation

journal homepage: www.elsevier.com/locate/peva

Multifractal and Gaussian fractional sum–difference models for Internet traffic

David Anderson [a], William S. Cleveland [b,*], Bowei Xi [b]

[a] Department of Mathematics, Xavier University of Louisiana, New Orleans, LA, United States
[b] Department of Statistics, Purdue University, West Lafayette, IN, United States

ARTICLE INFO

Article history:
Received 19 January 2016
Received in revised form 28 October 2016
Accepted 5 November 2016
Available online 15 November 2016

Keywords:
FARIMA
Multiplexing
Self-similar
Long-range dependence
Synthetic traffic generation
Network simulation

ABSTRACT

A multifractal fractional sum–difference model (MFSD) is a monotone transformation of a Gaussian fractional sum–difference model (GFSD). The GFSD is the sum of two independent components: a moving sum of length two of discrete fractional Gaussian noise (fGn); and white noise. Internet traffic packet interarrival times are very well modeled by an MFSD in which the marginal distribution is Weibull; this is validated by extensive model checking for 715,665,213 measured arrival times on three Internet links. The simplicity of the model provides a mathematical tractability that results in a foundation for understanding the statistical properties of the arrival process. The current foundation is time scaling: properties of aggregate arrivals in successive equal-length time intervals and how the properties change with the interval length. This scaling is also the basis for the widely discussed multifractal wavelet models. The MFSD provides a more fundamental foundation that is based on how changes in the fGn and white noise components result in changes in the arrival process as various factors change such as the aggregation time length or the traffic packet rate. Logistic models relate the MFSD model parameters to the packet rate, so only the rate needs to be specified in using the MFSD model to generate synthetic packet arrivals for network engineering simulation studies.

© 2016 Elsevier B.V. All rights reserved.

1. Background

1.1. Internet technology and network engineering

Internet traffic results from the transfers of information between pairs of computers, or hosts, across the Internet [1–3]. For simplicity we refer to the information as a "file". The file is broken up into packets with sizes typically up to 1460 bytes = 11 680 bits. The packets are sent from the source host over a path consisting of routers connected by transmission links, and the file is reassembled at the destination host. The two hosts establish a connection to carry out the transfer, which means each is listening for the arrival of packets from the other. Headers, typically 40 bytes in size, are added to each packet to manage the file transmission and packet routing. In addition, both hosts can send control packets with no file data, just headers, as part of the transmission management. This means that packet sizes range from 40 bytes to 1500 bytes. Each router has input links and output links; when a packet arrives on an input link, the router reads a field in the header to determine the destination host, and looks in a table to determine the output link over which the packet should be sent to get to the destination.

* Corresponding author.
E-mail addresses: danders2@xula.edu (D. Anderson), wsc@purdue.edu (W.S. Cleveland), xbw@purdue.edu (B. Xi).

http://dx.doi.org/10.1016/j.peva.2016.11.001
0166-5316/© 2016 Elsevier B.V. All rights reserved.

Each transmission link on the Internet at each point in time can be servicing many ongoing connections. The packet arrival times for transmission on the link are a superposition of the packet arrival times of the individual ongoing connections. Interestingly, the term "superposition" is used in statistics, but in network engineering, the term is "statistical multiplexing". We use the latter here to remind us that this area of Internet research is about statistics. If a packet arrives for transmission and the link is busy transmitting, then the arriving packet is put in a queue. The interface that writes the packet to the link has a speed in bits/second that determines the service time, the packet size in bits divided by the link speed. The queueing is a major factor in quality-of-service (QoS) for Internet connections; if queueing delay is too large, QoS degrades [4].

This work addresses a common type of traffic being carried on most Internet links. The traffic consists of a very wide range of applications such as downloading Web pages, sending email, logging in remotely to a computer, and streaming video or audio. We name this "multi-application traffic". We shall see from the work here that the statistical multiplexing homogenizes the traffic once there are enough connections using a link. This is just a beginning of the well known central limit theorem of point processes: they tend toward Poisson as the number of multiplexed processes increases. The detailed arrival behavior of individual applications is washed out by this, making statistical modeling possible. As we will see, this tends to happen at quite low traffic rates. More precisely we want to model the "offered load" on a router. This is traffic that arrives onto the back-plane of the router from input interface links and is destined for a specific output interface link.

1.2. The need for a validated statistical model for the multiplexed arrival process of the offered load

A statistical model for the offered load can be used to generate traffic for simulations which determine the maximum traffic load on the link that can achieve quality of service (QoS) standards. Packets from the back-plane going to the link are held in a queue to await transmission on the link if the link is busy. Traffic, in bits/second, is written to the link at the link speed, which is also in bits per second. So wait times are in the queue are determined by the relative sizes of the traffic rate and the link speed. The traffic load is chosen to achieve acceptable queue wait-time distributions.

One might think that this traffic engineering would be easy. Just institute a measurement program that collects arrival times and packet sizes of each packet in many streams of packets on many routers for different rates and types of traffic. The problem is that the Internet is not instrumented to do this. Even if it were, the complexity of finding just the right traffic conditions needed in an experiment would be quite hard, and impossible when the conditions do not yet exist. The most practical route for network engineering study is to run computer simulations with packet arrival times as inputs to a queue, and queueing delay as the output.

A simulation for network engineering requires a model for the arrival process. The queueing properties are determined by the statistical properties of the process [5–8]. We take the arrival process to be the sequence of interarrival times, t_u. The interarrivals have very complex statistical properties when studied directly, which in the past has made modeling complex. Three properties account for the complexity—long-range dependence, non-Gaussian behavior, and changing statistical properties with the packet arrival rate $\alpha = 1/\mathrm{E}(t_u)$.

It is critical that a model be valid. Validation must be carried out in substantial detail. In addition, to be most useful for simulation, there needs to be a way to accomplish fast generation of packet arrivals, especially at high packet rates. Interestingly, while there has been much past work in describing the statistical properties of the arrival process, cited in coming sections, there is still not a validated model that provides fast generation. The barrier has been the complexity due to the long-range dependence, nonlinearity, and changing statistics with the arrival rate. This paper contains a very substantial amount of validation. Not only do we look just at the model itself, but rather we drive properties of the model, and then check the derivations by comparing with empirical estimation of the properties based on the data.

The long-range dependence was discovered in the 1990s and reported in two pioneering articles [9,10]. Here, we take long-range dependence to mean that as the frequency f goes to zero, the power spectrum increases like f^{-2d} for $0 < d < 0.5$, which means as the lag k gets large, the autocorrelation function decreases like k^{2d-1}. These statistical properties make the arrival process "bursty", in the language of network engineering. Compared with Poisson arrivals that have the same arrival rate, the upper tail of queueing delays is longer, and the average amount of traffic that can be put on the link and maintain QoS is less [11–15].

One must, however, treat the notion of burstiness with immense care because there is more to the story. Consider the number of packet arrivals in a fixed interval of time. It has a mean and a standard deviation. Consider the ratio of the standard deviation to the mean, the coefficient of variation. We will see clearly the following properties here. As the traffic rate increases, the long-range dependence remains, but the coefficient gets smaller and smaller. So the traffic remains "bursty" in the technical sense, but eventually is not salient because the traffic gets smoother and smoother. In the early days after the discovery of long-range dependence, this smoothing was not appreciated and led to the wrong engineering concepts for the core of the Internet where traffic rates are high.

As one would expect, the interarrival sequence is non-Gaussian. It is expected because it is a waiting time until an event occurs and is a positive random variable. So we can expect the process to be nonlinear (non-Gaussian) since just the marginal distribution of the process is non-Gaussian.

One other matter must be considered for traffic engineering simulation. The statistical properties of the arrival process changes with the expected number of ongoing connections. The change is not just a change in α, the rate, but rather a profound change in the multivariate distributions of any sequence of m consecutive interarrivals. This is a very general result for point processes [16]. Traffic on a link has a "deterministic" component. By this we simply mean that the expected value

of the traffic has diurnal and day-of-the-week patterns. Usually, traffic engineering is carried out by running simulations adjusting the mean of the generated traffic to match that of the maximum. Service providers must provide satisfactory QoS at all times.

1.3. Article content

This article investigates the multifractal fractional sum–difference (MFSD) model, developed for t_u. The model is simple and very easy to understand. This arises from the discovery that a monotone nonlinear transformation, $z_u = T(t_u)$, is a long-range dependent Gaussian process that we call a Gaussian fractional sum–difference (GFSD) model. z_u is a linear combination of a long-range dependent process and white noise. Details are in Section 3.

MFSD and GFSD models were first put forward in [17,18]. In Section 15 we discuss what has been added in this paper. We also compare the MFSD and GFSD models with other models.

The properties of t_u – long-range dependent, non-Gaussian, changing statistical properties with the traffic rate α in packet/second (p/s) – when addressed without benefit of the model, appear immensely complex, requiring complex summary statistics to characterize their behavior. The MFSD and GFSD models enable deep insight into traffic properties. In this paper, we recast the model using Hosking's discrete-time analog [19] of fractional Gaussian noise (fGn). This allows us to derive for many informative quantities such as autocorrelations, exact or approximate formulas. This gives immense insights that in the past have only been described empirically. We also carry out a very broad and deep validation of the models.

One derivation provides a simple and fast mechanism for generation of arrivals for simulation. It is simple because only the traffic rate needs to be specified. It is fast, not because it is a fast algorithm, but rather an approximate model based on sums of m consecutive values of t_u in non-overlapping blocks. As we have emphasized, simple and fast generation is very important for network engineering studies. This is illustrated here by describing one simulation study using the approximation of the MFSD.

2. Past statistical foundations and modeling: self-similarity and fGn

In the earliest papers, packet arrival counts were described as self-similar and fractional Gaussian noise (fGn) was put forward as a model [20,9,10,21–24]. Using this fGn model, [25] derived a number of statistical properties, and investigated queueing properties as a part of network engineering study.

fGn is a tractable model and allows mathematical investigations, for example, the above work of [25]. The problem, however, is that more traffic analysis resulted in subsequent articles showing that fGn is not an adequate model; the arrival process is not self-similar across, and arrival counts for small interval lengths are non-Gaussian [26–28].

The inadequacy of fGn led to intensive study of time scaling properties and the development of new models based on scaling. Work focused on m-sums and m-means, defined as follows. For $u = 1, 2, \ldots$, let w_u be a time series. For positive integer m, the m-sum process consists of every mth value of a moving sum of length m:

$$w_v^{(m)} = \sum_{i=1}^{m} w_{(v-1)m+i}, \quad v = 1, 2, \ldots.$$

The m-mean process is $\bar{w}_v^{(m)} = w_v^{(m)}/m$. In a very large literature, the statistical properties of these m-statistics and how they change with m were studied in many ways [27,29–35,26,36–46,28,47–51].

Multifractal wavelet models based on the statistics were developed [32,43,52–54]. In almost all cases, w_u was taken to be arrival counts. In a few cases, the w_u were taken to be interarrival sequences [32,52]. The time scaling analyses and modeling formed a foundation for intuition about the statistical properties of the arrival time process.

In [55], the computation and communication workloads on the data-centers-on-chip (DCoC) show multifractal behavior. However the properties of the inter-event times are different from the network traffic, e.g., the increments are non-stationary and statistically asymmetric. [55] use fractional calculus and apply statistical physics concepts to model the DCoC workloads.

One widely-used method of scaling analysis is the variance-time plot [9,12,32,45,56]: the log of the sample variance of $\bar{w}_v^{(m)}$ is plotted against log m. Another example is autocorrelation-time analysis in which the standard nonparametric estimate of the autocorrelation function of the m-means is studied as a function of log m [35].

Another method is multifractal moment analysis, a study of the moments of normalized values of $w_v^{(m)}$. This is closely associated with the multifractal wavelet models in the above citations. The moments of the m-sums of packet counts in the measured traces at smaller time scales exhibit complex behavior, and tend toward mono-scaling at large time scales [29,8,26,27,57,58]. Multifractal moment analysis studies traffic's complex scaling properties, in particular traffic's high variability at small time scales. The multifractal concept is an enlargement of self-similar processes. Self-similar processes have a certain uniformity in the moments that make them monofractal. We have a moment-statistic study of the m-sums in Section 7, which has the details on how the multifractal moment analysis is carried out.

Multifractal wavelet models reproduce the statistical properties of packet arrival times, fixing the shortcoming of fGn. However, there are drawbacks. The wavelet models are fundamentally nonparametric, based on m-sums, which formally

means a description involving many parameters. This makes them complex, resulting in a mathematical tractability that does not readily facilitate derivations to study mathematically the many statistical quantities that provide insight into the arrival process. In addition, the models do not readily lead to a simple mechanism for synthetic traffic generation at a pre-specified packet arrival rate α for traffic engineering simulations.

In contrast to this previous work, the simple MFSD model is mathematically tractable, is an excellent fit to the arrival process at all traffic rates above 1000 packets/second, and leads to a fast traffic generator that needs only the packet rate to be specified in carrying out simulations for traffic engineering.

2.1. Section contents

The contents of the next sections are the following: Section 3 introduces the MFSD and GFSD models. Section 4 describes the live and multiplexed packet trace segments that were used for validating the models. Section 5 provides validation for the marginal distribution of the MFSD. Section 6 examines the power spectrum of the GFSD and provides validation for the GFSD. Section 7 provides validation for the MFSD through the m-sum moment-statistics. Section 8 derives equations for the parameters $\lambda(\alpha)$ and $\theta(\alpha)$ in terms of α. Section 9 examines the autocorrelation function of the GFSD and provides validation for the GFSD. Section 10 examines the autocorrelation functions of h_u, s_u and their m-scaled-sums, and discusses the near-self-similarity of h_u and s_u. Section 11 conducts a rate–time analysis of variance and autocorrelation for the GFSD. Section 12 examines the autocorrelation function of the MFSD and shows the second moment properties of the GFSD apply to the MFSD as well. Section 13 introduces a fast traffic generation method. Section 14 contains a VoIP queueing study which illustrates how the traffic generated using the MFSD model can be used in such studies. Section 15 concludes the paper.

3. GFSD and MFSD models

The GFSD and MFSD models employ the fractional ARIMA (FARIMA) models of Hosking [19]. ARIMA models involve autoregressive and moving average polynomials in the backward shift operator B. The FARIMA model multiplies the autoregressive polynomial by $(I - B)^d$ for $-0.5 < d < 0.5$, the fractional exponent. $(I - B)^d$ is defined by expanding in a power series in B. d is restricted to the shown range to make the FARIMA stationary. We restrict the exponent further to positive values because this enables modeling persistence, autocorrelations that fall off slowly, which is appropriate for the properties of Internet traffic.

3.1. h_u and s_u

Let h_u be fractionally differenced white noise, the Hosking discrete analog of fractional Gaussian noise (fGn),

$$(I - B)^d h_u = \epsilon_u.$$

ϵ_u is Gaussian white noise with mean 0, and variance σ_ϵ^2. We take

$$\sigma_\epsilon^2 = \frac{(1-d)\Gamma^2(1-d)}{2\Gamma(1-2d)}$$

for purposes stated below; the resulting variance of h_u is $\sigma_h^2 = (1-d)/2$.

Let s_u be a moving sum of length 2 of h_u,

$$s_u = h_u + h_{u-1}.$$

s_u can be written in another form,

$$(I - B)^d s_u = \epsilon_u + \epsilon_{u-1},$$

so s_u is a fractional moving-average process [19]. The above value of σ_ϵ^2 makes the variance of s_u equal to 1.

3.2. z_u, the GFSD, and t_u, the MFSD

Let n_u be Gaussian white noise with variance 1. A Gaussian fractional sum–difference (GFSD) model for a time series z_u has the form

$$z_u = \sqrt{(1-\theta)}s_u + \sqrt{\theta}n_u,$$

where s_u and n_u are independent processes and $0 \le \theta \le 1$. θ is the mixture coefficient of the GFSD. The mean of z_u is 0. The variance is 1 for notational convenience, and does not limit modeling.

A multifractal fractional sum–difference model (MFSD), t_u, is a stationary discrete-time series that is a nonlinear strictly monotone transformation of a GFSD, z_u. Let the cdf of t_u be $T(t)$. Let $Z(z)$ be the cdf of a Gaussian distribution with mean 0 and variance 1. Then

$$t_u = T^{-1}(Z(z_u)),$$

Table 1
Information about analyzed packet trace segments.

Factor	Auckland 15-min	Auckland 1-h
Link speed (megabits/second)	1000	1000
Transmission technology	Ethernet	Ethernet
Lab timestamp accuracy (μs)	0.030	0.030
Collection duration (days)	2.25	2.25
Collection date	March 2008	March 2008
Number live	96	24
Number live-multiplexed	24	0
Max utilization live	5.24%	4.95%
Min packet rate live (p/s)	1193	1268
Max packet rate live (p/s)	7674	7712
Min packet rate live-multiplexed (p/s)	2537	NA
Max packet rate live-multiplexed (p/s)	137 884	NA

and

$$z_u = Z^{-1}(T(t_u)).$$

z_u is the Gaussian image of t_u, and t_u is the multifractal image of z_u.

Suppose the marginal distribution of t_u is a Weibull with shape parameter λ. Let the arrival rate be $\alpha = 1/\mathrm{E}(t_u)$, measured in packets/second (p/s). Our parameterization of the Weibull is somewhat different than usual, replacing the usual scale parameter with the rate α, which is more meaningful for packet interarrivals. The cdf for this parameterization is

$$T(t_u) = W(t_u; \lambda, \alpha) = 1 - e^{-\{\alpha\Gamma(1+\lambda^{-1})t_u\}^{\lambda}}.$$

The transformation to the multifractal image is

$$t_u = W^{-1}(Z(z_u); \lambda, \alpha) = \frac{\{-\log(1 - Z(z_u))\}^{1/\lambda}}{\alpha\Gamma(1 + \lambda^{-1})}.$$

t_u is a Weibull MFSD.

For a multiplicative MFSD, the Weibull marginal is replaced by the log normal. μ is the mean of $\log(t_u)$ and τ^2 is its variance. The cdf is $T(t_u) = L(t_u; \mu, \tau^2)$. The transformation to the multifractal image,

$$t_u = L^{-1}(Z(z_u); \mu, \tau^2) = e^{\tau\sqrt{1-\theta}s_u} e^{\tau\sqrt{\theta}n_u} e^{\mu},$$

has a simple multiplicative form.

It is the Weibull MFSD that is the validated model for t_u. Section 5 discusses the observed marginal distribution of the interarrival process of the Internet traffic data, and the estimates of the Weibull parameter λ. However, for certain problems the log normal MFSD is used as an approximation because its simpler structure enables closed-form derivations. The approximation is the log normal whose first two moments match those of the Weibull.

4. Model validation: live packet trace segments

Validation was carried out by analyzing live packet traces for traffic in both directions of 3 Internet links: Auckland, Leipzig, and Bell. The total number of arrivals is 715,665,213. Bell was the Internet gateway link for a Bell Labs research location with about 500 users. Leipzig was the gateway link for the University of Leipzig campus. Auckland was a link near the edge of the University of Auckland network. All collection used Endace cards [59] to provide highly accurate, hardware timestamps, which is essential to the modeling. The collected data consist of network and transport headers, and timestamps of packet arrivals, but our analysis used only the packet size field and the timestamp.

The Bell live traces were obtained as a result of one author of this article being a part of the collection operation. The Leipzig live traces were obtained from the Center for Applied Internet Data Analysis (CAIDA) [60]. The Auckland live traces were obtained from the Waikato Internet Traffic Storage [61].

In coming sections, in the interest of space, we use just Auckland traces in our visual displays and numeric information. However, statistical properties and modeling conclusions were the same for all links. The Auckland traces available for these links were broken into trace segments of 15 min or 1 hr, and each segment analyzed individually. Not all available segments were appropriate for analysis for reasons given below. Table 1, discussed in more detail later in this section, gives information about the analyzed Auckland segments.

4.1. Stationarity

To accurately study changes in statistical properties with α, we need each trace segment to have a nearly constant expected rate for the duration of the segment. We insure this, first, by taking segments with small lengths, and second, by checking each segment for stationarity by visualization of measures of the packet rate such as the number of packets in 10 s intervals. We found that 15 min segments were typically quite close to stationary, and discarded any segments that showed more than minor nonstationarity in the mean. We also found that certain 1 h traces were close to stationary.

4.2. Packet rate above 1000 packets/second (p/s)

The MFSD model is not appropriate for packet rates α that are very low. This is associated with a small number of connections. Each connection has packet arrivals resulting from the TCP protocol that manages the communication between the two hosts of each connection. TCP creates specific patterns in the interarrivals. If there are a small number of connections, then these patterns dominate the interarrivals. There must be a sufficient amount of statistical multiplexing to "wash out" the patterns. The multiplexing is very powerful. For our data, a rate above 1000 p/s is sufficient, a very low rate indeed. We found that for rates below 1000 p/s TCP can create peaks in the power spectrum at frequencies of $1/k$ where k is a small integer greater than or equal to 2. These peaks are readily seen in estimates of the power spectrum; their frequencies change across the trace segments, likely due to changes in the Internet application that is dominant. If modeling is needed for very small rates, then a better, feasible strategy is to use simulations that run TCP, which is feasible when rates are low.

While we do not model segments with rates below 1000 p/s, we do make use of them by making them part of out numerical multiplexing of different segments that is described below in Section 4.5.

4.3. Modeled arrivals, measured arrivals, and timestamps

However, it is not practical to measure the back-plane offered load arrivals for a link on the back-plane itself, as mentioned in Section 1. Here, we provide more detail.

The data come from operational routers of service providers that are designed and finely tuned to push bits along at very high speeds. So it is not possible in almost all cases to get permission to run software to measure the back-plane which can interfere with performance. Instead, passive monitoring is carried out by putting a device on a link that mirrors the traffic and does not interfere with operations except for slight reductions in light intensities that are insignificant. This occurs not just for this research but all empirical research on Internet traffic. However, as explained next, we can get good approximations of the offered load.

The MFSD model applies to $t_u = a_u - a_{u-1}$ where the a_u are the arrival times of the offered load to a back-plane that is destined for an output interface link as explained in Section 1. However, we cannot measure the back-plane offered load arrivals for a link at the back-plane because routers are not equipped to do this, so we measure on the links the arrivals. The measured arrival time a'_u is the exit time from the queue for the link, a buffer that holds packets for transmission that arrive when the link is busy. $t'_u = a'_u - a'_{u-1}$ are the measured interarrivals.

If packet u arrives when there is no packet in service, then $a'_u = a_u$. If packet u arrives when packet $u-1$ is in the queue or in service, its transmission begins as soon as packet $u-1$ has finished, so t'_u is the service time of packet $u-1$. Let p_u be the size of packet u (bits), and let ℓ be the speed (bits/second) with which the router writes a packet to the link. Then we have $t'_u = p_{u-1}/\ell$. We refer to the t'_u as a "back-to-back interarrival". It is the smallest possible t'_u when packet $u-1$ has size p_{u-1}. The timestamps, $\tilde{a}_u$, are the a'_u plus measurement error, and the timestamp interarrivals are $\tilde{t}_u = \tilde{a}_u - \tilde{a}_{u-1}$.

4.4. Timestamp accuracy and identifying packets with queueing delay

The accuracy of timestamps is critical to the validity of the MFSD modeling. The prediction of timestamp accuracy from laboratory tests of the Endace card used for the Auckland trace segments is $\pm\phi$ where $\phi = 15$ ns. This is excellent if it is valid. We investigate ϕ empirically.

Identifying queued packets is also important for the trace segment selection upon which modeling is based. We selected live trace segments for analysis that have a small percent of delayed packets, less than about 10%, because modeling is for the t_u and not the t'_u. We need trace segments where the t'_u reflect the properties of the t_u. These segments are those with lower packet rates. We determine empirically the percent of queued packets as part of the same method that investigates accuracy.

For all delayed packets u, we have $t'_u = p_{u-1}/\ell$. Measurement errors, however result in timestamps $\tilde{t}_u$ of these delayed packets that lie in the interval $p/\ell \pm 2\phi$. Furthermore, we expect that the density of the $\tilde{t}_u$ will have a noticeable drop just above $t'_u + 2\phi$. This can lead to a revision in the value of ϕ, and allows identification of packets that experience delay.

An accuracy and delay-identification plot is shown in Fig. 1 for the Auckland live trace segment that has the largest bitrate, 33.5 megabits/second. On the plot, $\tilde{t}_u - p_{u-1}/\ell$ is graphed against p_{u-1}/ℓ for u with $\tilde{t}_u$ less than 100 ns. Because there are 690,239 such u, just a sample of the values are plotted. The horizontal lines above and below 0 are drawn at ± 30 ns, the laboratory values of $\pm 2\phi$. There is a dense band of points contained within the accuracy limits, and a sharp cutoff in density above the band. This verifies $\phi = 15$ ns, and packets within the band can be taken as the queued packets.

4.5. Numerical multiplexing

To study the changing statistical properties with the packet arrival rate α, we need trace segments with a wide range of observed traffic rates, not just the live 15-min and 1-h traces whose rates are kept small to ensure a low percent of delayed packets. To achieve larger-rate segments, we multiplexed on the computer 15-min live trace segments to produce numerically-multiplexed 15-min trace segments (as opposed to physically multiplexed). The numerically multiplexed trace

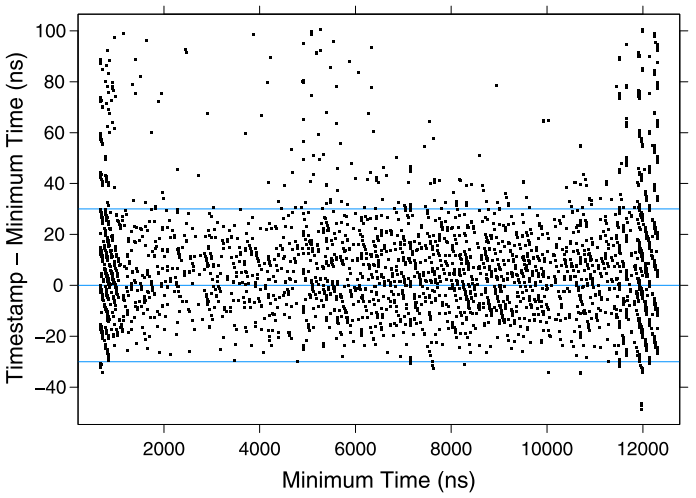

Fig. 1. Accuracy and delay-identification plot for one live Auckland 15-min trace segment.

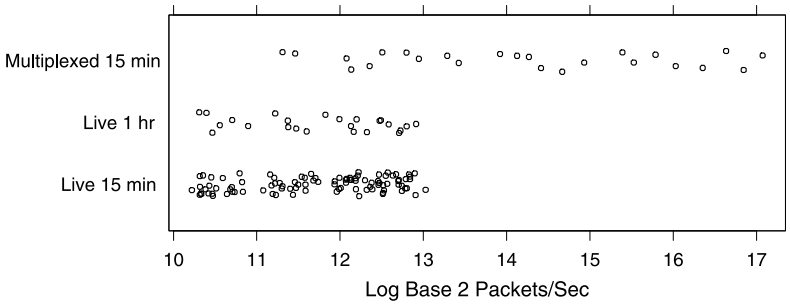

Fig. 2. Log base 2 observed packet rates of 144 Auckland trace segments.

segments are actually somewhat more realistic than our raw segments in that the former have interarrivals that are arbitrarily small as the actual interarrivals at the queue, but the latter have interarrivals with lower bounds varying by packet size. Table 1 gives information about the Auckland segments, 120 live and 24 numerically-multiplexed. Fig. 2 graphs the log packet rates of the segments.

4.6. Visual displays of 4 traces

Data visualization played a critical role in the validation process. There were many types of displays and each was applied to each analyzed trace segment. A number of these display types are shown in coming sections for 4 15-min Auckland trace segments to convey results. The packet rates of the 4 segments range from small to large, and are as close to equally spaced on a log scale as possible. The 2 with the smallest rates are live segments, and the 2 with the largest rates are numerically-multiplexed segments. The packet rates in packets/second (p/s) are $1771 = 2^{10.79}$, $5634 = 2^{12.46}$, $17\,928 = 2^{14.13}$, and $66\,913 = 2^{16.03}$.

5. MFSD model validation and properties: marginal distribution

5.1. Validation of the marginal distribution

Visualization methods were a critical part of modeling the marginal distribution of the interarrival process t_u. Quantile plots were used to check how well standard parametric distributions – Weibull, log-normal, and gamma – fitted the observed marginal distributions of the trace segments. The segments are well approximated by a marginal Weibull distribution, as illustrated below.

Let α be the packet arrival rate, the inverse of the mean of t_u, and let λ be the shape parameter. Note that rather than using the usual scale parameter for the Weibull distribution, we are using one that works well for modeling Internet traffic. For each trace segment, λ and α were estimated by the method of moments. $\hat{\alpha}$ is the inverse of the sample mean of the t_u. $\hat{\lambda}$ is the value for which a Weibull with rate $\hat{\alpha}$ has a variance equal to the sample variance of the t_u.

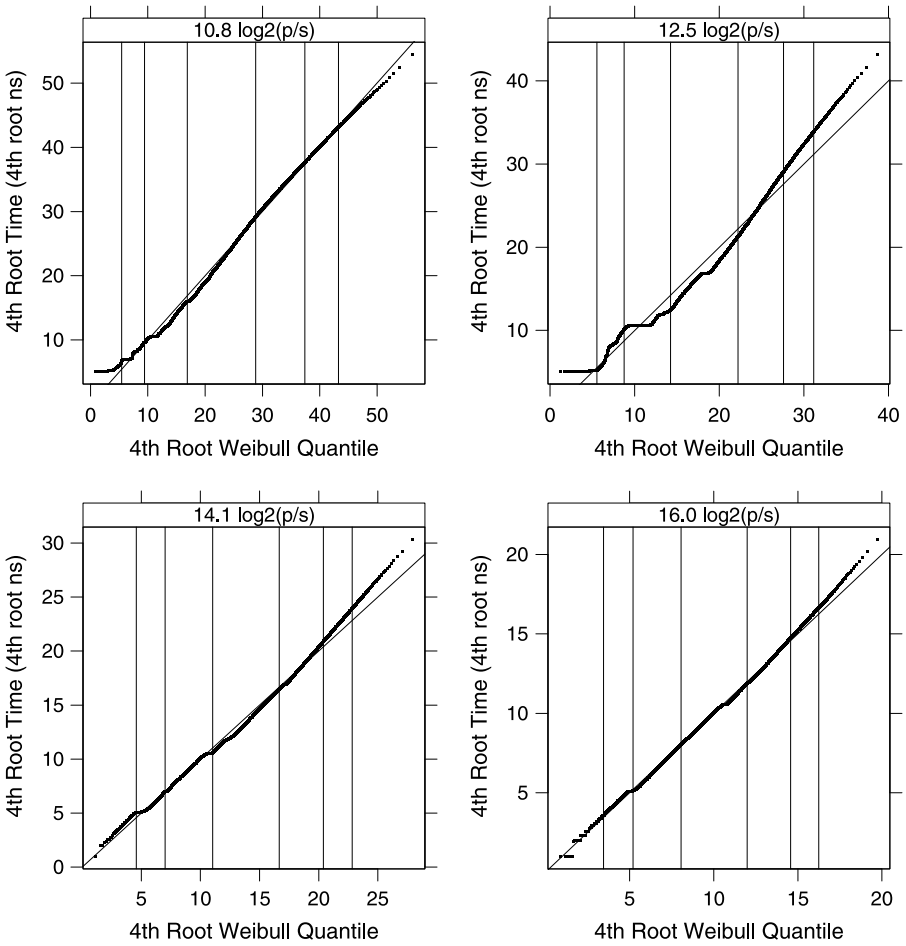

Fig. 3. Weibull quantile plots of interarrivals of 4 Auckland trace segments.

The Weibull quantile plot is illustrated in Fig. 3 for the 4 Auckland trace segments described in Section 4.6. For each segment, the fourth root of the quantiles of the observed t_u at frequencies 0.00005 to 0.99995 in steps of 0.0001 are plotted against the fourth root of the quantiles of a fitted Weibull using the above estimates. Fourth roots are taken because the resulting transformed distribution is close to symmetric for values of $\hat{\lambda}$ in the range of the trace segments. The vertical lines are drawn at the quantiles with probabilities 0.01, 0.05, 0.25, 0.75, and 0.95, and 0.99. The oblique line has slope 1 and intercept 0.

If the observed t_u are well approximated by a Weibull, then the pattern of the points on the plot follows the oblique line. In Fig. 3, and for almost all other analyzed trace segments, the Weibull provides an excellent fit, taking sampling variability and queueing artifacts into account.

There are small departures, atoms in the observed sample distributions of the live segments in the top 2 panels. The artifacts result from a small fraction of back-to-back measured interarrivals, $\hat{t}_u$, which are not the same as the modeled interarrivals, t_u, because of queueing delay; this is discussed in Section 4. The artifacts are nearly eliminated in the bottom two panels due to the numerical multiplexing, which tends to break up atoms. The deviations of the empirical distribution from the Weibull are small enough that the assumption of a Weibull is satisfactory.

5.2. The change in λ with α

Let $\hat{\lambda}_k$, $k = 1, \ldots, 144$, be the estimate of the shape λ for the kth Auckland trace segment, and let $\hat{\alpha}_k$ be the estimate of the packet arrival rate α. Fig. 4 graphs $\hat{\lambda}_k$ against $\log_2(\hat{\alpha}_k)$ where $\log_2$ is log base 2. The smallest values of $\hat{\lambda}_k$ are close to 0.6; they tend to 1 as $\log_2(\hat{\alpha}_k)$ increases, which means the marginal distribution tends to exponential.

Section 8 presents a derivation of λ as a function of α using the MFSD model. Equations are solved that yield numeric values, leading to a model $\lambda(\alpha)$ for the dependence of λ on α. The theoretical model agrees with the empirical pattern in Fig. 4. This dependence of λ on α is a critical aspect of the statistical properties of the packet arrival process, so we switch notation from λ to $\lambda(\alpha)$ in coming sections.

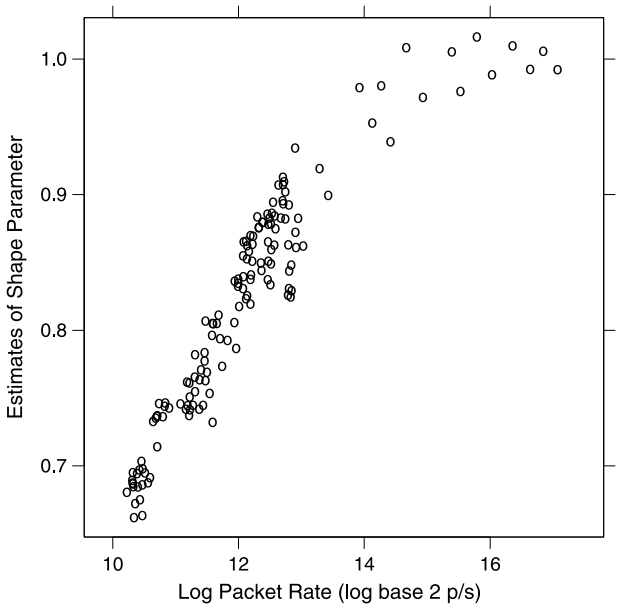

Fig. 4. $\hat{\lambda}_k$ is plotted against $\log_2(\hat{\alpha}_k)$ for the 144 Auckland trace segments.

6. GFSD model validation and properties: power spectrum

The four time series considered in the GFSD, which are defined in Section 3, are h_u, s_u, n_u, and z_u. This section presents formulas for their power spectra, which provides insight into statistical properties. Validation study is also carried out for the observed z_u of each trace segment by comparing nonparametric estimates of the power spectrum with that of a GFSD model fitted to the z_u.

6.1. Formulas and statistical properties

Let $-0.5 < d < 0.5$ and w_u be a FARIMA(p, d, q) series. It is defined as

$$\phi(B)(1-B)^d w_u = \theta(B)\epsilon_u,$$

where ϵ_u is a white noise series with variance σ_ϵ^2, $\phi(B) = 1 - \phi_1 B - \cdots - \phi_p B^p$, and $\theta(B) = 1 + \theta_1 B + \cdots + \theta_q B^q$. B is the backward shift operator, $Bw_u = w_{u-1}$. $(1-B)^d$ is expanded as a power series of B as follows.

$$\begin{aligned}(1-B)^d &= 1 + \sum_{j=1}^{\infty} \frac{\Gamma(j-d)}{\Gamma(j+1)\Gamma(-d)} B^j \\ &= 1 - dB - \frac{1}{2}d(1-d)B^2 - \frac{1}{6}d(1-d)(2-d)B^3 - \cdots .\end{aligned}$$

Let $0 < f \le 0.5$ denote frequency in units of cycles/interarrival. Power spectrum for a FARIMA(p, d, q) series is

$$p_w(f) = \sigma_\epsilon^2 \frac{\theta(e^{i2\pi f})\theta(e^{-i2\pi f})}{\phi(e^{i2\pi f})\phi(e^{-i2\pi f})} \{(1-e^{i2\pi f})(1-e^{-i2\pi f})\}^{-d}.$$

[19] showed the power spectra of h_u and s_u used in our model. In our model, h_u is a FARIMA$(0, d, 0)$ series, and $\sigma_\epsilon^2 = (1-d)\Gamma^2(1-d)/\{2\Gamma(1-2d)\}$. $\phi(B) = \theta(B) = 1$ for h_u. It has power spectrum

$$\begin{aligned}p_h(f) &= \sigma_\epsilon^2 \{(1-e^{i2\pi f})(1-e^{-i2\pi f})\}^{-d} \\ &= \frac{\sigma_\epsilon^2}{\{2\sin(\pi f)\}^{2d}} \\ &= \frac{(1-d)\Gamma^2(1-d)}{2\Gamma(1-2d)\{2\sin(\pi f)\}^{2d}}\end{aligned}$$

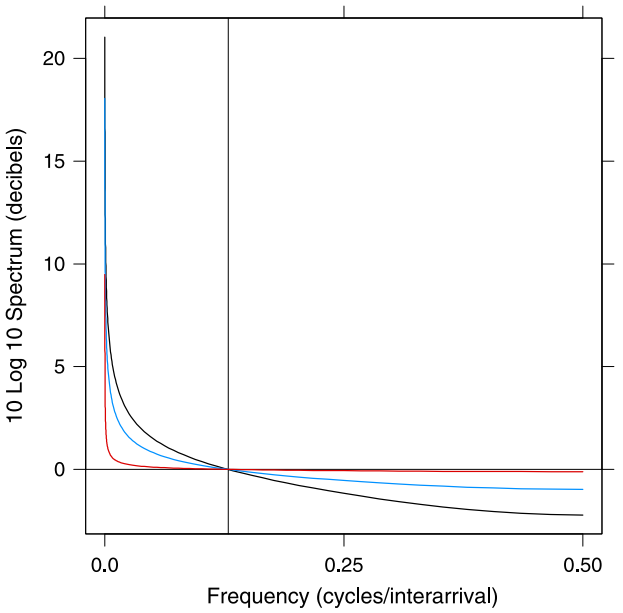

Fig. 5. Log power spectra $\ell_z(f)$ for $d = 0.31$ and 3 values of θ.

s_u is a FARIMA(0, d, 1) series with $\phi(B) = 1$ and $\theta(B) = 1 + B$. It has power spectrum

$$\begin{aligned} p_s(f) &= \sigma_\epsilon^2(1 + e^{i2\pi f})(1 + e^{-i2\pi f})\{(1 - e^{i2\pi f})(1 - e^{-i2\pi f})\}^{-d} \\ &= (1 + e^{i2\pi f})(1 + e^{-i2\pi f})p_h(f) \\ &= 4\cos^2(\pi f)p_h(f). \end{aligned}$$

White noise n_u has variance 1. Its power spectrum is

$$p_n(f) = 1.$$

Since $z_u = \sqrt{1-\theta}s_u + \sqrt{\theta}n_u$ is the sum of two independent components, it has power spectrum

$$p_z(f) = (1 - \theta)p_s(f) + \theta.$$

$p_h(f)$, $p_s(f)$, and $p_z(f)$ decrease strictly monotonically as f increases, and all go to infinity to order f^{-2d} at the origin, a signature property of the long-range dependence amply observed empirically in many previous studies. Let $\ell_z(f) = 10\log_{10}\{p_z(f)\}$ where $\log_{10}$ is log base 10. In visual displays of the power spectra, we use this decibel scale because it shows properties more effectively.

There are an infinite number of ways of decomposing z_u into a long-range dependent component plus a white noise component. The decomposition of the GFSD,

$$z_u = \sqrt{1-\theta}s_u + \sqrt{\theta}n_u,$$

is the one that maximizes the variance of the white noise because $p_s(0.5) = 0$. This means $p_z(0.5) = \theta$, which is used below in the estimation of θ.

Fig. 5 graphs 3 power spectra, $\ell_z(f)$, on the decibel scale. For each of the 3, $d = 0.31$. The values of θ are different: 0.6 (—), 0.8 (—), 0.975 (—). They span the range of the estimates of θ. $\ell_z(f)$ is evaluated at equally-spaced values of f from 2^{-16} to 1/2. The vertical line on each panel is drawn at frequency $f_0 = 0.129$ cycles/interarrival for reasons we explain next.

For fixed f and d, $p_z(f)$ is linear in θ with derivative $1 - p_s(f)$. Let f_0 be the frequency where $1 - p_s(f_0) = 0$, which means $p_z(f_0)$ and $\ell_z(f_0)$ do not change with θ. f_0 depends only on d, and for $d = 0.31$, $f_0 = 0.129$ cycles/interarrival, which has a period of 7.75 interarrivals. This is the value at which the vertical line is drawn in Fig. 5, illustrating the constancy at f_0. It is easy to see that $p_z(f)$ and $\ell_z(f)$ decrease with θ for $f > f_0$, and increase for $f < f_0$; this is also illustrated in Fig. 5.

6.2. Estimation of parameters d and θ

To carry out estimation and model checking for the GFSD model for each trace segment, the observed t_u for each segment were transformed to observed z_u by the function $z_u = Z^{-1}\{\hat{T}(t_u)\}$, where $\hat{T}$ is the empirical cumulative distribution function of the t_u, and Z is the normal cumulative distribution function with mean 0 and variance 1. Let n be the number of t_u in the segment. Let $r(u)$ be the rank of t_u. Then $z_u = Z^{-1}\{(r(u) - 0.5)/n\}$.

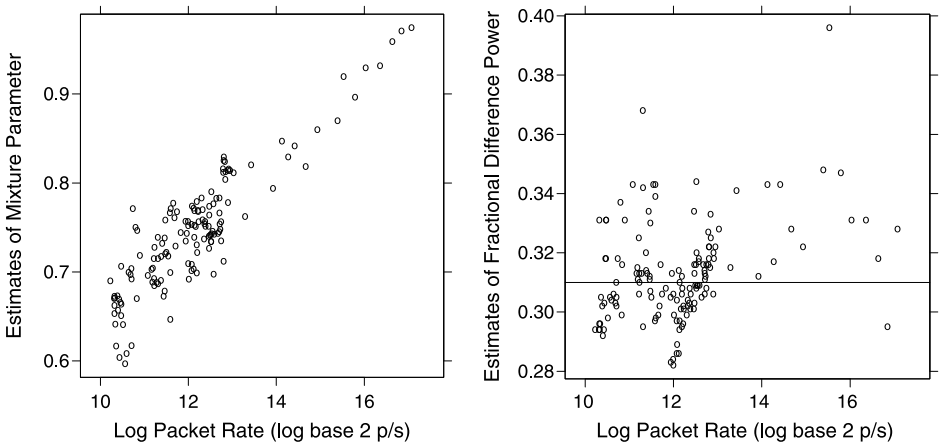

Fig. 6. Estimates of the GFSD mixture coefficient θ and fractional exponent d for the 144 Auckland traces. $\hat{\theta}_k$ (left panel) and $\hat{d}_k$ (right panel) are plotted against $\log_2(\hat{\alpha}_k)$, the log base 2 estimates of the packet rates. The horizontal line in the right panel shows the median, 0.31, of the $\hat{d}_k$.

The reason for using the empirical function, rather than a Weibull distribution function fitted to the t_u, was to have a portion of the model checking methods for z_u not depend on the validity of the specification of the marginal distribution of t_u. In Section 5, model checking of the marginal of t_u does not depend on the validity of the specifications for z_u. In a number of other sections, model checking depends jointly on specifications for the GFSD and the MFSD.

Estimation of the parameters d and θ of the GFSD are based on the periodogram of the observed z_u. In addition, the periodogram and the estimate of the power spectrum using the estimated parameters are a part of our model checking for validation of the GFSD. To enable subsequent study of the m-means and m-sums with exactly the same observations, we use just the first 2^b observations of each trace segment where b is the greatest integer in log base 2 of the number of z_u in the segment. For the 144 Auckland trace segments, the minimum value of b is 20 and the maximum is 26. The periodogram is computed at the Fourier frequencies $f_i = i/2^b$ for $i = 1, 2, 3, \ldots, 2^{b-1}$. These frequencies are divided into 2^{15} non-overlapping blocks of equal length, so each has 2^{b-16} values. For $j = 1, \ldots, 2^{15}$, let $\bar{f}_j$ be the mean of the frequencies in block j, and let $\bar{I}(\bar{f}_j)$ be the mean of the periodogram values in the block. Estimation and model checking proceed with $\bar{f}_j$ and $\bar{I}(\bar{f}_j)$.

Our parameter estimation method for each trace segment is designed to be robust to minor departures of the patterns in the $\bar{I}(\bar{f}_j)$ from the general form of the GFSD power spectrum. Some departures can adversely affect the estimation of d [62]. For example, minor low-frequency trends can remain because the detrending methods described in Section 4 cannot entirely remove the diurnal variation in the packet rate α.

Because $p_z(0.5) = \theta$, the estimate $\hat{\theta}$ of θ is taken to be the mean of the $\bar{I}(\bar{f}_j)$ for $f_j \geq 0.48$. This insures that the estimated power spectrum fits the pattern of the $\bar{I}(\bar{f}_j)$ for the highest frequencies. d is estimated from another frequency band: $0.01 \leq \bar{f}_j \leq 0.06$. $\hat{d}$ is the estimate arising from a nonlinear least squares fit of $10\log_{10}(p_z(\bar{f}_j))$ with $\theta = \hat{\theta}$ to $10\log_{10}(\bar{I}(\bar{f}_j))$ for f_j in the band. This is a variation of the method of [63] where the frequency band is $0 < f < a$ for a small a. The averaging of the periodogram before taking the log in the least-squares fitting falls in the category of an ATS method [64]; averaging before moving to a log scale results in efficient least-squares estimation.

6.3. The change in θ and d with α

For the 144 Auckland trace segments and $k = 1, \ldots, 144$, let $\hat{\theta}_k$ be the estimates of the mixture coefficient θ, let $\hat{d}_k$ be the estimates of the fractional exponent d, and let $\hat{\alpha}_k$ be the estimates of the packet rate α described in Section 3, the inverse of the sample mean of the interarrivals. Fig. 6 graphs $\hat{\theta}_k$ and $\hat{d}_k$ against $\log_2(\hat{\alpha}_k)$ where $\log_2$ is log base 2. The smallest values of $\hat{\theta}_k$ are close to 0.6; they tend to 1 as $\log_2(\hat{\alpha}_k)$ increases, which means that z_u tends to white noise. Except for two large outliers, values of $\hat{d}_k$ vary from about 0.28 to 0.35, a narrow range. The median, shown by the horizontal line, is 0.31. This suggests that d does not change with α so that a fixed d of 0.31 is reasonable in our mathematical study of traffic statistics based on the MFSD model. In fact, this is in keeping with other reports of values of d estimated in other ways [65].

Section 8 presents a derivation of θ as a function of α using the MFSD model. Equations are solved that yield numeric values, leading to a model $\theta(\alpha)$ for the dependence of θ on α. The theoretical model agrees with the empirical pattern in Fig. 6. This dependence of θ on α is a critical aspect of the statistical properties of the packet arrival process, so we switch to the notation $\theta(\alpha)$ in coming sections.

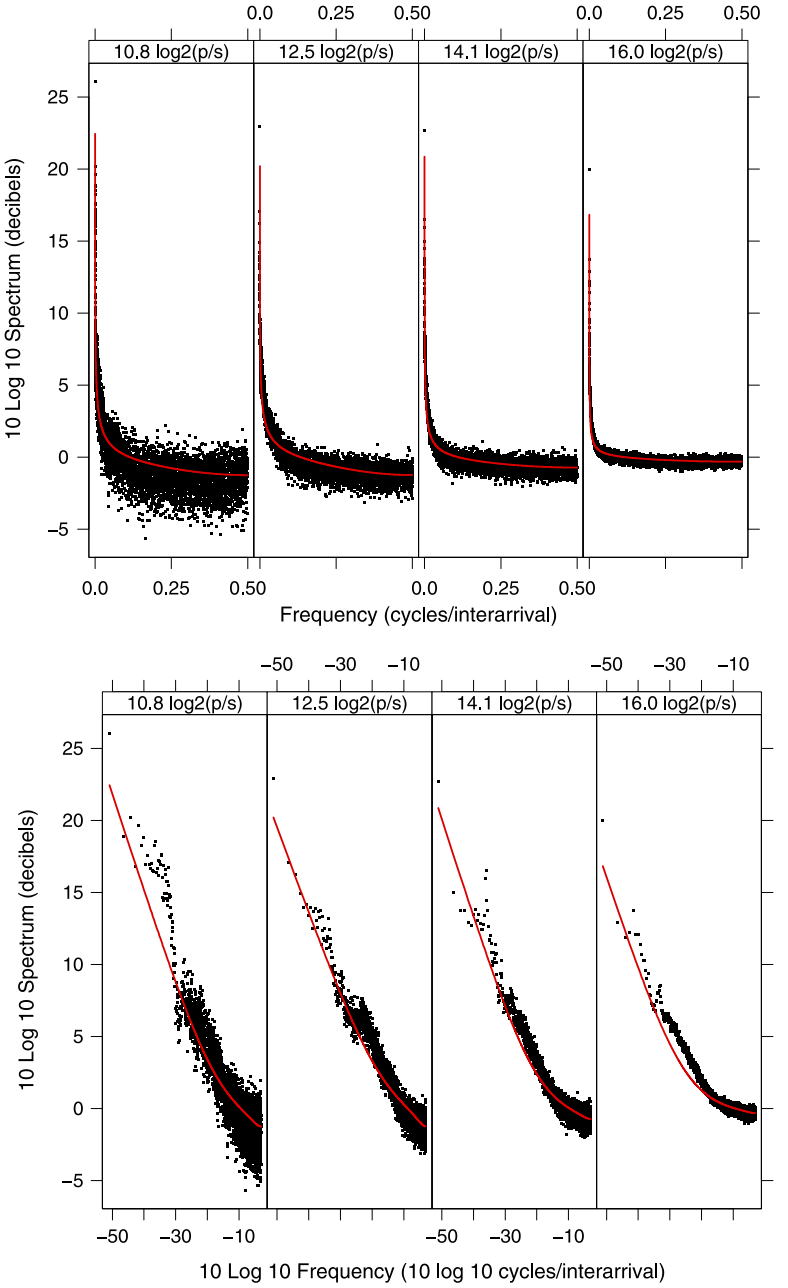

Fig. 7. $10\log_{10}(\bar{I}(\bar{f}_j))$ (•) and $10\log_{10}(\hat{p}_z(\bar{f}_j))$ (—) for 4 trace segments.

6.4. Model validation: properties of the power spectrum

The validity of the GFSD model – its ability to account for the statistical time-series properties of z_u – was explored by studying power spectra, one description of the properties. Other descriptions are studied in later sections.

We study validity by comparing $10\log_{10}(\bar{I}(\bar{f}_j))$, which is a (noisy) nonparametric estimate of the log power spectrum, and $10\log_{10}(\hat{p}_z(\bar{f}_j))$ the GFSD model estimate with $\theta(\alpha) = \hat{\theta}_k$ and $d = \hat{d}_k$. Fig. 7 show the results for the 4 Auckland traces described in Section 4.6. Each panel of the top row graphs $10\log_{10}(\bar{I}(\bar{f}_j))$ (•) and $10\log_{10}(\hat{p}_z(\bar{f}_j))$ (—) against $\bar{f}_j$. The bottom row is similar, except that values are graphed against $10\log_{10}(\bar{f}_j)$. Performance of the model fits is excellent; their departures from the nonparametric estimates are minor. This was the case for almost all of the packet trace segments of our validation study.

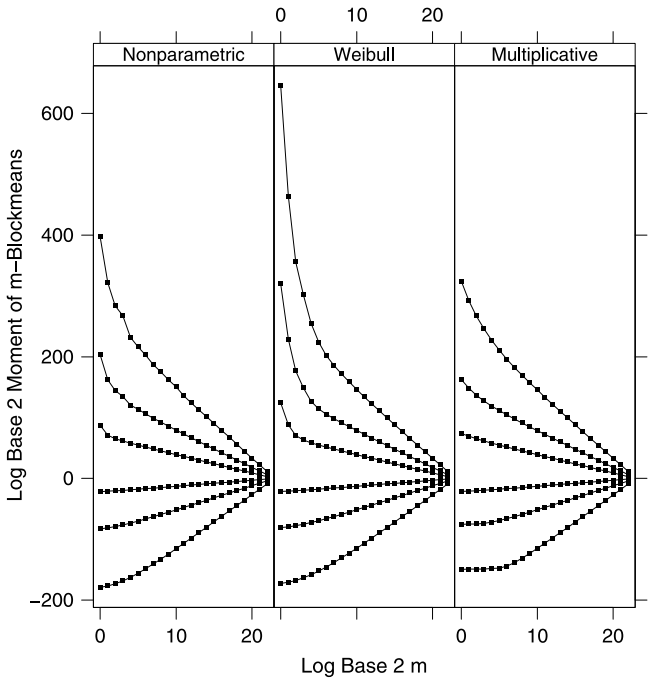

Fig. 8. Moment-statistic m-plot for one Auckland trace.

7. MFSD model validation: *m*-sum moment-statistics

This section addresses the nonlinearity of t_u through a moment-statistic study of the m-sum process $t_v^{(m)}$ as defined in Section 2. Normalized estimates of $\mathrm{E}\{(t_v^{(m)})^q\}$ are studied as a function of q and m, which is a time scaling study for each q. This multifractal analysis is a standard in the Internet traffic literature [27,58,32,52,26,42,49,57,8]. For each trace segment we compared nonparametric moment-statistics of the t_u with the theoretical moment-statistics from both the Weibull and multiplicative MFSD models fitted to the t_u. This provides an important look at nonlinear properties to aid validation, which very much justifies the analysis. However, it does not provide the foundational insights that arise from analyses in other sections.

For each trace segment, we estimated moments using the first $n = 2^b$ observations of t_u where b is the largest integer in the log base 2 of the number of interarrivals. This was the same data selection method used in Section 6. Let $t_\cdot = \sum_{u=1}^{n} t_u$. The nonparametric qth sample moment for the m-sum is

$$S_q^{(m)} = \sum_{v=1}^{2^{b-r}} \left(\frac{t_v^{(m)}}{t_\cdot} \right)^q . \tag{1}$$

Values of m were $m_r = 2^r$ for $r = 0, \ldots, b-1$, and the moments were $q = -10, -5, -2, 2, 5, 10$.

The Weibull MFSD t_u has 4 parameters. Two are for the Weibull marginal of t_u: the shape $\lambda(\alpha)$ and the packet rate α. Two are for the associated Gaussian image z_u: the fractional exponent d and the mixture coefficient $\theta(\alpha)$. The fitted Weibull MFSD for each trace segment is the MFSD with parameter values equal to the estimates described in Sections 5 and 6: $\hat{\alpha}$, $\hat{\lambda}$, $\hat{d}$, and $\hat{\theta}$. The multiplicative MFSD t_u has 4 parameters. Two are for the log normal marginal: the mean $\mu(\alpha)$ and variance $\tau^2(\alpha)$ of $\log(t_u)$. Their estimates $\hat{\mu}$ and $\hat{\tau}^2$ are the values for which the first and second moments of the log normal match the two moments of the Weibull with parameters $\hat{\alpha}$ and $\hat{\lambda}$. Two are for the associated Gaussian image z_u: d and $\theta(\alpha)$. Their estimates are also those of Section 6: $\hat{d}$ and $\hat{\theta}$, which are the same as those for the Weibull MFSD. We proceed with these estimates as if they were the true values.

We are unable to mathematically derive MFSD moment-statistics for the Weibull and multiplicative MFSD models, so simulation "derivations" were carried out for each trace segment. Each run for a trace segment consisted of generation of 2^b values of the interarrivals from the fitted model, which is the same number of values used for the nonparametric moment-statistics. Moment statistics for the run are computed using Eq. (1). Final values, $S_q^{(m)}$, are means across 100 runs.

Fig. 8 is a moment-statistic m-plot for one of the four trace segments described in Section 4, the one with packet rate $\hat{\alpha} = 2^{14.1}$ p/s. (The other 3 segments of the section are not shown in the interest of space.) For this trace segment, $b = 23$. Each panel plots $\log_2\{\hat{S}_q^{(m_r)}\}$ against $\log_2\{m_r\}$ for each value of q for one of three cases: nonparametric, Weibull MFSD, and

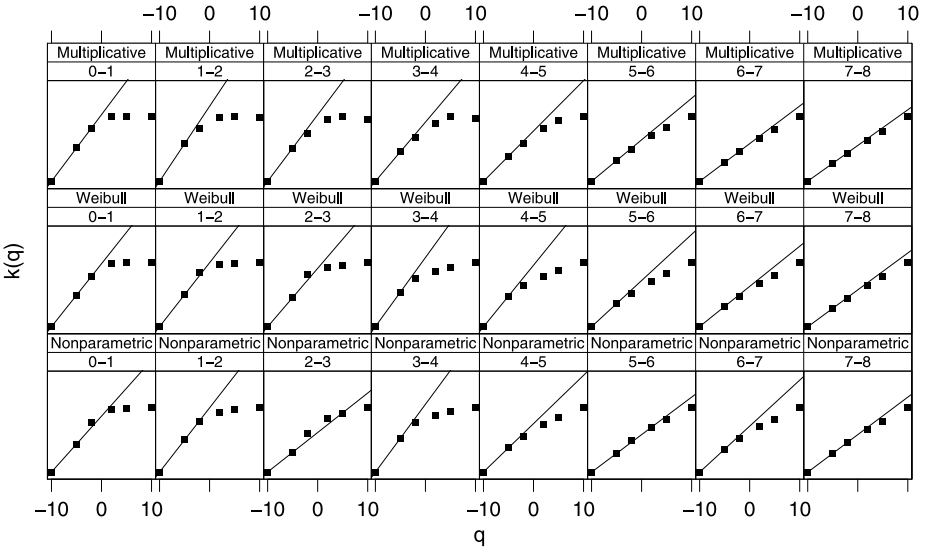

Fig. 9. Moment-statistic m-slopeplot for one Auckland trace.

multiplicative MFSD. Successive values for each q on a panel are connected by line segments. The resulting 6 curves, one per value of q, increase with q for fixed $\log_2\{m_r\}$.

For $r = 0, \ldots, b-2$, let

$$\kappa_{r,r+1}(q) = \frac{\log_2\{\hat{S}_q^{(m_{r+1})}\} - \log_2\{\hat{S}_q^{(m_r)}\}}{\log_2\{m_{r+1}\} - \log_2\{m_r\}} = \log_2\{\hat{S}_q^{(m_{r+1})}\} - \log_2\{\hat{S}_q^{(m_r)}\},$$

the slopes of the line segments shown in Fig. 8. Fig. 9 is a moment-statistic m-slopeplot. Each panel of the figure plots the following: $\kappa_{r,r+1}(q)$ against q for the case shown in the upper strip label of the panel and for the $[r, r+1]$ values shown in the lower strip label. This is done just for $r = 0$ to 7. The line on each panel goes through the first two points to help judge linearity.

The most important aspect of Figs. 8 and 9 is that the patterns for the nonparametric, Weibull MFSD, and multiplicative MFSD moment-statistics are similar, and agree with patterns of nonparametric moment-statistics seen in previous publications [30,32,52]. The two MFSD models are consistent with the data for this model checking method.

The nonlinearity of $\kappa_{r,r+1}(q)$ as a function of q shown in the panels of Fig. 9 is an indicator of multifractal behavior [32,31]. The patterns are concave for [0, 1] and tend toward linear as $[r, r+1]$ increases. Patterns for [8, 9] and above, not shown, are very close to linear.

8. Modeling the changing statistical properties with the packet rate

We have seen in Sections 5 and 6 that $\lambda(\alpha)$ and $\theta(\alpha)$ increase toward 1 with increasing α. The statistical properties of t_u change in a profound way with an increase in the traffic rate α because the expected number of active connections tends to increase with α. The parameters of the Weibull MFSD – the Weibull shape parameter $\lambda(\alpha)$ and the Gaussian image mixture parameter $\theta(\alpha)$ – reflect this change. Sections 5 and 6 show that each tends to 1 with α; the fractional exponent d does not change appreciably with α and is taken to be 0.31. The limit of $\lambda(\alpha)$ means that the marginal distribution of the Weibull MFSD t_u tends to exponential. The limit of $\theta(\alpha)$ means the z_u tend to Gaussian white noise. So the t_u tend to the interarrivals of a Poisson process. This is a critical property that has an immense impact on queueing delay, and therefore on the network engineering.

Using the Weibull MFSD model, we can also study theoretically the change in $\lambda(\alpha)$ and $\theta(\alpha)$ with α. We do this in two ways. The first is a derivation by simulation in which traffic is generated using the Weibull MFSD model. The second is a heuristic mathematical derivation whose detail is described in the Appendix. For both, we fix $d = 0.31$ and use initial values $\lambda_0 = 0.70$ and $\theta_0 = 0.55$ at the traffic rate $\alpha_0 = 2^{10.22}$ p/s, the smallest rate for the Auckland trace segments. The initial values were chosen so that the derivations provide the best fit to the estimates $\hat{\lambda}_k$ and $\hat{\theta}_k$ of Sections 5 and 6 as functions of the packet rate estimates $\hat{\alpha}_k$.

For the simulation, we generated 2 Weibull MFSD series, each with parameters λ_0 and θ_0 for the rate α_0. The two MFSD series were then numerically multiplexed, forming a series with rate $2^{11.22}$ p/s. $\lambda(\alpha)$ and $\theta(\alpha)$ were estimated using the methods employed in Sections 5 and 6 for the live and numerically multiplexed data, but with d fixed at 0.31. Then two series were generated at rate $2^{11.22}$ p/s using the estimated parameters, these two series were multiplexed, and then the

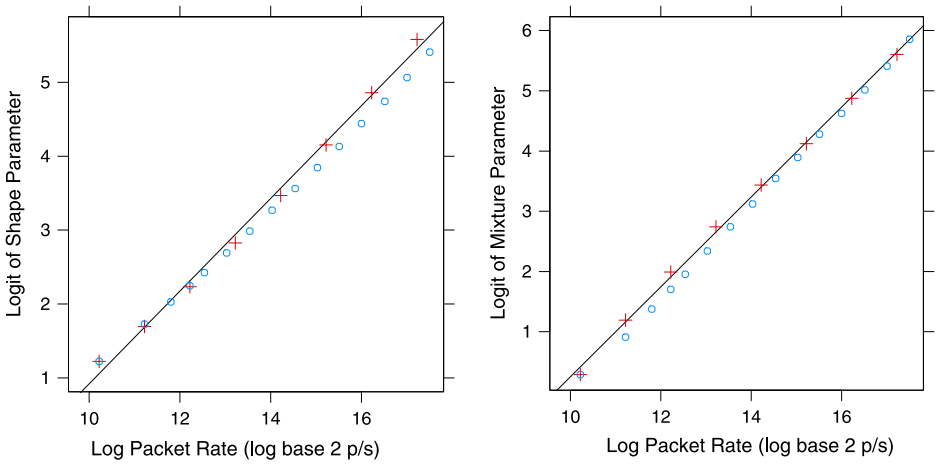

Fig. 10. Logit transformations of $\lambda(\alpha)$ (left) and $\theta(\alpha)$ (right), derived by simulation (+) and derived mathematically (∘), are plotted against $\log_2(\alpha)$.

parameters again estimated. This process continued up to rate $2^{17.22}$ p/s. The result is 8 values of $\lambda(\alpha)$ and $\theta(\alpha)$ including the initial values, and 8 associated values of α.

For the mathematical derivation, the process proceeds in a similar way, but with a different multiplexing method. r Weibull MFSD series with rates α_0 and parameters λ_0 and θ_0 were assumed to be multiplexed. Then values of $\lambda(\alpha)$ and $\theta(\alpha)$ for the multiplexed series were derived. The values of r were 2, 3, 4, 5, 7, 10, 14, 20, 28 39, 55, 78, 110, 155. The rates for the derived parameters range from $2^{10.22}$ p/s to $2^{17.5}$ p/s. The result is 15 values of $\lambda(\alpha)$ and $\theta(\alpha)$ including the initial values, and 15 associated values of α.

Fig. 10 graphs logit transformations of the derived values of the parameters, $\text{logit}_2(\lambda(\alpha)) = \log_2\{\lambda(\alpha)/(1-\lambda(\alpha))\}$ and $\text{logit}_2(\theta(\alpha)) = \log_2\{\theta(\alpha)/(1-\theta(\alpha))\}$, against $\log_2(\alpha)$. Each panel shows the simulation derived values (+) and the mathematically derived values (∘) for one parameter. For both parameters, the two derivations are very close. This provides a necessary validation of the mathematically derived values because they use certain assumptions that are not true for a Weibull MFSD model, but that are believed not to affect the results. The logit transformation results in a nearly linear dependence on $\log_2(\alpha)$. The line on each panel is the least squares fit to the simulated values. The equations are

$$\text{logit}_2(\lambda(\alpha)) = -5.36 + 0.63\log_2(\alpha)$$
$$\text{logit}_2(\theta(\alpha)) = -7.21 + 0.75\log_2(\alpha).$$

The equations on the scales of the parameters are

$$\lambda(\alpha) = \frac{2^{-5.36}\alpha^{0.63}}{1 + 2^{-5.36}\alpha^{0.63}}. \tag{2}$$

$$\theta(\alpha) = \frac{2^{-7.21}\alpha^{0.75}}{1 + 2^{-7.21}\alpha^{0.75}}. \tag{3}$$

For arrival process generation for simulation, all parameters are now accounted for. α is specified as part of the simulation, $\lambda(\alpha)$ and $\theta(\alpha)$ are computed by the above models, and $d = 0.31$.

Fig. 11 graphs $\hat{\lambda}_k$ and $\hat{\theta}_k$ against $\hat{\alpha}_k$ for the 144 Auckland traces. The curves are an evaluation of Eqs. (2) and (3) plotted against $\log_2(\alpha)$. There is substantial statistical variability in $\hat{\lambda}_k$, $\hat{\theta}_k$, and $\hat{\alpha}_k$; the reason is that measured segments must be short to insure stationarity. The curves do a reasonable job of fitting the patterns of the estimates considering this variability.

9. GFSD model validation and properties: autocorrelation

The four time series considered in the GFSD, which are defined in Section 3, are h_u, s_u, n_u, and z_u. This section presents formulas for their autocorrelations, which sets the stage for Sections 10 and 12, where approximations of the autocorrelations are derived for both z_u and t_u that provide important insight about statistical properties.

This section also describes results of the validation study carried out for the observed z_u of each trace segment by comparing the standard nonparametric estimates of the autocorrelation function with that of a GFSD model fitted to the z_u. This parallels the analysis of Section 6 that used the power spectrum for model checking. Mathematically, the autocorrelation function is equivalent to the power spectrum in that each is a Fourier transform of the other, but both are used for validation

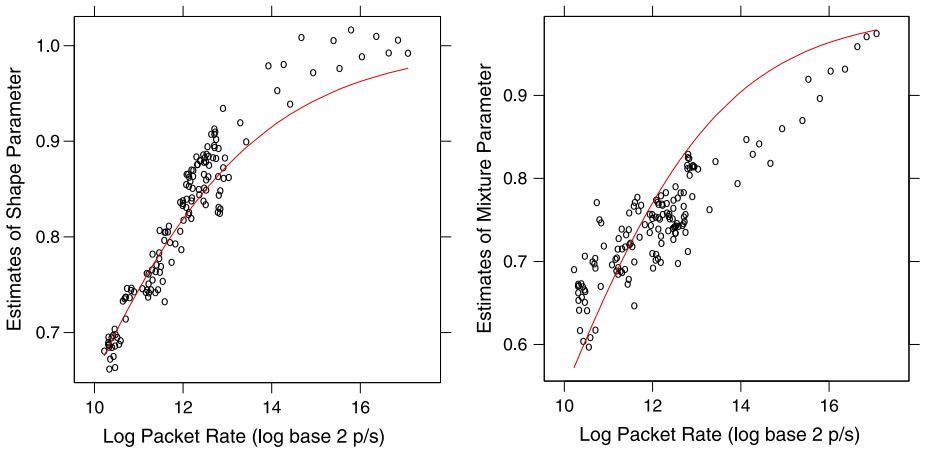

Fig. 11. $\hat{\lambda}_k$ (upper panel ∘) and $\hat{\theta}_k$ (lower panel ∘) are plotted against $\log_2(\hat{\alpha}_k)$. Derived equations $\lambda(\alpha)$ (upper panel —) and $\theta(\alpha)$ (lower panel —) are plotted against $\log_2(\alpha)$.

since a small consistent departure across lags or frequencies of one can translate to a large departure locally at certain frequencies or lags of the other.

Autocorrelation $\rho_w(k)$ $(k \geq 1)$ of a series w_u is computed from its power spectrum $p_w(f)$ $(0 < f \leq 0.5)$ as follows.

$$\rho_w(k) = \frac{\int_0^{0.5} \cos(2\pi fk) p_w(f) df}{\int_0^{0.5} p_w(f) df}.$$

The autocorrelation function for n_u for $k \geq 1$ is $\rho_n(k) = 0$. The 3 other series, which are long-range dependent, have formulas that are easily derived from their power spectra in Section 6, and from results of [19].

The autocorrelation at lag $k \geq 1$ for h_u, s_u and z_u, respectively, are

$$\rho_h(k) = \frac{\Gamma(1-d)}{\Gamma(d)} \frac{\Gamma(k+d)}{\Gamma(k-d+1)} = \prod_{i=1}^{k} \frac{(d+i-1)}{i-d} \tag{4}$$

$$\rho_s(k) = \rho_h(k) \frac{2(1-d)k^2 - (1-d)^2}{k^2 - (1-d)^2} \tag{5}$$

$$\rho_z(k) = (1 - \theta(\alpha))\rho_s(k). \tag{6}$$

$\rho_z(k) > 0$ and $\rho_z(k)$ goes to 0 with k to order k^{2d-1}, a signature property of the long-range dependence amply observed empirically in many studies.

The validation process for each trace segment begins, as described in Section 6, with a transformation to the observed Gaussian image z_u from the observed multifractal image t_u for $i = 1, \ldots, n$. The nonparametric estimate of autocorrelation at lag k for a segment is

$$n^{-1} \sum_{u=1}^{n-k} z_{u+k} z_u.$$

The GFSD model estimates of the autocorrelations are an evaluation of Eq. (6) with d and $\theta(\alpha)$ equal to the estimates $\hat{d}_k$ and $\hat{\theta}_k$ from Section 6.

Fig. 12 graphs the nonparametric estimates (•) and the fitted GFSD estimates (—) against $\sqrt{k}$ for the 4 Auckland trace segments described in Section 4. The square-root lag is used because it allows better assessment of the autocorrelations for small lags. The fitted GFSD estimates provide an excellent fit to the nonparametric estimates. This is the case for almost all trace segments.

10. GFSD model properties: near-self-similarity of h_u and s_u

The continuous time fGn process, which is the stationary increment of self-similar fBm, has an autocorrelation function approaching $d(2d+1)k^{2d-1}$ as $k \to \infty$, a simple mathematical form that allows much insight and tractable mathematics [25]. It has the attractive property that the log of the autocorrelation is linear in the log of the lag. Hosking proposed h_u as a discrete-time analog of continuous-time fGn [19].

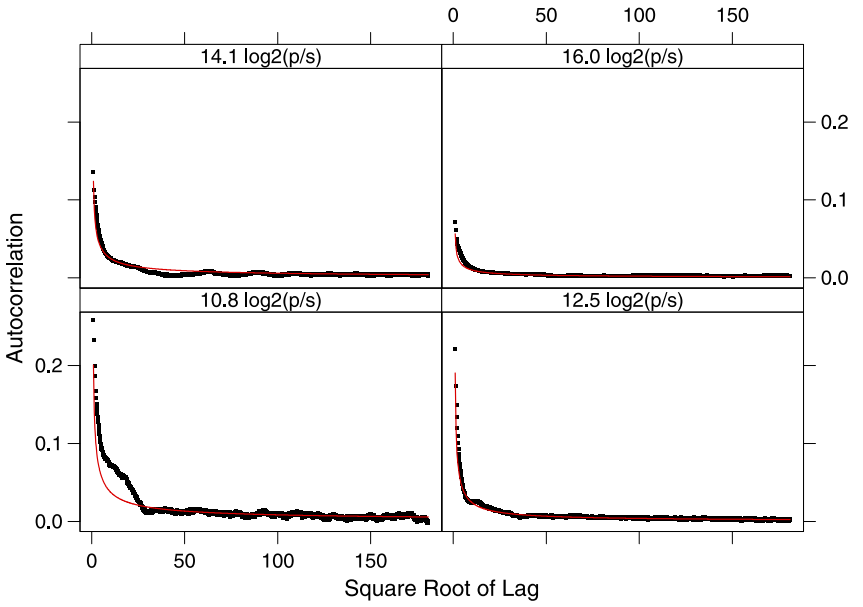

Fig. 12. Nonparametric estimates (•) and GFSD model estimates (—) of autocorrelation are plotted against square root lag for 4 Auckland trace segments.

The two independent components of the GFSD, s_u and n_u, are dependence extremes. One is white noise and the other is long-range dependent. The properties of n_u are easy to understand. The question is how we conceive of the long-range dependence of s_u as an aid to our foundational understanding of the properties of z_u. How close is s_u, a moving two-sum of h_u, to self-similar? This depends on how close h_u is to self-similar.

Following the notation of Section 2, the m-sums of s_u and h_u are $s_v^{(m)}$ and $h_v^{(m)}$. The m-scaled-sums are these m-sums divided by $m^{d+0.5}$,

$$\tilde{s}_v^{(m)} = m^{-d-0.5} s_v^{(m)},$$
$$\tilde{h}_v^{(m)} = m^{-d-0.5} h_v^{(m)}.$$

The m-scaled-sums of a self-similar process are identical processes. We study closeness to self-similarity by studying how close the m-scaled-sums of s_u and h_u are to being identical processes. Since h_u and s_u are Gaussian processes, closeness is determined by the closeness across values of m of the autocovariance function of each m-sum.

Throughout this section we denote the fractional exponent as d. However, as in previous sections, we take $d = 0.31$ when the values of expressions including d are enumerated to check approximations. Of course this means the results apply only to $d = 0.31$, sufficient for our purposes here.

10.1. Approximating the autocorrelation functions of h_u and s_u

A first question is whether the autocorrelation function of h_u,

$$\rho_h(k) = \frac{\Gamma(1-d)}{\Gamma(d)} \frac{\Gamma(k+d)}{\Gamma(k-d+1)},$$

is well approximated by a constant times k^{2d-1}. From Stirling's formula,

$$\lim_{k\to\infty} \frac{\Gamma(k+d)/\Gamma(k-d+1)}{k^{2d-1}} = 1,$$

so we approximate by

$$\ddot{\rho}_h(k) = \frac{\Gamma(1-d)}{\Gamma(d)} k^{2d-1}.$$

This is not the only possibility; for example, we could attempt an approximation in which the constant is chosen so that $\ddot{\rho}_h(1) = \rho_h(1)$.

The left panel of Fig. 13 plots $\log_2(\rho_h(k))$ (∘) and $\log_2(\ddot{\rho}_h(k))$ (—), both with $d = 0.31$, against $\log_2(k)$ for $k = 1, \ldots, 8$. $\ddot{\rho}_h(k)$ is an excellent approximation for these 8 lags. The largest discrepancy is at $k = 1$: $\rho_h(1)/\ddot{\rho}_h(1) = 1.012$, which is very small. For $k = 2, \ldots, 8$, the discrepancy decreases, and continues decreasing for $k > 8$.

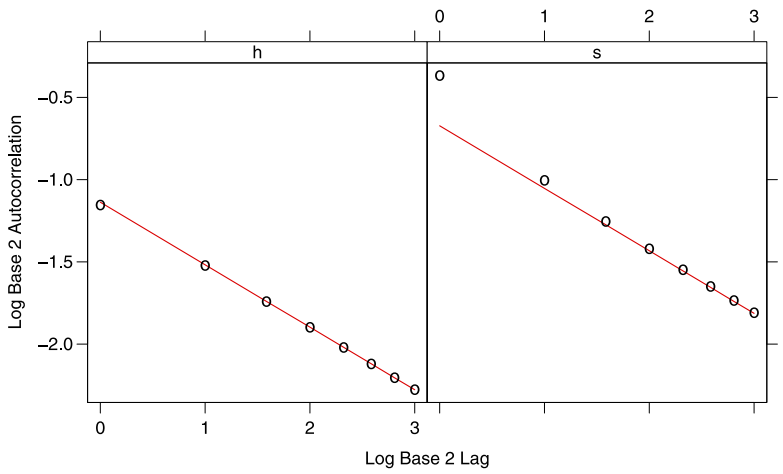

Fig. 13. Log base 2 autocorrelation with $d = 0.31$ vs. log base 2 lag. Left: $\rho_h(k)$ (∘), $\ddot{\rho}_h(k)$ (—). Right: $\rho_s(k)$ (∘), $\ddot{\rho}_s(k)$ (—).

For s_u, the approximation of $\rho_s(k)$ uses that for $\rho_h(k)$ and then takes another step. Since $s_u = h_u + h_{u-1}$,

$$\rho_s(k) = \frac{1-d}{2}\{\rho_h(k-1) + 2\rho_h(k) + \rho_h(k+1)\}.$$

The first step approximates by substituting $\ddot{\rho}_h(k)$ for $\rho_h(k)$,

$$\frac{(1-d)\Gamma(1-d)}{2\Gamma(d)}\{(1-1/k)^{2d-1} + 2 + (1+1/k)^{2d-1}\}k^{2d-1}.$$

The second approximation simplifies by replacing each of the two terms $(1-1/k)^{2d-1}$ and $(1+1/k)^{2d-1}$ by 1 to get

$$\ddot{\rho}_s(k) = \frac{2\Gamma(2-d)}{\Gamma(d)}k^{2d-1} = 2(1-d)\ddot{\rho}_h(k).$$

The approximation is exact in the limit,

$$\lim_{k\to\infty}\frac{\ddot{\rho}_s(k)}{\rho_s(k)} = 1.$$

In the right panel of Fig. 13, $\log_2(\rho_s(k))$ (∘) and $\log_2(\ddot{\rho}_s(k))$ (—), again with $d = 0.31$ are plotted against $\log_2(k)$ for $k = 1, \ldots, 8$. For $k = 1, \ldots, 4$, values of $\rho_s(k)/\ddot{\rho}_s(k)$ are 1.235, 1.034, 1.014, and 1.007. For $k > 4$, the differences are negligible. $\ddot{\rho}_s(k)$ as an approximation is reasonably close at lag $k = 1$ and excellent for $k \geq 2$.

10.2. Autocovariances of m-sums

This section examines how close the autocovariance functions of $\tilde{s}_v^{(m)}$ are to one another across m, and similarly for $\tilde{h}_v^{(m)}$. We first determine an m beyond which the autocovariance functions are nearly the same. The smaller this value of m, the closer h_u or s_u is to self-similar. Variances are treated first and then autocovariances at positive lags. The approximations $\ddot{\rho}_h(k)$ and $\ddot{\rho}_s(k)$ are used in formulas for the autocovariances in place of $\rho_h(k)$ and $\rho_s(k)$ for two reasons. First, it aids the assessment of self-similarity. Second, it provides simplification of formulas, both here and in later sections, which aids mathematical investigations.

10.2.1. Variances

For $m \geq 1$, the variance of $\tilde{s}_v^{(m)}$ is

$$V_{\tilde{s}_v^{(m)}} = \frac{1}{m^{2d+1}}\left(m + 2\sum_{k=1}^{m}(m-k)\rho_s(k)\right). \tag{7}$$

Replacing $\rho_s(k)$ with $\ddot{\rho}_s(k)$ in the right side of this equation, we have

$$\frac{1}{m^{2d+1}}\left(m + \frac{4\Gamma(2-d)}{\Gamma(d)}\sum_{k=1}^{m}(m-k)k^{2d-1}\right).$$

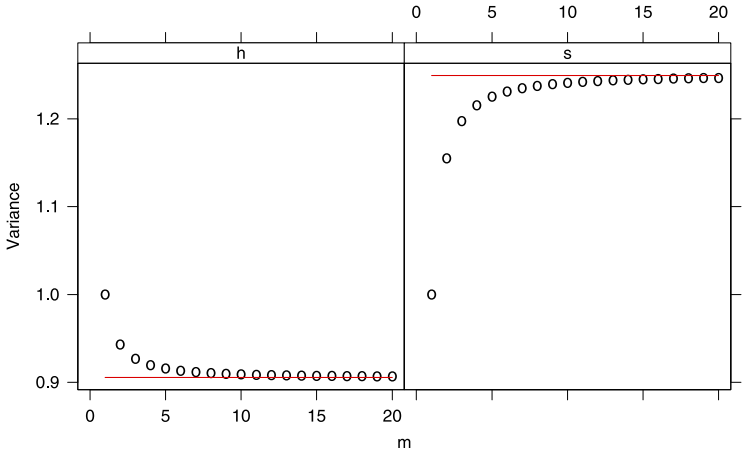

Fig. 14. Plotted against m for $d = 0.31$ are $V_{\tilde{h}_v^{(m)}}/\sigma_h^2$ (○) and $\ddot{V}_{\tilde{h}^{(m)}}/\sigma_h^2$ (—) in the left panel, and $V_{\tilde{s}_v^{(m)}}$ (○) and $\ddot{V}_{\tilde{s}_v^{(m)}}$ (—) in the right panel.

Replacing the summation by an integral in this last expression we get

$$\frac{2\Gamma(2-d)}{\Gamma(d+1)(2d+1)} + \left(1 - \frac{2\Gamma(2-d)/\Gamma(d)}{d}\right) m^{-2d} + \frac{4\Gamma(2-d)/\Gamma(d)}{2d+1} m^{-2d-1}.$$

For $d = 0.31$, the first term dominates quickly as m increases, so we drop the other two terms, which results in the final approximation

$$\ddot{V}_{\tilde{s}_v^{(m)}} = \frac{2\Gamma(2-d)}{\Gamma(d+1)(2d+1)}. \tag{8}$$

Following the same line of reasoning for h_u results in an approximation for the variance of $\tilde{h}_v^{(m)}$,

$$\ddot{V}_{\tilde{h}_v^{(m)}} = \frac{\sigma_h^2 \Gamma(1-d)}{\Gamma(d+1)(2d+1)} = \frac{\sigma_h^2 \ddot{V}_{\tilde{s}_v^{(m)}}}{2(1-d)}, \tag{9}$$

where $\sigma_h^2 = (1-d)/2$ is the variance of h_u.

The approximations of $V_{\tilde{s}_v^{(m)}}$ and $V_{\tilde{h}_v^{(m)}}$ in Eqs. (8) and (9) are constants for all m, the result for a self-similar process. So we judge the self-similarity of h_u and s_u by the closeness of the approximations. Fig. 14 graphs both the true (○) and the approximate (—) variances against m for $d = 0.31$. For $\tilde{h}_v^{(m)}/\sigma_h$, the true variances are very close to constant. The largest deviation of the approximation, at $m = 1$, is 8%; the remaining deviations fall quickly toward 0 as m increases. Thus h_u is very close to satisfying the variance property of self-similarity. For $\tilde{s}_v^{(m)}$, the deviation of the approximation is a moderate 25% at $m = 1$, but falls quickly to 8% at $m = 2$ and 4% at $m = 3$. Thus s_u is very close to satisfying the variance property of self-similarity for $m \geq 2$.

The results of Eqs. (8) and (9) are consistent with Theorem 2.2 of [66] on asymptotic behavior, derived using the power spectrum. The theorem states that if the autocorrelation function of a long-range dependent series with fractional exponent d converges to ck^{2d-1} as k gets large, then the m-scaled-sum variance converges to $c(d(2d+1))^{-1}$. In our case, $c = 2\Gamma(2-d)/\Gamma(d)$ for s_u and $c = \sigma_h^2 \Gamma(1-d)/\Gamma(d)$ for h_u.

10.2.2. Autocovariances for lags greater than 0

For $k \geq 1$, the autocovariance function of $\tilde{s}_v^{(m)}$ is

$$C_{\tilde{s}_v^{(m)}}(k) = \frac{1}{m^{2d+1}} \left(\sum_{j=-m}^{m} (m - |j|)\, \rho_s(km + j) \right).$$

We approximate using an approach similar to that for the variance. $\ddot{\rho}_s(k)$ approximates $\rho_s(k)$, and an integral from $-m$ to m approximates the resulting summation. This results in the final approximation of $C_{\tilde{s}_v^{(m)}}(k)$ for $k \geq 1$,

$$\ddot{C}_{\tilde{s}_v^{(m)}}(k) = \frac{2\Gamma(2-d)}{\Gamma(d)} k^{2d-1} = \ddot{\rho}_s(k). \tag{10}$$

The same line of reasoning is used for approximating $C_{\tilde{h}_v^{(m)}}(k)$,

$$\ddot{C}_{\tilde{h}_v^{(m)}}(k) = \sigma_h^2 \frac{\Gamma(1-d)}{\Gamma(d)} k^{2d-1} = \sigma_h^2 \ddot{\rho}_h(k). \tag{11}$$

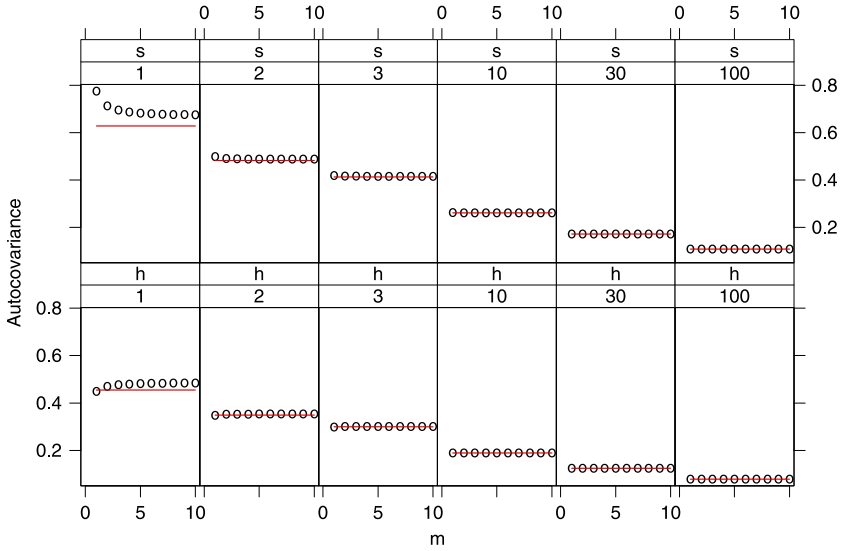

Fig. 15. Each panel plots true (∘) and approximate (—) autocovariances against m for the scaled sum shown in the top strip label of the panel (h for h_u, and s for s_u) and for the lag shown in the lower strip label.

The approximations of $C_{\tilde{s}_v^{(m)}}$ and $C_{\tilde{h}_v^{(m)}}$ in Eqs. (10) and (11) are constants for all m, the result for a self-similar process. So we judge the self-similarity of h_u and s_u by the closeness of the approximations. The top row of Fig. 15 graphs $C_{\tilde{s}_v^{(m)}}(k)$ (∘) and $\ddot{C}_{\tilde{s}_v^{(m)}}(k)$ (—) against m for $d = 0.31$ and 6 values of lag: $k = 1, 2, 3, 10, 30, 100$. The bottom row does the same for $C_{\tilde{h}_v^{(m)}}(k)/\sigma_h^2$ and $\ddot{C}_{\tilde{h}_v^{(m)}}(k)/\sigma_h^2$. For $\tilde{h}_v^{(m)}$, the approximation is excellent for all values of m and k shown, becomes more accurate as either m or k increase. The same is true of $\tilde{s}_v^{(m)}$ for $k \geq 2$; for $k = 1$ the approximation deviates somewhat for $m \leq 5$.

10.3. Near-self-similarity of h_u and s_u

The above results show h_u is very close to self-similar for $d = 0.31$. The term "discrete fGn" is certainly appropriate. s_u is not as close to self-similar, but is not far off; this would be expected since s_u is a moving 2-sum of h_u. However, s_u is certainly close enough to self-similar to allow it be thought of as such for the purpose of foundational, intuitive reasoning for the Gaussian image, z_u. z_u is a mixture of a near-self-similar series and a white noise series where the mixture parameter $\theta(\alpha)$ increases toward 1 as $\alpha \to \infty$. As we will see in later sections, we can also apply this intuition for the second moment properties of the multifractal image t_u.

11. GFSD model properties: rate–time analysis of variance and autocorrelation

This section treats the variances of the m-statistics of z_u, and how they change with increasing m, which is time scaling.

An m-statistic study is a time scaling analysis that has been widely used as a basis for understanding traffic statistical properties. The variance time plot is one of most used methods of such study. It consists of a display of the log m-mean variance against log m [10,9,67,32,56,44,45,68,69]. The common pattern of the log variance with increasing log m is decreasing, convex, and a slope tending to $2d - 1$.

Time scaling analysis is at its base, a frequency domain concept. The m-mean is a digital low-pass filter whose pass-band becomes more concentrated near zero as m increases. However, analyses have stayed in the time domain through a study of the variances and autocorrelation properties. This classical study has been almost exclusively empirical using nonparametric estimates of autocovariances, and with little guidance from theory. In this section, with the benefit of the structure of the GFSD model for z_u, we add much insight into the statistical properties of m-statistics. The model provides more fundamental drivers of the properties.

11.1. Variance

11.1.1. GFSD model validation

We use a variance time plot for validation of the GFSD model as done using the power spectrum in Section 6 and the autocorrelation function in Section 9. This follows the same initial procedures as in Section 6. The first 2^b values of the n observations of t_u are analyzed, where b is the greatest integer in $\log_2(n)$, and $\log_2$ is log base 2. The 2^b values are transformed

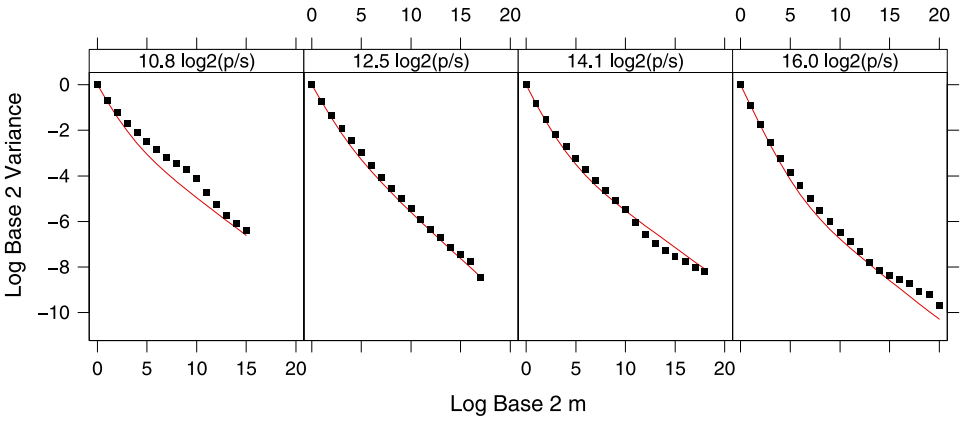

Fig. 16. Four variance rate–time plots for 4 Auckland trace segments. The log model variance, $\log_2 V_{\tilde{z}_v^{(m)}}$ (—), and the log nonparametric variance, $\log_2 \hat{V}_{\tilde{z}_v^{(m)}}$ (•), are graphed against $\log_2 m$.

to observed z_u using the empirical cumulative distribution function of the selected t_u. The values of m are taken to be $m_r = 2^r$ for $r = 0, \ldots, b-5$.

The classical nonparametric estimate of the variance for the variance time plot for each m_r is the sample variance, $\hat{V}_{\bar{z}_v^{(m_r)}}$, of the values of $\bar{z}_v^{(m_r)}$, v = 1, ..., $2^b/m_r$. Fig. 16 is a variance rate–time plot that compares the nonparametric and model values. On each panel, $\log_2\{V_{\bar{z}_v^{(m_r)}}\}$ (—) and $\log_2\{\hat{V}_{\bar{z}_v^{(m_r)}}\}$ (•) are plotted against $\log_2(m_r)$ for each one of the 4 Auckland trace segments described in Section 4. The packet rates of the segments are shown in the strip labels at the tops of the panels. The model variances are very close to the nonparametric variances for the 4 segments, as they are for almost all trace segments, providing another validation of the GFSD modeling of z_u.

11.1.2. Time and rate scaling analysis of variance

The driver of the effects of m and α on $V_{\bar{z}_v^{(m)}}$ is the changing relative contributions of the independent Gaussian components of z_u, $\sqrt{\theta}n_u$ and $\sqrt{1-\theta}s_u$, to $V_{\bar{z}_v^{(m)}}$ as m and α change. The two components are dependence extremes, white noise and near-self-similar, respectively.

$$\begin{aligned} \bar{z}_v^{(m)} &= \sqrt{(1-\theta(\alpha))}\bar{s}_v^{(m)} + \sqrt{\theta(\alpha)}\bar{n}_v^{(m)}, \\ V_{\bar{z}_v^{(m)}} &= (1-\theta(\alpha))V_{\bar{s}_v^{(m)}} + \theta(\alpha)V_{\bar{n}_v^{(m)}} \\ &= (1-\theta(\alpha))V_{\bar{s}_v^{(m)}} + \theta(\alpha)m^{-1}. \end{aligned} \tag{12}$$

We see that m and α influence the variance by changing the relative importance of the first and second terms. Fig. 16 is close to the classic plotting method. It becomes a rate–time plot simply because four traces are displayed with different rates. This was fine for validation purposes. While we can see certain properties as α and m change, other alterations in methodology increase substantially the effectiveness of the analysis.

The form of the GFSD makes clear that it is not sufficient to study just time scaling in isolation. Time scaling and the packet rate, α, interact. So we replace the classical method with a time–rate analysis method, taking both m and α into account. In addition, we change the classical m-statistic, the m-mean, replacing it by the m-scaled-sum, $\tilde{z}_v^{(m)} = m^{-d-0.5}z_v^{(m)}$, which is used in Section 10. This change, while it might seem minor, has a very big impact on the effectiveness of visual assessment of the patterns on the time–rate visual display because a self-similar process now has constant variance and autocorrelation across m for fixed rate α.

The m-scaled-sums of z_u, s_u, and n_u are related by

$$\tilde{z}_v^{(m)} = \sqrt{(1-\theta(\alpha))}\tilde{s}_v^{(m)} + \sqrt{\theta(\alpha)}\tilde{n}_v^{(m)},$$

so their variances are related by

$$V_{\tilde{z}_v^{(m)}} = (1-\theta(\alpha))V_{\tilde{s}_v^{(m)}} + \theta(\alpha)V_{\tilde{n}_v^{(m)}} = (1-\theta(\alpha))V_{\tilde{s}_v^{(m)}} + \theta(\alpha)m^{-1}.$$

The effects of m and α are determined by the changing relative contributions of $\theta(\alpha)m^{-1}$ and $(1-\theta(\alpha))V_{\tilde{s}_v^{(m)}}$ to $V_{\tilde{z}_v^{(m)}}$. In other words, the simple structure of the GFSD leads to a straightforward assessment.

We approximate $V_{\tilde{z}_v^{(m)}}$ using the approximation of $V_{\tilde{s}_v^{(m)}}$ given in Eq. (8), which results in

$$\ddot{V}_{\tilde{z}_v^{(m)}} = (1-\theta(\alpha))\frac{2\Gamma(2-d)}{(2d+1)\Gamma(d+1)} + \theta(\alpha)m^{-2d}. \tag{13}$$

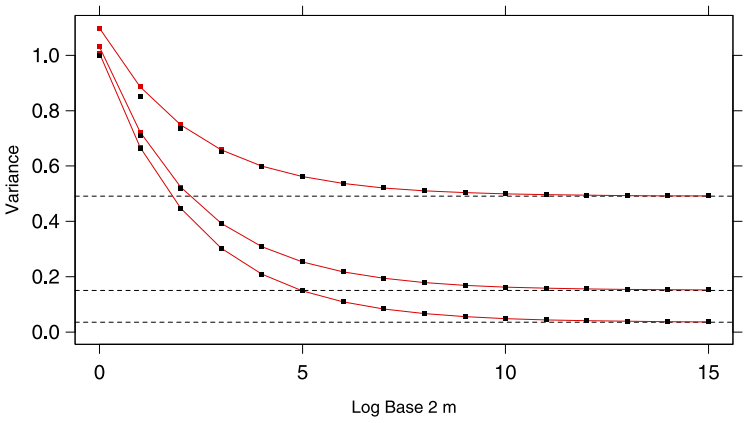

Fig. 17. Variance rate–time plot for 3 values of $\log_2(\alpha)$. The exact $V_{\tilde{z}_v^{(m)}}$ (•) and the approximate $\ddot{V}_{\tilde{z}_v^{(m)}}$ (—) are plotted against r.

For $d = 0.31$, the estimate of d described in Section 8, Eq. (13) becomes

$$\ddot{V}_{\tilde{z}_v^{(m)}} = 1.25(1 - \theta(\alpha)) + \theta(\alpha) m^{-0.62}. \tag{14}$$

We do not, as in the classical approach, take the log of $\ddot{V}_{\tilde{z}_v^{(m)}}$ since it would make the simple additive structure non-additive and thus harder to assess. Instead, assessment is aided by letting $m = 2^r$ and studying the variance as a function of r and α.

Overall, the accuracy of the approximation is excellent. For $m = 1$, $V_{\tilde{z}_v^{(m)}} = 1$, but $\ddot{V}_{\tilde{z}_v^{(m)}} = 1 + 0.25(1 - \theta(\alpha))$. At $\theta(\alpha) = 0.6$, close to the smallest value observed in the data, the error is 0.1, off by only 10%; as $\theta(\alpha)$ goes to 1 from 0.6, the error decreases to 0. As r increases, the error for fixed $\theta(\alpha)$ monotonically decreases with r. The approximation is, in fact, the asymptotic value

$$\lim_{m \to \infty} \frac{V_{\tilde{z}_v^{(m)}}}{\ddot{V}_{\tilde{z}_v^{(m)}}} = 1.$$

The high accuracy is illustrated in Fig. 17. $V_{\tilde{z}_v^{(m)}}$ (•) and $\ddot{V}_{\tilde{z}_v^{(m)}}$ (—) are plotted against 16 values of r from 0 to 15 for 3 values of $\log_2(\alpha)$: 10.5, 13.5, and 16.5 $\log_2$(packets/second). 10.5 and 16.5 are close to the minimum and maximum log rates of the Auckland trace segments. The horizontal lines show the asymptotic values. The accuracy is certainly good enough to use $\ddot{V}_{\tilde{z}_v^{(m)}}$ for study of general properties of time and rate scaling.

So far, the discussion has shown the rate effect through $\theta(\alpha)$, which increases to 1 with increasing α. However, Section 8 derives a logistic model for $\theta(\alpha)$. We substitute the expression for $\theta(\alpha)$, Eq. (3), into Eq. (14) to relate $\ddot{V}_{\tilde{z}_v^{(m)}}$ directly to α. We could study the relative contributions of the two additive terms in Eq. (14) through the ratio of the second term to the sum, which is the fraction of variance due to the white noise component. Instead, because of the logistic model for $\theta(\alpha)$, we study the "log odds",

$$\zeta(\alpha, m) = \log_2 \left\{ \frac{\theta(\alpha) m^{-0.62}}{1.25(1 - \theta(\alpha))} \right\}.$$

Substituting for $\theta(\alpha)$ yields a very simple equation,

$$\zeta(\alpha, m) = -7.53 + 0.75 \log_2(\alpha) - 0.62 \log_2(m). \tag{15}$$

Eq. (15) gives much insight into the time–rate scaling of z_u. If $\log_2(\alpha)$ increases by 1 for fixed m, which means the packet rate doubles, then $\zeta(\alpha, m)$ increases by 0.75 and the variance ratio by a factor of $2^{0.75} = 1.68$. For a given $m = m_0$, let $m_0' > m_0$ be the value of m that makes up for the increase in the variance of the white noise term and brings the ratio back to the original value, that is $\zeta(\alpha, m_0') = \zeta(2\alpha, m_0)$. This achieved by $m_0' = 2.31 m_0$, which means a time scale increase must "over-compensate" to equalize the effect of rate scale increase. This is a different instance of the concept of "multiplexing gains" from rate increase discussed in the literature [70,11,12,71,17,44], but just qualitatively, without the form and quantification of Eq. (15).

Eq. (15) also shows quantitatively the progression of $\tilde{z}_v^{(m)}$ to white noise as α increases for any fixed value of m, and the progression of $\tilde{z}_v^{(m)}$ to near-self-similar as m increases for any fixed value of α. This is illustrated in Fig. 18. $\zeta(\alpha, m)$ is plotted against $\log_2(m)$ for each of four values of $\log_2(\alpha)$: 10.5, 13.5, 16.5, and 19.5 log base 2 p/s. The points for each value of $\log_2(\alpha)$ lie on a line; the intercepts of the lines increase with $\log_2(\alpha)$. The lowest rate is close to the value of 1000 p/s below which the MFSD model is not valid, as discussed in Section 1. To help the interpretation of the quantitative information on the

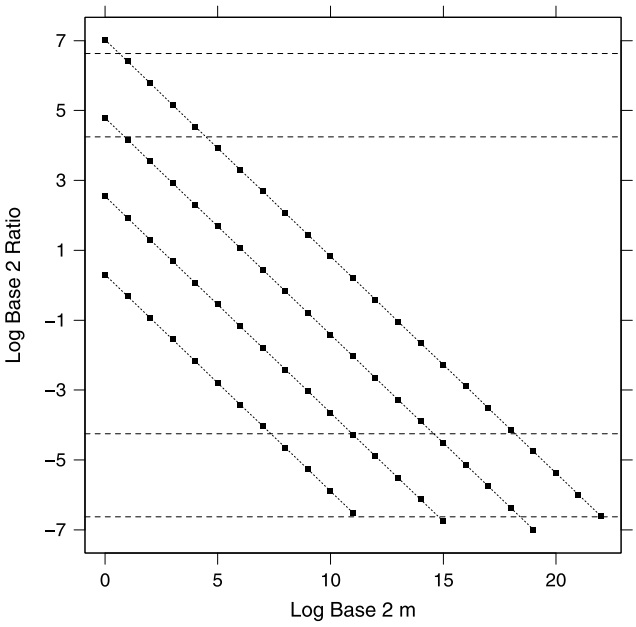

Fig. 18. $\zeta(\alpha, m)$ is plotted against $\log_2(m)$ for each of four values of $\log_2(\alpha)$: 10.5, 13.5, 16.5, and 19.5 log base 2 p/s. The points for each value of $\log_2(\alpha)$ lie on a line; the intercepts of the lines increase with $\log_2(\alpha)$. The horizontal dashed lines show values of $\zeta(\alpha, m)$ for which the percentages of the variances of the white noise term are, from top to bottom, 99%, 95%, 5%, and 1%.

display, the horizontal dashed lines show values of $\zeta(\alpha, m)$ for which the percentages of the variances of the white noise term are, from top to bottom, 99%, 95%, 5%, and 1%.

For $\log_2(m) = 0$ and $\log_2(\alpha) = 19.5$, $\sqrt{\theta(\alpha)}\tilde{n}_v^{(0)} = \sqrt{\theta(\alpha)}n_v$ accounts for a little more than 99% of the variance of $\tilde{z}_v^{(0)} = z_v$, so z_v is very close to white noise. It takes an m of 2^{22} to get to a point where $\sqrt{(1-\theta(\alpha))}\tilde{s}_v^{(m)}$ is about 99% of the variance of $\tilde{z}_v^{(m)}$, making $\tilde{z}_v^{(m)}$ very close to self-similar. By contrast, for $\log_2(m) = 0$ and $\log_2(\alpha) = 10.5$, $\sqrt{\theta(\alpha)}n_v$ accounts for just 56% of the variance of z_v. In this case, an m of only 2^{11} is sufficient to get to a point where $\sqrt{(1-\theta(\alpha))}\tilde{s}_v^{(m)}$ is about 99% of the variance of $\tilde{z}_v^{(m)}$, because the initial z_u is closer to self-similar than when the rate α is larger.

11.2. Autocorrelation

In the past, little time scaling analysis of autocorrelation has been carried out directly, compared with that for variance. In one instance, a very informative work, [35] observed empirically that the autocorrelations of m-means of arrival counts in fixed intervals increase with increasing m. Next we carry out rate–time analysis of the autocorrelation function, $\rho_{\tilde{z}_v^{(m)}}(k)$. The analysis proceeds in a manner very similar to that of the above analysis of the variance, so we proceed quickly to the approximate formulas that provide the insight.

First,

$$\rho_{\tilde{z}_v^{(m)}}(k) = \frac{C_{\tilde{z}_v^{(m)}}(k)}{V_{\tilde{z}_v^{(m)}}} = \frac{(1-\theta(\alpha))C_{\tilde{s}_v^{(m)}}(k)}{V_{\tilde{z}_v^{(m)}}}.$$

Using approximations in Eqs. (10) and (13) leads to the $\rho_{\tilde{z}_v^{(m)}}(k)$ approximation

$$\ddot{\rho}_{\tilde{z}_v^{(m)}}(k) = \frac{(1-\theta(\alpha))2\Gamma(2-d)/\Gamma(d)}{\theta(\alpha)m^{-2d} + (1-\theta(\alpha))2\Gamma(2-d)/[\Gamma(d+1)(2d+1)]}k^{2d-1}. \tag{16}$$

Taking $d = 0.31$, the overall estimate of d from our trace segments, and using Eqs. (14) and (15), results in

$$\ddot{\rho}_{\tilde{z}_v^{(m)}}(k) = \frac{0.50}{1+2^{\zeta(\alpha,m)}}k^{2d-1} = \frac{0.50}{1+2^{-7.53+0.75\log_2(\alpha)-0.62\log_2(m)}}k^{2d-1}. \tag{17}$$

Eq. (17) shows the driver for the rate and time properties, and our understanding of them, is $\zeta(\alpha, m)$. This makes conclusions for the autocorrelations remarkably similar to those for the variance. The equation shows quantitatively the progression of $\rho_{\tilde{z}_v^{(m)}}(k)$ to the zero autocorrelations of white noise as the rate α increases for any fixed value of m, and the progression to the near-self-similar autocorrelations $0.5k^{2d-1}$ as m increases for any fixed value of α. Note that for m, this is

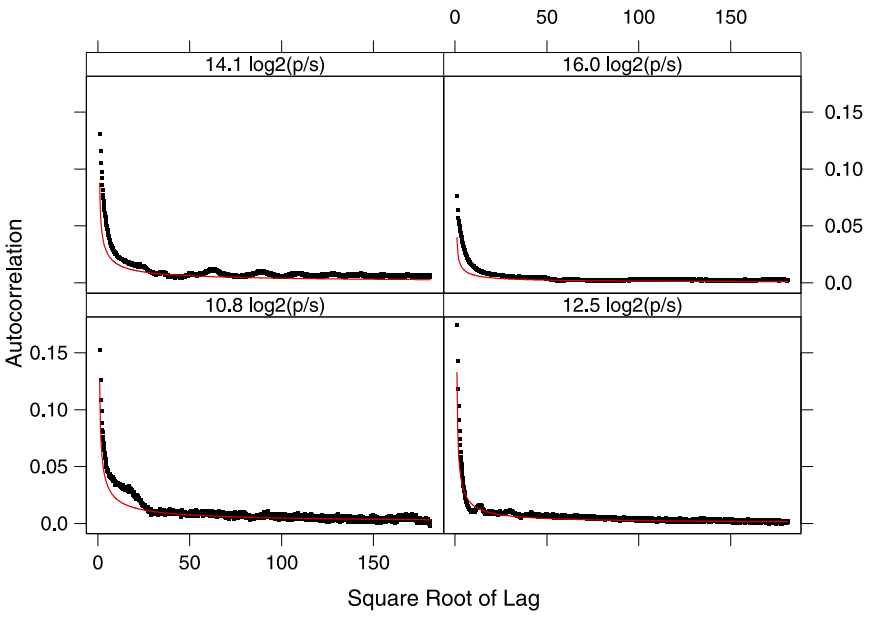

Fig. 19. Nonparametric estimates of autocorrelation (•) and estimates from fitted multiplicative MFSD models (—).

in agreement with the above results of [35]. As with the variance, the greater the value of α, the greater m must be to bring the autocorrelation function close to that of near-self-similar. In addition, the increase in m must "overcompensate", a part of the concept of multiplexing gains.

12. MFSD autocorrelation

In previous sections we have studied extensively the properties of z_u, the Gaussian image of t_u, as a prelude to understanding t_u. In this and coming sections we address the MFSD model for t_u. In doing this, however, we use the multiplicative MFSD in place of the Weibull MFSD. The increased tractability leads to important results, and our validation process shows the approximation is excellent.

The log normal parameters are the mean and variance, $\mu(\alpha)$ and $\tau^2(\alpha)$, of $\log(t_u)$, the notation reflecting the fact that the mean and variance change with α. The chosen log normal is that whose first two moments match those of the Weibull. The resulting values are

$$\tau^2(\alpha) = \log(\Gamma(1 + 2/\lambda(\alpha))) - 2\log(\Gamma(1 + 1/\lambda(\alpha))) \tag{18}$$

$$\mu(\alpha) = -\tau^2(\alpha)/2 - \log(\alpha) \tag{19}$$

$$\sigma_t^2(\alpha) = \frac{\Gamma\left(1 + \frac{2}{\lambda(\alpha)}\right)}{\alpha^2\Gamma^2\left(1 + \frac{1}{\lambda(\alpha)}\right)} - \frac{1}{\alpha^2}. \tag{20}$$

As $\alpha \to \infty$, $\lambda(\alpha) \to 1$, so $\mu(\alpha) \to -\infty$ and $\tau^2(\alpha) \to \log(2) = 0.693$. The minimum value of $\lambda(\alpha)$ observed in our data is about 0.6 for which $\tau^2(\alpha) = \log(4.09) = 1.41$. $\tau(\alpha)^2$ decreases monotonically as α increases.

In this section we consider the autocorrelation function of t_u and its approximation. Because $t_u t_{u-k}$ is also log normal, $L(2\mu(\alpha), 2\tau(\alpha)^2(1 + \rho_z(k)))$, the autocorrelations have a simple formula

$$\rho_t(k) = \frac{e^{\tau^2(\alpha)\rho_z(k)} - 1}{e^{\tau^2(\alpha)} - 1}. \tag{21}$$

There is also strong empirical validation. The log normal MFSD $\rho_t(k)$ with $d = 0.31$ and $\mu(\alpha)$ and $\tau(\alpha)^2$ estimated from each of the trace segments, almost always provides a good fit to the standard nonparametric estimate of autocorrelation for the segment. This is illustrated in Fig. 19. The nonparametric estimates (•) and the estimates from the log normal MFSD $\rho_t(k)$ (—) are plotted against square root of lag for the 4 Auckland trace segments used in previous sections. Taking statistical variability into account, which includes correlation in the estimates across the lags, the fit is good.

The autocorrelations of z_u, $\rho_z(k)$, while highly persistent, are not large; almost all values are below 0.25, and beyond the first few lags, are below 0.10. The values of $\tau^2(\alpha)$ for α above about 1000 packets/second, range from about 1.41 to 0.693

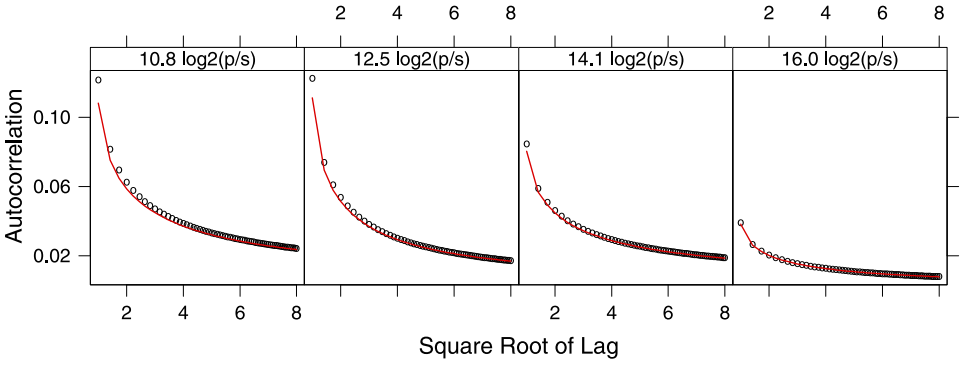

Fig. 20. Power series approximation (—) of multiplicative MFSD autocorrelations (∘).

as α increases. The resulting values of $\tau(\alpha)^2\rho_z(k)$ are small enough that the term $\exp\{\tau(\alpha)^2\rho_z(k)\}$ in Eq. (21) is very well approximated by a first order power series approximation, $1 + \tau^2(\alpha)\rho_z(k)$. Let

$$\phi(\alpha) = \tau^2(\alpha)/(e^{\tau^2(\alpha)} - 1). \tag{22}$$

The above first order approximation results in an even simpler autocorrelation approximation of $\rho_t(k)$ for lag $k \geq 1$

$$\ddot{\rho}_t(k) = \phi(\alpha)\rho_z(k) = \phi(\alpha)(1 - \theta(\alpha))\rho_s(k) = (1 - \ddot{\theta}(\alpha))\rho_s(k) \tag{23}$$

where

$$\ddot{\theta}(\alpha) = 1 - \phi(\alpha)(1 - \theta(\alpha)). \tag{24}$$

For $d = 0.31$, Fig. 20 graphs the values of the log normal MFSD $\rho_t(k)$ from Fig. 19 for lags 1 to 64 (∘) and the values of $\ddot{\rho}_t(k)$ (—). Except for lag 1, which has a minor departure, the approximation is very close. Figs. 19 and 20 provide an important validation for the autocorrelation structure of the MFSD model.

The astonishing result is that the autocorrelations of the multiplicative MFSD are very well approximated by those of a GFSD, not the Gaussian image of the MFSD with the value $\theta(\alpha)$, but rather with $\ddot{\theta}(\alpha)$. We have

$$\ddot{\theta}(\alpha) - \theta(\alpha) = (1 - \phi(\alpha))(1 - \theta(\alpha)),$$

so $\ddot{\theta}(\alpha)$ is larger, which means the approximating GFSD has a larger white noise component than the Gaussian image. For α increasing, starting at about 1000 packets/second, the lower limit of validity of the MFSD and GFSD models, we have $\theta(\alpha)$ goes from 0.60 to 1, $\tau^2(\alpha)$ go from 1.41 to 0.693, $\phi(\alpha)$ goes from 0.456 to 0.693, $\theta'(\alpha)$ goes from 0.818 to 1, and $\theta'(\alpha) - \theta(\alpha)$ goes from 0.218 to 0. What this means is that the second moment results for z_u in Sections 9–11 hold for t_u as well.

13. Fast traffic generation for network traffic engineering simulation studies

Simulation studies of network designs are critical to network traffic engineering. This has been discussed in Section 1. We need a fast traffic generation method for simulations, especially for those with large traffic rates α. We have developed a fast generation method. The method uses the multiplicative MFSD. Next we describe the method and the heuristic reasoning behind it.

Consider the $t_v^{(m)}$ of a multiplicative MFSD. So long as m is not too large and α is large enough, arrivals within $t_v^{(m)}$ are approximately Poisson. The reason is that for the generating GFSD, the component s_u does not contribute as much to the variation in $t_v^{(m)}$ if m is not too big because the $(1 - \theta(\alpha))s_u$ component has most of its power at low frequencies, while the $\theta(\alpha)n_u$ component is white noise. This means that, conditional on $t_v^{(m)}$, the arrival process within the interval is approximately uniform over the interval.

So we generate an MFSD series that serves as $t_v^{(m)}$, and then simply generate $m - 1$ random uniforms for each interval to get the interarrivals. This makes traffic generation faster by a factor of about m. $m = 100$ is a big saving, but yet m qualifies for all but small α. In fact, the smallest m that works increases with increasing α because the influence of $\theta(\alpha)n_u$ increases and the contribution of $(1 - \theta(\alpha))s_u$ decreases. A full quantitative study of m and α for this matter is beyond the scope of the paper.

The first step in a generation is to choose α. Next we generate $t_v^{(m)}$ as a multiplicative MFSD, $\exp\{\tau^*(\alpha)z_v^* + \mu^*(\alpha)\}$, where z_v^* is a GFSD with parameters $\theta^*(\alpha)$ and d. $d = 0.31$ based on results from Section 6. Next we describe how we get the parameters $\tau^*(\alpha)$, $\mu^*(\alpha)$, and $\theta^*(\alpha)$.

We need $\theta(\alpha)$ and $\lambda(\alpha)$, calculated from the logistic models described in Section 11. We need $\mu(\alpha)$ and $\tau^2(\alpha)$, calculated from $\lambda(\alpha)$ and α using Eqs. (18) and (19). We then compute $\ddot{\theta}(\alpha)$ from Eqs. (22) and (24), which is larger than $\theta(\alpha)$. In addition, we compute the variance of t_u, $\sigma_t^2(\alpha)$ from Eq. (20).

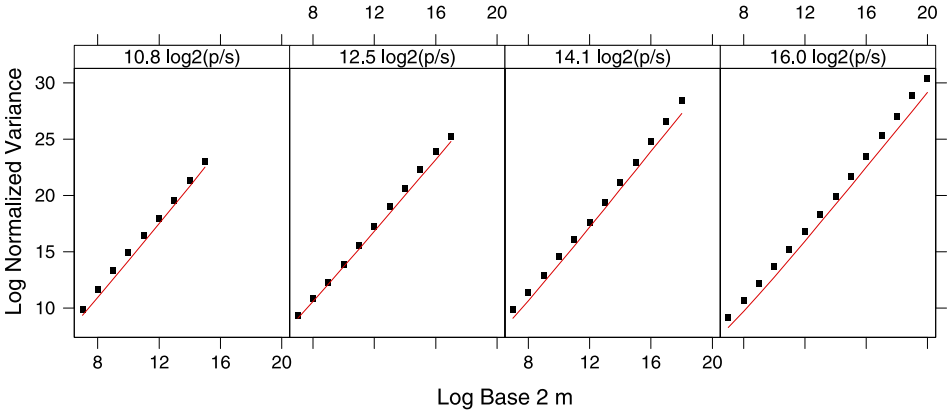

Fig. 21. The log base 2 of normalized nonparametric estimates of variances (•), and the log base 2 of normalized approximate variances (—) for $t_v^{(m)}$, are plotted against $\log_2(m)$ for the 4 Auckland trace segments.

We derive an approximation of the mean, variance and autocovariance of $t_v^{(m)}$. The derivation brings together results of Sections 10–12. One derivation is for the variance $V_{t_v^{(m)}}$, and the other for the autocovariance $C_{t_v^{(m)}}(k)$ for $k \geq 1$. The success depends on the result of Section 12 that the autocorrelations of t_u are very well approximated by those of a GFSD (Eq. (23)). This allows us to apply the results in Section 10 for $z_v^{(m)}$ to obtain results for $t_v^{(m)}$. It was convenient in that section to work with m-scaled-sums, $\tilde{z}_v^{(m)}$, but results hold also for $z_v^{(m)}$ with obvious changes in formulas to take account of the scaling factor $m^{d+0.5}$ and the variance $\sigma_t^2(\alpha)$ not being equal to 1.

To get the approximate variance $\ddot{V}_{t_v^{(m)}}$ and approximate autocovariance $\ddot{C}_{t_v^{(m)}}(k)$, we simply apply the results of the approximations in Section 10 for a GFSD model (Eqs. (8) and (10)) with d and mixture parameter $\ddot{\theta}(\alpha)$, and multiply the results by the factors m^{2d+1} and $\sigma_t^2(\alpha)$. We then have the approximate variance and autocovariance

$$\ddot{V}_{t_v^{(m)}} = \sigma_t^2(\alpha)\left(\ddot{\theta}(\alpha)m + (1-\ddot{\theta}(\alpha))\frac{2\Gamma(2-d)}{(2d+1)\Gamma(d+1)}m^{2d+1}\right) \tag{25}$$

$$\ddot{C}_{t_v^{(m)}}(k) = \sigma_t^2(\alpha)(1-\ddot{\theta}(\alpha))\frac{2\Gamma(2-d)}{\Gamma(d)}m^{2d+1}k^{2d-1}. \tag{26}$$

Furthermore it is straightforward to obtain the exact mean of $t_v^{(m)}$,

$$E\left(t_v^{(m)}\right) = \frac{m}{\alpha}. \tag{27}$$

We obtain $\theta^*(\alpha)$, $\tau^*(\alpha)$, and $\mu^*(\alpha)$ by matching the mean, variance, and autocovariance of $\exp\{\tau^*(\alpha)z_v^* + \mu^*(\alpha)\}$ using Eqs. (25), (26), and (27).

$$\tau^{*2}(\alpha) = \log\left(1 + \frac{\ddot{V}_{t_v^{(m)}}}{(m/\alpha)^2}\right) \tag{28}$$

$$\mu^*(\alpha) = \log\left(\frac{m}{\alpha}\right) + \frac{1}{2}\log\left(1 + \frac{\ddot{V}_{t_v^{(m)}}}{(m/\alpha)^2}\right) \tag{29}$$

$$\theta^*(\alpha) = 1 - \frac{d(2d+1)\Gamma(d)}{2\Gamma(2-d)}\frac{\exp(\tau^{*2}(\alpha)) - 1}{\tau^{*2}(\alpha)}. \tag{30}$$

The results in Eqs. (25) and (26) are validated using live trace segments. For validation, we compared the approximate variances and autocovariances from Eqs. (25) and (26) to standard nonparametric estimates of the variances and autocovariances. We are interested in reasonable values of m, since these m values achieve a significant reduction in the number of values that describe the traffic.

We found that Eqs. (25) and (26) did an excellent job of fitting the nonparametric estimates. Figs. 21 and 22 illustrate the results for the 4 Auckland traces described in Section 4. The standard nonparametric sample variances and autocovariances of $t_v^{(m)}$ (•) are normalized by dividing by the sample variance of the t_u. The approximate variances and autocovariances (—) are normalized by dividing by the variance $\sigma_t^2(\alpha)$. As we have done in all validation studies discussed in the paper, the base model parameters are taken to be the estimates $\hat{\lambda}$, $\hat{\theta}$, and $\hat{d}$ described in Sections 5 and 6.

In Fig. 21, the log base 2 of the normalized approximate variances and nonparametric variances are plotted against $\log_2(m)$. The approximation is excellent considering that variance estimates from long-range dependent time series have

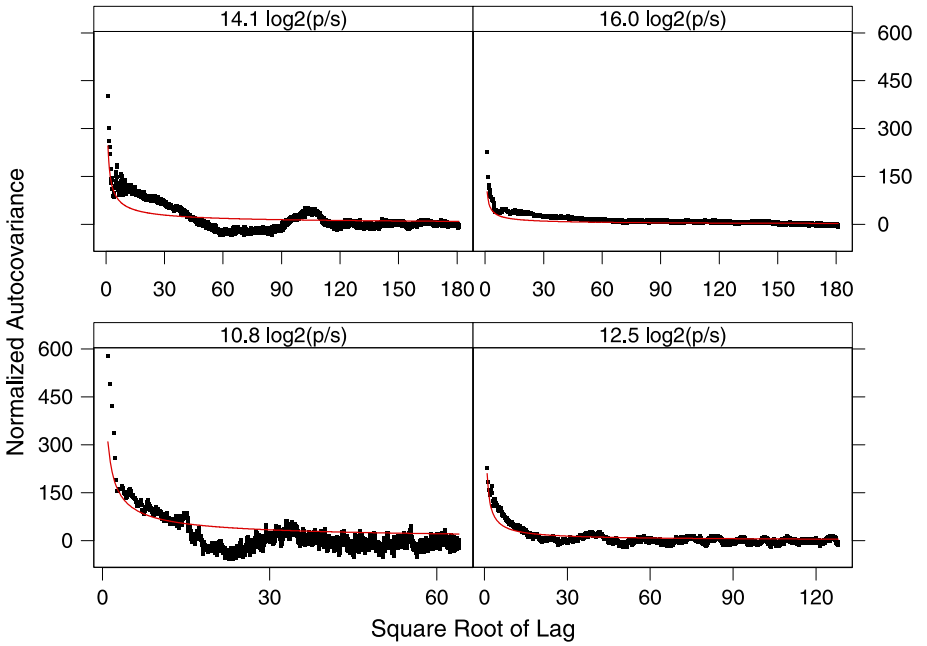

Fig. 22. For $m = 128$, normalized nonparametric estimates of autocovariances (•), and normalized approximate autocovariances (—) for $t_v^{(m)}$ are plotted against $\sqrt{k}$ for the 4 Auckland trace segments.

large variability. In Fig. 22 the normalized approximate and nonparametric autocovariances are plotted against square root lag for $m = 128$. The fits are excellent, again given the variability due to long-range dependence.

14. VoIP traffic engineering

14.1. Background

The MFSD model in this article applies to the superposed packet arrival process of the many different types of applications on commodity Internet links. For example, a core link on the Comcast or Verizon or AT&T network. One part of traffic engineering addresses this aggregate traffic.

However, contained within this traffic are applications that need special attention because they have more stringent QoS requirements than most applications. This is true of certain real–time applications. VoIP, voice over the Internet is one of them. Service providers often give priority queueing to VoIP packets. An arriving VoIP packet is moved forward in the queue to be in front of all packets with no priority queueing.

Among the handful of studies that report on VoIP packet traces, [72,73] study VoIP packet traces collected on FastWeb, where VoIP did not get priority queueing. [74] analyzed live VoIP data collected on the Global Crossing international network, where VoIP calls received priority queueing. They studied the statistical properties of the VoIP traffic as it enters the network on VoIP gateways. They built a generation model for this "offered load". It operates quite differently from the MFSD. Each call consists of two semi-calls, caller-to-callee and callee-to-caller. Individual semi-calls are modeled. To carry out generation for a simulation, semi-calls are generated through time and multiplexed on the computer, resulting in a single stream for simulation.

VoIP packets need to reach the destination quickly, no more than about 150 ms from end-to-end. VoIP packets have 20 ms spacing, which must be maintained to a degree at the destination so the codec can receive them and assemble them to provide real–time continuous speech. Deviations from 20 ms are jitter; absolute jitter must achieve an upper bound of 30 ms with very high probability.

One question is whether priority queueing is needed. We ran a small simulation to illustrate one way to answer the question. A real simulation would need to be much more expansive in investigating a very wide range of traffic rates. In our illustrative simulation the VoIP packets are multiplexed with MFSD packets but are not given priority queueing. We investigate how much MFSD traffic can be mixed with the VoIP traffic before VoIP QoS problems arise.

For this simulation, we do not need to use VoIP traffic, for example, generated by the above VoIP model. Simple test VoIP traffic will do because the result is dominated by the queueing properties of the MFSD traffic. As in real life, in our simulation the VoIP traffic has a much smaller bit rate.

Table 2
Tail quantiles of absolute jitter and delay in milliseconds.

Quantile	Utilization 0.1		Utilization 0.3		Utilization 0.5	
	Absolute jitter	Delay	Absolute jitter	Delay	Absolute jitter	Delay
0.99	0.0456	0.0431	0.2305	0.2085	0.7692	0.6980
0.999	0.0729	0.0642	0.3486	0.3250	1.8256	1.7322
0.9999	0.0932	0.0884	0.4742	0.4517	3.5570	4.0116

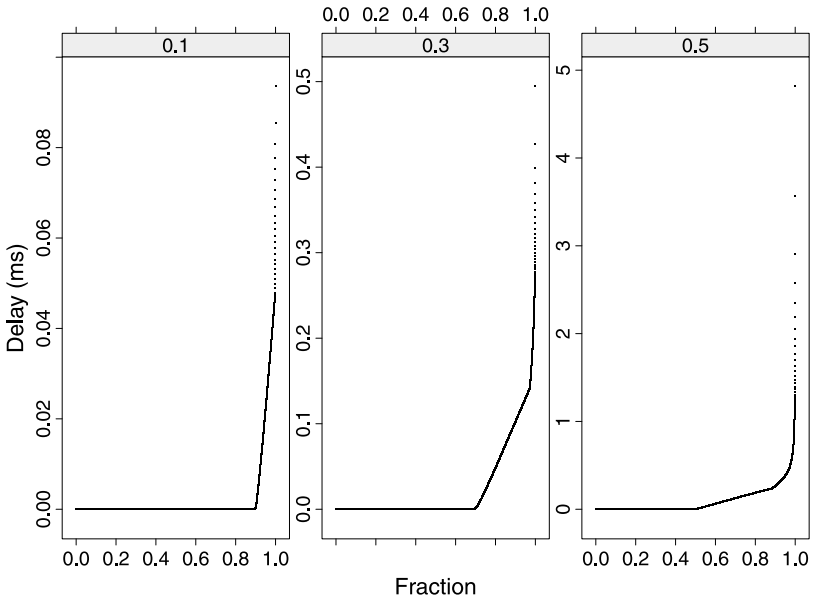

Fig. 23. Quantiles of delay.

14.2. Illustrative simulation

We generated commodity multi-application traffic using our MFSD model and the fast method of Section 13. We need packet sizes, too, for this simulation. The packet sizes were generated independently using the empirical distribution of sizes from the traffic collection of Section 4. We set the packet rate as $\alpha = 2^{12}$ packets/second. The average packet size is 772.22 bytes/packet. So the mean bit-rate of the synthetic multi-application traffic is 25.3 megabits/second.

We take the VoIP packets to be 200 bytes each, the value for the commonly used 64 kilobits/second capture rate, take the 20 ms accumulation interval, as in [74]. We generate a constant stream of VoIP packets arriving at the queue at a rate of one per ms. This represents a stream for multiple semi-calls. Note the VoIP traffic rate of 1.6 megabits/second is small compared with α, that of the other traffic. The synthetic multi-application traffic and the synthetic VoIP traffic are superposed to form the input stream to a FIFO queue. Simulation run time was 1 h.

We describe the three simulation runs of our study. For each, we first choose the utilization U, the average traffic rate divided by the link speed. The three runs have values $U = 0.1$, 0.3, and 0.5 respectively. The traffic packet rate is $\alpha = 2^{12}$ packets/second $= 25.3$ megabits/second, so the corresponding link speeds are $L = 25.3/U$ megabits/second.

Let A_i be the arrival time of the ith packet and B_i its packet size. The queueing delay of the ith packet in the multiplexed packet stream is calculated as

$$D_i = [D_{i-1} + B_{i-1}/L - (A_i - A_{i-1})]^+.$$

In each run, we study the distribution of the delays of the VoIP packets V_k. Jitter is computed from the delays of the VoIP packets that are 20 ms apart,

$$J_k = V_k - V_{k-20}.$$

We study the distribution of the absolute jitter values $|J_k|$.

For each of the three utilizations – $U = 0.1$, 0.3, 0.5 – we compute the sample quantiles of the two distributions, delay and absolute jitter, at frequencies 0.00005 to 0.99995 in steps of 0.0001. Figs. 23 and 24 graph the quantiles against the frequencies. Both delay and absolute jitter increase significantly as U increases. VoIP QoS criteria specify upper bounds, so the upper tails are of particular interest. Table 2 shows the 0.99, 0.999, 0.9999 quantiles.

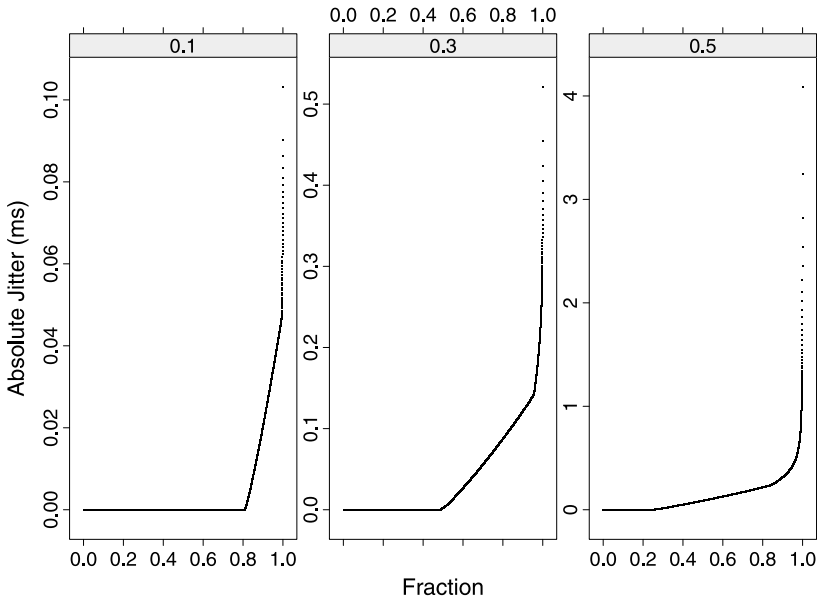

Fig. 24. Quantiles of absolute jitter.

Now the simulation study is for one hop which means one queue. The criteria – 150 ms for delay and 30 ms for absolute jitter – are end-to-end, which means many queues. Other studies show that the number of hops has a limit of about 30. To be conservative we use this for planning. We simply divide the end-to-end criteria to get single hop criteria of 5 ms for delay and 1 ms for absolute jitter. Table 2 shows that a utilization between 0.3 and 0.5 would satisfy criteria for a traffic rate of 25.3 megabits/second. Simulation runs at other utilizations can, of course, find exact values. Just these values alone are encouraging. Elimination of VoIP priority queueing might in fact be feasible. The traffic rate of 25.3 megabits/second is not large, and a U in the range suggested for this rate is not unreasonable. Furthermore, higher traffic rates can be expected to give even better results.

15. Summary and discussion of results

15.1. What is needed for traffic engineering

Following the discovery of long-range dependence of Internet traffic in 1994–1995, there was a very active 15-year period of empirical study that provided much understanding of the properties of the packet arrival process. However, what was not achieved was the development of a mathematical, statistical model for the interarrival process that achieved three important goals: (1) extensive validation; (2) mathematical tractability; (3) a usable synthetic generation of packet arrivals for studies in network engineering. Work declined over the 15-year period in model development. However, the need for such a model for mathematical study and simulation study is as strong as in 1994. Network traffic engineering depends heavily on both. Network traffic engineering is still critical for optimal Internet performance.

15.2. What has been achieved by this work

MFSD and GFSD models were first put forward in [17,18]. In the first reference, the long-range dependence was described by a nonparametric estimate of the power spectrum. In the second reference this was replaced by a FARIMA model for the long-range dependent series which here we denote s_u: $(I - B)^d s_u = \epsilon_u + \epsilon_{u-1}$ where ϵ_u is white noise.

Here we use s_u, but reformulate it to be $s_u = h_u + h_{u-1}$ a first-order moving average of Hosking's fractional Gaussian noise (fGn). This allows us to derive here, for the first time, many exact or approximate formulas for informative quantities such as autocorrelations. This gives immense quantitative insights about the traffic that in the past have only been described qualitatively and empirically, for example, by the sample autocorrelation function. One reason for a clear understanding of properties is that we verify, surprisingly for the first time, that Hosking's h_u is so close to self-similar that we can think of it as such. Furthermore, we are able to include in the formulas, the dependence of the quantities on the traffic rate, α p/s. This change is a critical property not understood before this paper.

Being able to derive quantities, has allowed us to carry out a much deeper model validation of the MFSD and GFSD than in [17,18]. In fact, the depth and breadth of the validation here is unprecedented in Internet traffic modeling.

For traffic generation here, one approximation allows a much faster generation than in [17,18]. Also, the model parameters θ and λ are modeled as a function of the traffic rate α p/s. So a simulation needs only to specify the rate, which makes generation as simple and straightforward as is possible.

One surprising result is that while the nonlinear transformation of the GFSD to the MFSD certainly changes dramatically the statistical properties of the MFSD, the second moment properties of the MFSD are very close to those of a GFSD.

15.3. Other modeling approaches

Multiple methods were proposed in the past to model and generate self-similar traffic. A $M/G/\infty$ approach was used in [10], where connections were generated following a Poisson process and connection durations followed heavy tailed distribution. [75] developed a random midpoint placement algorithm to generate an approximate fBm series. The algorithm recursively subdivided an interval, and gave a value at the midpoint based on the values at the end points. [23] used a fast Fourier approach to generate an approximate fGn series. [76] used a further approximation to fGn power spectrum in generation. [77] generated a large number of ON/OFF sources. Aggregation of the ON/OFF sources converged to fBm. The problem, as we have clearly shown here, is that the traffic is not well modeled by a self-similar process, which has very strong assumptions, and is generally not likely to accurately characterize the stochastic properties of time series data.

It is surprising that models like the MFSD and GFSD, given their simplicity, did not appear much earlier. One reason is that the vast majority of traffic studies analyzed packet counts in fixed intervals such as 10 ms, instead of the packet interarrivals. Notably, [29–31,36,39,47,45,78,79] studied the aggregate traffic, i.e., packet counts/bytes on different times scales. [29] performed a multifractal moment analysis of measured traces. WAN traces showed asymptotic self-similar behavior at larger time scales, and multifractal scaling behavior at smaller time scales. [30,31] used a cascade structure to model the packet counts on different time scales, to capture the complex scaling behavior of the packet counts. [39] suggested that the hierarchical nature of the IP networks gave rise to the observed scaling phenomenon. [47] studied the wavelet spectrum of the packet counts and provided a wavelet estimator of the Hurst parameter. [45] used a n-level hierarchical on–off process to model the aggregate traffic. [78] applied wavelet techniques to fGn, FARIMA processes, and measured traffic traces to determine the time scales where long range dependence dominates. [79] generated on–off flows in a large scale controlled experimental environment, measured and studied the aggregate traffic series.

The aggregate traffic represents a data reduction that makes computation easier through having a much smaller dataset. However, it is not possible to discover and investigate the MFSD by analyzing just counts. Although important properties of the network traffic were learned from the study of the moments and the scaling of the counts, the aggregate traffic does not describe the stochastic process of the individual packet arrivals. The work here addresses directly the arrival process because that is what the routers see, and what must be addressed in studies of network engineering, and because it is mathematically, the foundational process.

A cascade construction was used [80,52] to generate multifractal aggregate traffic. The multipliers in the cascade structure were random variables. [32] developed a multifractal wavelet model, which was an extension of the simple random multiplicative cascades. The wavelet and the scaling coefficients followed a special multiplicative structure. They can be estimated from the measured traces, and in turn be used to generate synthetic traffic data [81].

Th multifractal wavelet models and the random cascade models are complex and have a large number of parameters. They do not give much real insight about the stochastic properties of traffic. Looking at moments of a random variable does not give good insight into the variables distribution.

Furthermore, wavelets and cascades are not attractive for generation. They are nonparametric. To generate at rate α p/s one has to find live traffic with rate α p/s, fit, and then generate. We must remember we cannot generally generate traffic at rate α_0 p/s by generating at rate $\gamma\alpha_0$ p/s, and then dividing the interarrivals by γ, because much more than the mean of the interarrivals changes with changing rate. It is only true if one is considering traffic rates so high that the arrival process is Poisson.

Acknowledgments

William S. Cleveland was supported by ARO MURI Award W911NF-08-1-0238, NSF FODAVA Award CCF-0937123 and NSF DMS-122834. Bowei Xi was supported by NSF DMS-0904548, ARO MURI Award W911NF-08-1-0238, NSF DMS-1228348 and ARO W911NF-12-1-0558.

Appendix. Derivations of λ and θ for changing α

A.1. Heuristic derivation of λ

We present a heuristic derivation for λ in this section, under the simplified assumption of renewal processes. Assume there are r i.i.d. renewal processes, where r is a positive integer. Each renewal process has a Weibull marginal distribution with parameters α and λ. From the analysis of the trace segments in the previous sections, we notice the marginal

distribution for the live traffic statistical multiplexing process (superposition process) is Weibull with parameters $\lambda(r)$ and $\alpha(r)$, where the increasing traffic rate is

$$\alpha(r) = r\alpha.$$

Based on the Weibull marginal distribution for the individual renewal processes, we have the marginal density of their statistical multiplexing process [82]:

$$g^r(t) = -\frac{d}{dt}\left(e^{-\left(t\alpha\Gamma\left(1+\frac{1}{\lambda}\right)\right)^{\lambda}}\left(\int_t^{\infty}\alpha e^{-\left(x\alpha\Gamma\left(1+\frac{1}{\lambda}\right)\right)^{\lambda}}dx\right)^{r-1}\right).$$

The median β^r of the distribution with density $g^r(t)$ is found by solving the following equation:

$$e^{-\left(\beta^r\alpha\Gamma\left(1+\frac{1}{\lambda}\right)\right)^{\lambda}}\left(1-\alpha\int_0^{\beta^r}e^{-\left(x\alpha\Gamma\left(1+\frac{1}{\lambda}\right)\right)^{\lambda}}dx\right)^{r-1} = 0.5. \tag{31}$$

We approximate the distribution with density $g^r(t)$ by a Weibull distribution with parameters $\lambda(r)$ and $\alpha(r)$ whose median matches β^r.

$$\beta^r = \frac{(\log 2)^{\frac{1}{\lambda(r)}}}{r\alpha\Gamma(1+1/\lambda(r))}. \tag{32}$$

We solve Eq. (32) for $\lambda(r)$. The right hand side of Eq. (32) changes monotonically with $\lambda(r)$. Hence there is a unique solution for $\lambda(r)$.

A.2. Heuristic derivation of θ

Assume there are r i.i.d. MFSD source processes t_u. Each has a Weibull marginal distribution with parameters α and λ. The corresponding Gaussian image z_u of a source process t_u follows a GFSD model with parameters θ and d. From the analysis of trace segments in the previous sections, we observe the values of the fractional difference parameter d do not change appreciably under different traffic rates. We then fix the value of d to be the median of the estimates in Section 6. Let

$$d(r) = d = 0.31.$$

For the Weibull marginal distribution of the statistical multiplexing process t_u^r, we have the estimates $\hat{\alpha}(r) = r\alpha$, and $\hat{\lambda}(r)$ obtained from Eq. (32).

$\theta(r)$ can be obtained from the autocorrelation at lag 1 for the Gaussian image z_u^r of the statistical multiplexing process t_u^r as follows:

$$\theta(r) = 1 - \frac{\rho_z^r(1)(2-d)}{d+1}. \tag{33}$$

To estimate $\theta(r)$, we first compute $\rho_t^r(1)$ for the statistical multiplexing process t_u^r, then obtain $\rho_z^r(1)$ from $\rho_t^r(1)$, and apply Eq. (33).

In order to obtain $\rho_t^r(1)$, first we examine the sources of the arrivals that lead to the two consecutive interarrival times in the superposition process, t_u^r and t_{u+1}^r. Assume the uth arrival a_u^r in the statistical multiplexing process is the jth arrival from source 1. Note $t_{u+1}^r = a_{u+1}^r - a_u^r$. Let $t_{i,j}$ be the jth interarrival time from source i, and V_i be a forward recurrence time (the time from an arbitrary time point until the next arrival) for source i. To find the autocorrelation between t_u^r and t_{u+1}^r, we examine the following five cases:

1. The arrivals a_{u-1}^r and a_{u+1}^r both come from source 1. Then t_u^r and t_{u+1}^r are two consecutive interarrival times from source 1. We have $\rho_t^r(1) = \rho_t(1)$.
2. The arrival a_{u+1}^r comes from source 1, but the arrival a_{u-1}^r comes from a different source i. We have $t_{u+1}^r = t_{1,j+1}$. And $t_u^r = V_1 = t_{1,j} - \eta$, where η is the sum of interarrival times and forward recurrence times from sources other than source 1. Then as in Case 1, $\rho_t^r(1) = \rho_t(1)$.
3. The arrival a_{u-1}^r comes from source 1, but the arrival a_{u+1}^r comes from a different source i. Then $t_u^r = t_{1,j}$ and $t_{u+1}^r = V_i$. Since the individual source processes are independent, $\rho_t^r(1) = 0$.
4. The arrivals a_{u-1}^r and a_{u+1}^r come from different sources, and neither comes from source 1. Assume arrival a_{u+1}^r come from source i, $i \neq 1$. Then $t_u^r = V_1$ and $t_{u+1}^r = V_i$. We have $\rho_t^r(1) = \mathrm{Corr}(V_1, V_i) = 0$.
5. The arrivals a_{u-1}^r and a_{u+1}^r both come from the same source i, $i \neq 1$. This is exactly the same as Case 4. Again $t_u^r = V_1$ and $t_{u+1}^r = V_i$. $\rho_t^r(1) = 0$.

Hence under the first two cases $\rho_t^r(1) = \rho_t(1)$, while under the last three cases $\rho_t^r(1) = 0$. Let the minimum forward recurrence time from all other sources be $V_{\min} = \min_{\{i=2\ldots r\}} V_i$. Case 1 and 2 occur when a_u^r and a_{u+1}^r both come from source 1. This implies $t_{1,j+1} < V_{\min}$. Therefore,

$$\Pr(\text{Case 1 or 2}) = \Pr(t_{1,j+1} < V_{\min}).$$

Thus, we have

$$\rho_t^r(1) = \Pr(t_{1,j+1} < V_{\min})\text{Corr}(t_{1,j}, t_{1,j+1} | t_{1,j+1} < V_{\min}). \tag{34}$$

It remains to solve for both terms on the right hand side of Eq. (34). Based on the density of a forward recurrence time from one source process [82], we obtain the density for $V_{\min}$, the minimum of $r-1$ forward recurrence times. Since $t_{1,j+1}$ is Weibull with parameters λ and α, and independent of $V_{\min}$, we have

$$P(t_{1,j+1} < V_{\min}) = 1 - (r-1)\alpha \int_0^\infty e^{-2y^\lambda/\psi} \left(1 - \alpha \int_0^y e^{-x^\lambda/\psi} dx\right)^{r-2} dy,$$

where $\psi = \left(\alpha \Gamma(1 + \frac{1}{\lambda})\right)^{-\lambda}$.

Next we approximate the joint density of two consecutive interarrival times from the same source process, t_j and t_{j+1}, by converting the joint density of their Gaussian images z_j and z_{j+1}, $z_j = Z^{-1}(W(t_j))$, and ignoring the correlation between t_j and t_{j+1} in the Jacobian matrix. Let $\rho = \rho_z(1) = \text{Corr}(z_j, z_{j+1})$. We have the following approximate density, up to a normalizing factor:

$$f_W(t_j, t_{j+1}) \propto \frac{1}{\sqrt{1-\rho^2}} \exp\left\{\frac{-1}{2-2\rho^2}\left[\rho^2\left(Z^{-1}(W(t_j)) + Z^{-1}(W(t_{j+1}))\right)^2 + 2(\rho^2 - \rho)Z^{-1}(W(t_j))Z^{-1}(T(t_{j+1}))\right]\right\} \frac{\lambda^2}{\psi^2}(t_j t_{j+1})^{\lambda-1} \exp\{-(t_j^\lambda + t_{j+1}^\lambda)/\psi\}.$$

Using $f_W(t_j, t_{j+1})$ combined with $P(t_{1,j+1} < V_{\min})$, $\text{Corr}(t_{1,j}, t_{1,j+1} | t_{1,j+1} < V_{\min})$ can be calculated.

There are two ways to compute $\rho_t^r(1)$, either using Eq. (34) or directly using the approximate joint density $f_W(t_j, t_{j+1})$, because $f_W(t_j, t_{j+1})$ can be applied to two consecutive interarrival times t_u^r and t_{u+1}^r in the superposition process as well. This provides a numerical method for us to find $\rho_z^r(1)$ from a given $\rho_t^r(1)$.

We first obtain the estimate $\hat{\rho}_t^r(1)$ using Eq. (34). With $\hat{\lambda}(r)$ obtained using Eq. (32) and $\hat{\alpha}(r) = r\alpha$, we evaluate $f_W(t_j, t_{j+1})$ over a grid of potential $\rho_z^r(1)$ values. For each $\rho_z^r(1)$ value we compute the corresponding $\rho_t^r(1)$ directly using $f_W(t_j, t_{j+1})$. The estimate $\hat{\rho}_z^r(1)$ is the one that provides the closest match to $\hat{\rho}_t^r(1)$ obtained from Eq. (34). Then we apply Eq. (33) to have an estimate of $\hat{\theta}(r)$.

References

[1] W. Stevens, TCP/IP Illustrated, Volume 1: The Protocols, Addison-Wesley, 1994.
[2] L. Peterson, B. Davie, Computer Networks: A Systems Approach, Morgan Kaufmann, 1999.
[3] G. Kesidis, An Introduction to Communication Network Analysis, Wiley-IEEE Press, 2007.
[4] D. Rolls, G. Michailidis, F. Hernández-Campos, Queueing analysis of network traffic: Methodology and visualization tools, Comput. Netw. 48 (3) (2005) 447–473.
[5] F. de Pereira, N. da Fonseca, D. Arantes, On the performance of generalized processor sharing servers under long-range dependent traffic, Comput. Netw. 40 (2002) 413–431.
[6] P. Belottia, A. Caponeb, G. Carellob, F. Malucelli, Multi-layer mpls network design: The impact of statistical multiplexing, Comput. Netw. 52 (2008) 1291–1307.
[7] P. Bogdan, R. Marculescu, Non-stationary traffic analysis and its implications on multicore platform design, IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. 30 (2011) 508–519.
[8] F.H. Vieira, L.L. Lee, An admission control approach for multifractal network traffic flows using effective envelopes, AEU-Int. J. Electron. Commun. 64 (2010) 629–639.
[9] W. Leland, M. Taqqu, W. Willinger, D. Wilson, On the self-similar nature of ethernet traffic, IEEE/ACM Trans. Netw. 2 (1994) 1–15.
[10] V. Paxson, S. Floyd, Wide-area traffic: The failure of Poisson modeling, IEEE/ACM Trans. Netw. 3 (1995) 226–244.
[11] N. Duffield, Economies of scale in queues with sources having power-law large deviations scalings, J. Appl. Probab. 33 (1996) 840–857.
[12] A. Erramilli, O. Narayan, W. Willinger, Experimental queueing analysis with long-range dependent packet traffic, IEEE/ACM Trans. Netw. 4 (1996) 209–223.
[13] D. Heyman, T. Lakshman, What are the implications of long-range dependence for VBR-video traffic engineering? IEEE/ACM Trans. Netw. 4 (1996) 301–317.
[14] K. Park, G. Kim, M. Crovella, On the effect of traffic self-similarity on network performance, in: Proceedings SPIE Intl. Conf. Perf. and Control of Network Systems, 1997.
[15] V. Ribiero, R. Riedi, R. Baraniuk, Multiscale queuing analysis, IEEE/ACM Trans. Netw. 14 (5) (2006) 1005–1018.
[16] D.R. Cox, H.D. Miller, The Theory of Stochastic Processes, Chapman and Hall, London, 1977.
[17] J. Cao, W. Cleveland, D. Lin, D. Sun, Internet traffic tends toward Poisson and independent as the load increases, in: C. Holmes, D. Denison, M. Hansen, B. Yu, B. Mallick (Eds.), Nonlinear Estimation and Classification, Springer, New York, 2002, pp. 83–109.
[18] J. Cao, W.S. Cleveland, D.X. Sun, Bandwidth estimation for best-effort Internet traffic, Statist. Sci. 19 (2004) 518–543.
[19] J. Hosking, Fractional differencing, Biometrika 68 (1) (1981) 165–176.
[20] I. Csabai, 1/f noise in computer network traffic, J. Phys. A: Math. Gen. 27 (1994) L417–L421.
[21] W. Willinger, M. Taqqu, W. Leland, D. Wilson, Self-similarity in high-speed packet traffic: Analysis and modeling of ethernet traffic measurements, Statist. Sci. 10 (1995) 67–85.

[22] W. Willinger, M. Taqqu, A. Erramilli, A bibliographical guide to self-similar traffic and performance modeling for modern high-speed networks, in: F.P. Kelly, S. Zachary, I. Ziedins (Eds.), Stochastic Networks: Theory and Applications, Clarendon Press (Oxford University Press), Oxford, 1996, pp. 339–366.
[23] V. Paxson, Fast approximate synthesis of fractional Gaussian noise for generating self-similar network traffic, Comput. Commun. Rev. (1997) 5–18.
[24] M. Taqqu, W. Willinger, R. Sherman, Proof of a fundamental result in self-similar traffic modeling, Comput. Commun. Rev. 27 (1997) 5–23.
[25] I. Norros, A storage model with self-similar input, Queueing Syst. 16 (1994) 387–396.
[26] P. Abry, R. Baraniuk, P. Flandrin, R. Riedi, D. Veitch, The multiscale nature of network traffic: Discovery, analysis, and modelling, IEEE Signal Process. Mag. 19 (2002) 28–46.
[27] R. Riedi, J. Vehel, Multifractal properties of TCP traffic: a numerical study, Tech. Rep. 3129, INRIA Rocquencourt, France, 1997.
[28] V. Ribeiro, Z. Zhang, S. Moon, C. Diot, Small-time scaling behavior of Internet backbone traffic, Comput. Netw. 48 (3) (2005) 315–334.
[29] M. Taqqu, V. Teverovsky, W. Willinger, Is network traffic self-similar or multifractal, Fractals 5 (1997) 63–73.
[30] A. Feldmann, A.A. Gilbert, W. Willinger, Data networks as cascades: Investigating the mulifractal nature of Internet WAN traffic, in: ACM SIGCOMM, 1998, pp. 42–55.
[31] A. Gilbert, W. Willinger, A. Feldmann, Scaling analysis of random cascades, with applications to network traffic, IEEE Trans. Inform. Theory 45 (3) (1999) 971–991.
[32] R. Riedi, M. Crouse, V. Ribeiro, R. Baraniuk, A multifractal wavelet model with application to network traffic, IEEE Trans. Inform. Theory 45 (3) (1999) 992–1019.
[33] A. Veres, M. Boda, The chaotic nature of TCP congestion control, in: IEEE INFOCOMM, 2000, pp. 1715–1723.
[34] J. Yuan, Y. Ren, X. Shan, Self-organized criticality in a computer network model, Phys. Rev. E 61 (2) (2000) 1067–1071.
[35] J. Hannig, J. Marron, R. Riedi, Zooming statistics: Inference across scales, J. Korean Stat. Soc. 30 (2001) 327–345.
[36] A. Erramilli, M. Roughan, D. Veitch, W. Willinger, Self-similar traffic and network dynamics, in: Proceedings of the IEEE, 2002, pp. 800–819.
[37] D. Figueiredo, B. Liu, V. Misra, D. Towsley, On the autocorrelation structure of TCP traffic, Comput. Netw. 40 (2002) 339–361.
[38] T. Mikosch, S. Resnick, H. Rootzen, A. Stegeman, Is network traffic approximated by stable Levy motion or fractional Brownian motion? Ann. Appl. Probab. 12 (1) (2002) 23–68.
[39] W. Willinger, R. Govindan, S. Jamin, V. Paxson, S. Shenker, Scaling phenomena in the Internet: Critically examining criticality, Proc. Natl. Acad. Sci. USA 99 (3) (2002) 2573–2580.
[40] T. Dang, S. Molnar, I. Maricza, Some results on multiscale queueing analysis, in: Proceedings of the 10th International Conference on Telecommunications ICT, Papeete, French Polynesia, 2003, pp. 1631–1638.
[41] N. Liu, J. Baras, Statistical modeling and performance analysis of multi-scale traffic, in: Proceedings of IEEE Infocom, 2003.
[42] K. Maulik, S. Resnick, Small and large time scale analysis of a network traffic model, Queueing Syst. 43 (2003) 221–250.
[43] S. Resnick, A. Gilbert, W. Willinger, Wavelet analysis of conservative cascades, Bernoulli 9 (1) (2003) 97–135.
[44] T. Karagiannis, M. Molle, M. Faloutsos, A. Broido, A nonstationary Poisson view of Internet traffic, in: Proceedings of IEEE INFOCOM, 2004.
[45] W. Gong, Y. Liu, V. Misra, D. Towsley, Self-similarity and long range dependence on the Internet: A Second look at the evidence, origins and implications, Comput. Netw. 48 (2005) 377–399.
[46] H. Jiang, C. Dovrolis, Why is the Internet traffic bursty in short time scales, in: Sigmetrics, ACM Press, 2005, pp. 241–252.
[47] S. Stoev, M. Taqqu, C. Park, J. Marron, On the wavelet spectrum diagnostic for Hurst parameter estimation in the analysis of Internet traffic, Comput. Netw. 48 (2005) 423–445.
[48] D. Veitch, N. Hohn, P. Abry, Multifractality in TCP/IP traffic: the case against, Comput. Netw. 48 (2005) 293–313.
[49] M. Masugi, T. Takuma, Multi-fractal analysis of IP-network traffic for assessing time variations in scaling properties, Physica D 225 (2007) 119–126.
[50] M. Roughan, D. Veitch, Some remarks on unexpected scaling exponents, SIGCOMM Comput. Commun. Rev. 37 (2007) 71–74.
[51] M. Ashoura, T. Le-Ngoc, Priority queuing of long-range dependent traffic, Comput. Commun. 31 (2008) 3954–3963.
[52] J. Gao, I. Rubin, Multiplicative multifractal modeling of long-range-dependent network traffic, Int. J. Commun. Syst. 14 (2001) 783–201.
[53] J. Gao, I. Rubin, Superposition of multiplicative multifractal traffic streams, in: Proceedings of ICC2000, 2001.
[54] R. Riedi, Multifractal processes, in: P. Doukhan, G. Oppenheim, M.S. Taqqu (Eds.), Theory and Application of Long-Range Dependence, Birkhauser, Basel, 2002, pp. 625–715.
[55] P. Bogdan, Mathematical modeling and control of multifractal workloads for data-center-on-a-chip optimization, in: Proceedings of the 9th International Symposium on Networks-on-Chip, 2015, p. 21.
[56] C. Fraleigh, F. Tobagi, C. Diot, Provisioning IP backbone networks to support latency sensitive traffic, in: IEEE INFOCOM, 2003.
[57] P. Shanga, Y. Lub, S. Kamae, Detecting long-range correlations of traffic time series with multifractal detrended fluctuation analysis, Chaos Solitons Fractals 36 (2008) 82–90.
[58] A. Feldmann, A. Gilbert, W. Willinger, T. Kurtz, The changing nature of network traffic: Scaling phenomena, SIGCOMM Comput. Commun. Rev. 28 (1998) 5–29.
[59] Endace DAG. http://www.endace.com/.
[60] The CAIDA Tools Site. http://caida.org/tools.
[61] WITS: Waikato Internet Traffic Storage. http://www.wand.net.nz/wits/catalogue.php.
[62] C. Hurvich, E. Moulines, P. Soulier, The fexp estimator for potentially non-stationary linear time series, Stochastic Process. Appl. 97 (2002) 307–340.
[63] J. Geweke, S. Porter-Hudak, The estimation and application of long memory time series models, J. Time Series Anal. 4 (1983) 221–238.
[64] W. Cleveland, C. Mallows, J. McRae, Ats methods: Nonparametric regression for non-Gaussian data, J. Amer. Statist. Assoc. 88 (1993) 821–835.
[65] L. Muscariello, M. Meillia, M. Meo, M.A. Marsan, R.L. Cigno, An MMPP-based hierarchical model of Internet traffic, in: 2004 IEEE International Conference on Communications, pp. 2143–2147.
[66] J. Beran, Statistics for Long-Memory Processes, Chapman & Hall, 1994.
[67] M. Crovella, A. Bestavros, Self-similarity in world wide web traffic: Evidence and possible causes, IEEE/ACM Trans. Netw. 5 (1997) 835–846.
[68] D. Tutsch, G. Babin, P. Kropf, Application-layer traffic analysis of a peer-to-peer system, IEEE Internet Comput. Mag. 12 (2008) 70–77.
[69] C. Park, F. Hernández-Campos, L. Le, J.S. Marron, J. Park, V. Pipiras, F.D. Smith, R.L. Smith, M. Trovero, Z. Zhu, Long-range dependence analysis of Internet traffic, J. Appl. Stat. 38 (2011) 1407–1433.
[70] K. Sriram, W. Whitt, Characterizing superposition arrival processes in packet multiplexers for voice and data, IEEE J. Sel. Areas Commun. 4 (1986) 833–846.
[71] K. Krishnan, A new class of performance results for a fractional Brownian traffic model, Queueing Syst. 22 (1996) 277–285.
[72] R. Birke, M. Mellia, M. Petracca, D. Rossi, Understanding VoIP from backbone measurements, in: INFOCOM 2007: The 26th IEEE International Conference on Computer Communications, 2007, pp. 2027–2035.
[73] D. Ciullo, M. Mellia, M. Meo, Traditional IP measurements: What changes in a today multimedia IP network, in: Telecommunication Networking Workshop on QoS in Multiservice IP Networks, IT-NEWS 2008, 2008, pp. 262–267.
[74] B. Xi, H. Chen, W.S. Cleveland, T. Telkamp, Statistical analysis and modeling of Internet VoIP traffic for network engineering, Electron. J. Stat. 4 (2010) 58–116.
[75] Wing-Cheong Lau, Ashok Erramilli, Jonathan L. Wang, Walter Willinger, Self-similar traffic generation: The random midpoint displacement algorithm and its properties, in: 1995 IEEE International Conference on Communications, 1995, pp. 466–472.
[76] Sergio Ledesma, Derong Liu, Synthesis of fractional Gaussian noise using linear approximation for generating self-similar network traffic, ACM SIGCOMM Comput. Commun. Rev. 30 (2000) 4–17.
[77] P. Pruthi, A. Erramilli, Heavy-tailed On/Off source behavior and self-similar traffic, in: 1995 IEEE International Conference on Communications, ICC'95, 1995, pp. 445–450.
[78] D. Veitch, P. Abry, M.S. Taqqu, On the automatic selection of the onset of scaling, Fractals 11 (04) (2003) 377–390.
[79] P. Loiseau, P. Gonçalves, G. Dewaele, P. Borgnat, P. Abry, P.V.-B. Primet, Investigating self-similarity and heavy-tailed distributions on a large-scale experimental facility, IEEE/ACM Trans. Netw. 18 (4) (2010) 1261–1274.
[80] K. Kant, On aggregate traffic generation with multifractal properties, in: 1999 Global Telecommunications Conference. GLOBECOM'99, pp. 1179–1183.
[81] V.J. Ribeiro, R.H. Riedi, M.S. Crouse, R.G. Baraniuk, Simulation of nonGaussian long-range-dependent traffic using wavelets, ACM SIGMETRICS Perform. Eval. Rev. 27 (1999) 1–12.
[82] D. Cox, Renewal Theory, Methuen and Co. Ltd., London, 1962.