

Joint Characterization of the Packet Arrival and Packet Size Processes of Multifractal Traffic based on Stochastic L-Systems

Paulo Salvador ^a, António Nogueira ^a, Rui Valadas^a

^aInstitute of Telecommunications, University of Aveiro
Campus Universitário, Aveiro, Portugal
phone: (+351) 234377900, fax: (+351) 234377901
e-mails: salvador@av.it.pt, nogueira@av.it.pt, rv@det.ua.pt

Multifractal behavior was recently observed in several traces of IP WAN traffic. This paper proposes a novel multifractal traffic model, which characterizes the joint process of packet arrivals and packet sizes. The construction of the traffic process is based on stochastic L-Systems, which were introduced by biologist A. Lindenmayer as a method to model plant growth. We work with a single L-System alphabet and production rule, where the alphabet is a set of pairs, and each pair element represents a packet arrival rate and a packet mean size. In this way, the traffic model is able to capture correlations between arrivals and sizes, leading to an accurate prediction of the queuing behavior. We provide a detailed comparison with a related multifractal model based on conservative cascades. Our results, that include applying the fitting procedure to real observed data with multifractal scaling behavior on both the packet arrival and packet size processes, show that our L-System based model can achieve excellent fitting performance in terms of first and second order statistics and queuing behavior.

keywords: Traffic modeling, Multifractal, L-System.

1. Introduction

Characterizing the correlations between packet arrivals and packet sizes is important for accurate modeling of IP traffic. In particular, this is required for providing a close fit of the queuing behavior (i.e., the packet loss ratio or average packet delay suffered in a network node). The queuing behavior addresses the effect of traffic on network performance, and is one of the most important criteria to assess the suitability of traffic models (and associated parameter fitting procedures). Here, the analysis consists in comparing the curves of packet loss ratio (or average packet delay) versus buffer size, obtained with the measured traces (through trace-driven simulation) and with the inferred traffic model (using again trace-driven simulation or numerical computation of the performance measures whenever possible). When dealing with models that characterize only the arrival process, it is common practice to assume that the packet size is fixed and equal to the average packet size of the measured trace. This may lead to large errors

This work was part of project POSI/CPS/42069/2001 "Traffic Modeling and Performance Evaluation of Multiservice Networks", funded by Fundação para a Ciência e Tecnologia, Portugal. P. Salvador wishes to thank Fundação para a Ciência e a Tecnologia, Portugal, for support under grant BD/19781/99.

when packets have variable size, such as in IP traffic. Moreover, models describing independently the packet arrivals and the packet sizes may still lack accuracy, due to the effect that correlations have on queuing behavior.

Recent analysis of measured Internet WAN traffic has revealed that multifractal structures, such as random cascades, can help explaining the behavior typically associated to networking mechanisms operating on small time scales (e.g. TCP flow control). A cascade (or multiplicative process) is a process that fragments a set into smaller and smaller components according to a fixed rule, and at the same time fragments the measure of the components by another (possibly random) rule. Random cascades were introduced by Mandelbrot as a physical model for turbulence [1]. In the traffic modeling context, the set can be interpreted as a time interval and the measure as the number of arrivals or bytes (in that interval). The multifractal nature of network traffic was first noticed by Riedi and Lévy Véhel [2]. Subsequently various studies have addressed the characterization and modeling of multifractal traffic, essentially within the framework of random cascades [3–6].

In [7] we have introduced the use of stochastic Lindenmayer-Systems (hereafter referred to as L-Systems) as a model for network traffic. In this paper, we propose a new modeling approach based on stochastic L-Systems, and its associated parameter fitting procedure, which jointly characterizes the packet arrival and packet size processes. L-Systems are string rewriting techniques which were introduced by biologist A. Lindenmayer in 1968 as a method to model plant growth [8]. They are characterized by an alphabet, an axiom and a set of production rules. The alphabet is a set of symbols; the production rules define transformations of symbols into strings of symbols; starting from an initial string (the axiom), an L-System constructs iteratively sequences of symbols replacing each symbol by the corresponding string according to the production rules. Stochastic L-Systems are a method to construct recursively random sequences with multifractal behavior [9]. The proposed traffic model is based on a single L-System where the alphabet is a set of pairs, and each pair element represents a packet arrival rate (number of arrivals per time interval, in pkts/sec) and a packet mean size (mean size computed over a time interval, in bytes). The traffic model also includes the ability of modeling multiple scaling behavior. A multifractal traffic model also characterizing the packet arrival and packet size processes was introduced in [6]. This model is based on so-called conservative cascades, which were suggested by Feldmann et al. [3] as a model for IP WAN traffic. When compared with conservative cascades, stochastic L-Systems allow for a dependence on the construction process (due to the production rules), that has meaningful physical explanation, and can help understanding the joint impact of network mechanisms and resource limitations on observed traffic. Moreover, in [6] the packet arrival and packet size processes are modeled independently. Our formulation allows characterizing correlations between packet arrivals and packet sizes, leading to an accurate prediction of the queuing behavior.

This paper is organized as follows. In section 2 we give some background on L-Systems. In sections 3 and 4 we present the traffic model and describe the associated fitting procedure. Section 5 discusses the results of applying the proposed fitting procedure to measured traffic traces. Finally, section 6 presents the conclusions.

2. L-Systems Background

The basic idea behind L-Systems is to define complex objects by successively replacing parts of a simple object using a set of rules. The L-System is a feedback machine that operates on strings of symbols. The set of symbols is called the alphabet. Starting from an initial state (called axiom), an L-System operates, at each iteration, by applying the set of production (or rewriting) rules simultaneously to all symbols of an input string to give an output string. For a comprehensive introduction to L-Systems see [9].

Consider a simple example of an organism growing through cell subdivisions. There are two types of cells represented by letters A and B. Cell subdivisions are modeled by replacing these symbols with strings of symbols: cell A subdivides into two cells represented by string AB; cell B subdivides into two cells represented by string AA. The ordering of the symbols is relevant in an L-System. The organism modeled by this L-System grows by repeated cell subdivisions. At birth the organism is the single cell A. After one subdivision the organism is two cells represented by string AB. After two subdivisions, the organism has four cells given by string ABAA, and after three subdivisions the organism has eight cells represented by string ABAAABAB. Using the formalism of L-Systems this growth process can be described as:

Alphabet:	{A,B}
Axiom:	A
Rules:	A \rightarrow AB
	B \rightarrow AA

The production rules can be stochastic. In stochastic L-Systems there may be several production rules for one symbol, and the specific rule is selected according to a probability distribution. Taking previous example, one production rule could be to convert A into AB with probability 0.4 or into BB with probability 0.6 (instead of converting always A into AB). In this case, after 3 iterations several strings are possible, e.g., ABAAABAB, BBAAABAB, or BBBB BBBB. Stochastic L-Systems are a method to construct recursively random sequences with multifractal behavior [9].

In the context of traffic modeling, one possibility is to interpret the L-System symbols as packet arrival rates, which are associated to time intervals in each time scale, and the L-System evolution as a traffic construction process that goes from coarser to finer time scales by duplicating the number of arrival rates according to some L-System production rule and, at the same time, halving the width of time intervals associated with each arrival rate. In this way, as the construction process progresses, more detail is included in the traffic process description. A natural axiom for this model is the average packet arrival rate (computed over all time intervals of interest), which must then be included in the alphabet of arrival rates. This model was introduced in [7].

3. Traffic Model

This section introduces the proposed multifractal traffic model, which jointly characterizes the packet arrivals and the packet sizes. The traffic process will be a time series of pairs describing the arrival rate and the mean size of packets over (small) non-overlapping time intervals. The traffic process construction is based on a stochastic L-System, with the following characteristics: (i) the alphabet is a set of pairs, where one pair element is a packet arrival rate, λ_l , and the

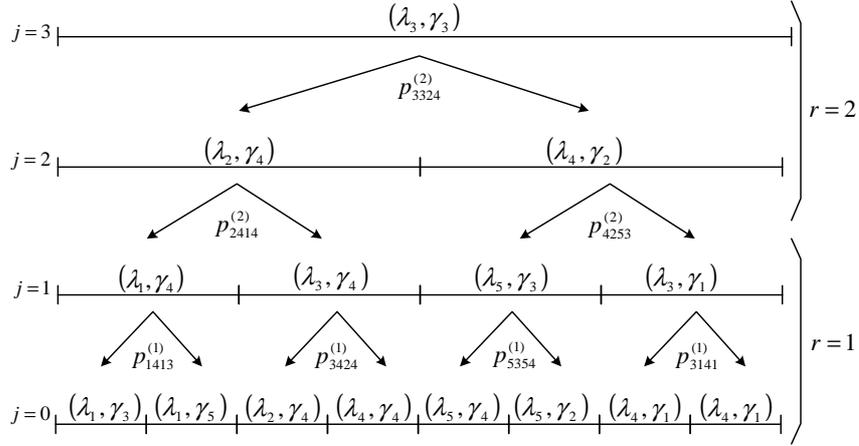


Figure 1. Construction of L-System based traffic model.

other is a mean packet size, γ_g , i.e.

$$\{(\lambda_l, \gamma_g) : \lambda_l \in \vec{\lambda}, \gamma_g \in \vec{\gamma}\} \quad (1)$$

where $\vec{\lambda} = \{\lambda_1, \lambda_2, \dots, \lambda_L\}$, $\lambda_l \in \mathbb{R}_0^+$, $l = 1, \dots, L$, $\vec{\gamma} = \{\gamma_1, \gamma_2, \dots, \gamma_G\}$, $\gamma_g \in \mathbb{R}_0^+$, $g = 1, \dots, G$; (ii) the production rules randomly generate two pairs (of arrival rates and mean sizes) from a previous one; (iii) the axiom is a predefined initial pair. Without loss of generality, we assume $\lambda_1 < \lambda_2 < \dots < \lambda_L$ and $\gamma_1 < \gamma_2 < \dots < \gamma_G$.

The traffic process is constructed progressively, governed by a L-System machine, where each iteration produces a new time scale. Starting with the coarsest time scale, where traffic is characterized by a single arrival rate and mean size pair over a single time interval, each iteration generates a finer time scale by (i) division of each (parent) time interval in two new equal length (child) subintervals and (ii) association of arrival rate and mean size pairs (belonging to the alphabet) to each new subinterval, according to the production rules of the stochastic L-System. We allow the grouping of time scales in time scale ranges and the definition of different sets of production rules for each time scale range. The consideration of different sets of production rules enables the incorporation of multiple scaling behavior, with distinct fractal characteristics, on the bytes-per-time process (process of arrival rates times mean sizes, in bytes/sec). The traffic process construction is illustrated in Figure 1.

To characterize the traffic process we define the pair $(X_{(j,r)}^{(i)}, Y_{(j,r)}^{(i)})$, where $X_{(j,r)}^{(i)} \in \vec{\lambda}$ and $Y_{(j,r)}^{(i)} \in \vec{\gamma}$ are the arrival rate and the mean size at time interval i of time scale j and time scale range r . Let the number of scales be S and the number of ranges of scales be R . For convenience, we let j decrease from $j = S - 1$ (at the coarsest time scale) to $j = 0$ (at the finest time scale). Also, we let r decrease from R (the ranges of coarsest time scales), to 1 (the range of finest time scales). Thus, the number of time intervals at time scale j , which we will denote by N_j , is 2^{S-j-1} . Moreover, assuming a unitary width for the intervals of the finest time scale, $j = 0$, the width at scale j will be 2^j . To relate time scales and time scale ranges we define j_r as the coarsest scales j in range r . Thus, in Figure 1, $S = 4$, $R = 2$, $j_2 = 3$ and $j_1 = 1$.

In order to assure that the mean packet arrival rate and the mean packet size (computed over all time intervals) are the same in all time scales, so as to maintain physical meaning, we will

impose the following conditions to the production rules:

$$X_{(j,r)}^{(i)} = \frac{1}{2}X_{(j-1,r')}^{(2i-1)} + \frac{1}{2}X_{(j-1,r'')}^{(2i)} \quad \text{and} \quad Y_{(j,r)}^{(i)} = \frac{1}{2}Y_{(j-1,r')}^{(2i-1)} + \frac{1}{2}Y_{(j-1,r'')}^{(2i)} \quad (2)$$

i.e., the mapping of arrival rates (mean sizes) is such that the arrival rate (mean size) averaged over the left and right child subintervals will be equal to the parent arrival rate (mean size).

The generation of arrival rates and mean sizes can be described by axiom $(X_{(S-1,R)}^{(1)}, Y_{(S-1,R)}^{(1)})$, the arrival rate and mean size at the coarsest time scale, and by production rules defined by

$$(X_{(j,r)}^{(i)}, Y_{(j,r)}^{(i)}) = (\lambda_l, \gamma_m) \xrightarrow{p_{lmqn}^{(r)}} \begin{cases} (X_{(j-1,r')}^{(2i-1)}, Y_{(j-1,r')}^{(2i-1)}) = (\lambda_q, \gamma_n) \\ (X_{(j-1,r'')}^{(2i)}, Y_{(j-1,r'')}^{(2i)}) = (2\lambda_l - \lambda_q, 2\gamma_m - \gamma_n) \end{cases} \quad (3)$$

where $\sum_{q=1}^L \sum_{n=1}^G p_{lmqn}^{(r)} = 1, \forall l, \forall m$. Thus, a pair with arrival rate λ_l and mean size γ_m in interval i , scale j and range r produces, with probability $p_{lmqn}^{(r)}$, a pair with arrival rate λ_q and mean size γ_n at the left subinterval $2i - 1$ and arrival rate $2\lambda_l - \lambda_q$ and mean size $2\gamma_m - \gamma_n$ at the right subinterval $2i$, of next scale $j - 1$ and range r' . The production rules can be totally described by the R matrices with dimension $L \times G \times L \times G$, $\mathbf{P}^{(r)} = (p_{lmqn}^{(r)})$, with $l, q = 1, \dots, L$, $m, n = 1, \dots, G$ and $r = 1, \dots, R$.

In order to guarantee that the alphabet of packet arrival rates is closed with respect to the production rules we impose the following conditions: (i) $\lambda_i - \lambda_{i-1} = \frac{\lambda_L - \lambda_1}{L-1}, i = 2, 3, \dots, L$, i.e., the λ_i values will be equidistant; (ii) $p_{lmqn}^{(r)} = 0$ if $q > l + \min(L - l, l - 1)$ or $q < l - \min(L - l, l - 1)$. Similar conditions have to be imposed on mean packet sizes. Thus, (i) $\gamma_i - \gamma_{i-1} = \frac{\gamma_G - \gamma_1}{G-1}, i = 2, 3, \dots, G$; (ii) $p_{lmqn}^{(r)} = 0$ if $n > m + \min(G - m, m - 1)$ or $n < m - \min(G - m, m - 1)$.

Finally, the L-System construction defines, at scale j and range r , the sequence $Z_{(j,r)} = \{(X_{(j,r)}^{(i)}, Y_{(j,r)}^{(i)}), i = 1, \dots, N_j\}$.

4. Fitting Procedure

The fitting procedure determines the parameters of the L-System from real data observations. It starts by fixing a sampling interval Δ and considering the time series representing the number of packet arrivals and the mean packet size in each non-overlapping sampling interval. Let this (empirical) time series be $\{(A_k, B_k), k = 1, 2, \dots, K\}$, where A_k and B_k represent the number of arrivals and the mean packet size in sampling interval k . For convenience, we take the length of the time series K to be a power of 2. The inference procedure can then be divided in three steps: (i) determination of alphabet and axiom, (ii) identification of the time scale ranges, (iii) inference of the production rules. The flow diagram of the L-System fitting procedure is represented in figure 2.

The L-System alphabet will consist in a set with LG pairs (λ_l, γ_g) , where both the λ_l and γ_g elements are equidistant values ranging from the minimum to the maximum present in data. Both L and G are a compromise between accuracy and complexity. Due to the mass conservation property of L-Systems defined in (2), the first element of the axiom pair is inferred as the mean arrival rate of $\{A_k\}$, and the second element of the axiom pair is inferred as the mean of $\{B_k\}$, both rounded to the closest alphabet element, i.e.,

$$X_{(S-1,R)}^{(1)} = \Lambda_{\bar{\lambda}} \left((1/K\Delta) \sum_{k=1}^K A_k \right) \quad \text{and} \quad Y_{(S-1,R)}^{(1)} = \Lambda_{\bar{\gamma}} \left((1/K) \sum_{k=1}^K B_k \right) \quad (4)$$

where $\Lambda_{\vec{\alpha}}(x)$ represents a function that rounds x towards the nearest element of $\vec{\alpha}$.

The identification of time scale ranges is based on wavelet scaling analysis. We use the method described in [5], which resorts to the (second-order) logscale diagram. A (second-order) logscale diagram is a plot of y_j against j , together with confidence intervals about the y_j , where y_j is a function of the wavelet discrete transform coefficients at scale j . The time scale ranges correspond to the set of time scales for which, within the limits of the confidence intervals, the y_j fall on a straight line, i.e., the scaling behavior is linear in a time scale range. This procedure is applied to the process of bytes-per-time obtained from $\{A_k B_k, k = 1, 2, \dots, K\}$. Figure 3 shows the logscale diagram of a trace measured at the University of Aveiro (which is described in section 5). It can be seen that there are 3 time scale ranges (within a total of 18 time scales) defined by $j_1 = 5, j_2 = 10, j_3 = 17$.

The next step is the inference of the production rules of the L-System, which are fully characterized by the $\mathbf{P}^{(r)}$ matrix. First, data is rounded in order to define the sequences of packet arrival rates and mean packet sizes at each time scale. This comprises obtaining the arrival rates $X_{(j,r)}^{(i)}$ from $\{A_k\}$ and the mean sizes $Y_{(j,r)}^{(i)}$ from $\{B_k\}$ through

$$X_{(j,r)}^{(i)} = \Lambda_{\vec{\lambda}} \left(\frac{N_j}{K\Delta} \sum_{k=K(i-1)/N_j+1}^{Ki/N_j} A_k \right) \quad \text{and} \quad Y_{(j,r)}^{(i)} = \Lambda_{\vec{\gamma}} \left(\frac{N_j}{K} \sum_{k=K(i-1)/N_j+1}^{Ki/N_j} B_k \right) \quad (5)$$

with $i = 1, \dots, N_j$, for each j .

Letting $c_{lmqn}^{(r)}$ represent the number of times that, over all scales of range r , the parent $(X_{(j,r)}^{(i)}, Y_{(j,r)}^{(i)}) = (\lambda_l, \gamma_m)$ produced the left child $(X_{(j-1,r')}^{(2i-1)}, Y_{(j-1,r')}^{(2i-1)}) = (\lambda_q, \gamma_n)$, the production rule probabilities can be inferred as

$$p_{lmqn}^{(r)} = c_{lmqn}^{(r)} / \sum_{q=1}^L \sum_{n=1}^G c_{lmqn}^{(r)} \quad (6)$$

with $l = 1, \dots, L, m = 1, \dots, G$ and $r = 1, \dots, R$.

5. Numerical Results

We have applied our fitting procedure to two traces of IP traffic: (i) the well known pOct.TL Bellcore trace and (ii) one trace measured at the University of Aveiro (UA) which exhibits non-trivial multifractal scaling behavior both on packet arrivals and packet sizes. The sampling interval was 0.1 seconds in both traces. The UA trace is representative of Internet access traffic produced within a University campus environment. The University of Aveiro is connected to the Internet through a 10 Mb/s ATM link and the measurements were carried out in a 100 Mb/s Ethernet link connecting the border router to the firewall, which only transports Internet access traffic. The UA trace consists of 20 millions packets captured on July 3rd 2001 from 8.00 pm to 3.15 am. The traffic analyzer was a 1.2 GHz AMD Athlon PC, with 1.5 Gbytes of RAM, running WinDump. The measurements recorded the arrival instant and the IP header of each packet. WinDump reported no packet drops during the measurement periods. The mean arrival rate of the UA trace is 766 pkts/sec.

The pOct.TL trace was fitted to an alphabet with 12150 elements, corresponding to $L = 243$ arrival rates and $G = 50$ mean packet sizes. The minimum and maximum arrival rates present

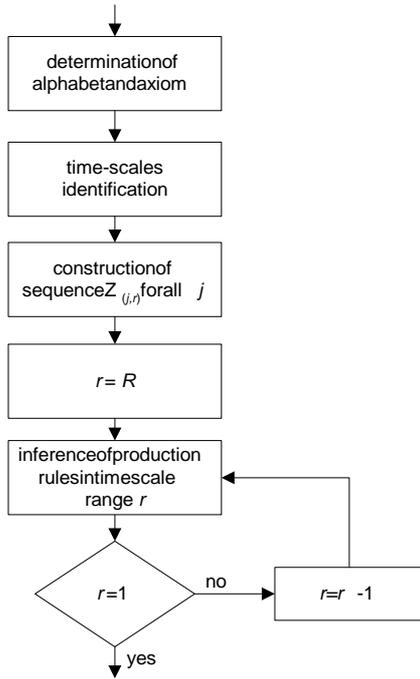


Figure 2. Flow diagram of inference procedure.

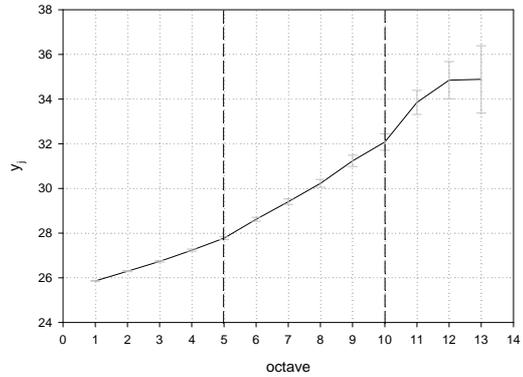


Figure 3. Logscale diagram, bytes-per-time of trace UA.

in data were 10 and 2430 pkts/sec, respectively, and a step of 10 pkts/sec was used (because it corresponds to one arrival in the sampling interval); in the case of mean sizes, the maximum and minimum values were 64 bytes and 1421 bytes, respectively, and a step of 28 bytes was used. The logscale diagram identified 4 time scale ranges (within a total of 14 time scales) defined by $j_1 = 5$, $j_2 = 8$, $j_3 = 9$ and $j_4 = 13$.

In the case of the UA trace the fitted alphabet had 23450 elements, corresponding to $L = 469$ arrival rates, ranging from 170 pkts/sec to 4850 pkts/sec, in steps of 10 pkts/sec, and $G = 50$ mean packet sizes ranging from 66 bytes to 1214 bytes, in steps of 23 bytes. As referred before, the UA trace has 3 time scale ranges (Figure 3).

The parameter estimation took less than 2 minutes, using a MATLAB implementation running in the PC described above. This shows that the fitting procedure is computationally very efficient (note that the size of the alphabet, the number of ranges and the size of the trace, which determine the computational time, are all relatively large).

We assess the suitability of the traffic model and the accuracy of the fitting procedure using several criteria. We compare both the probability mass and autocovariance functions of the bytes-per-time process obtained with the fitted stochastic L-System model and with the original data trace. We also analyze the queuing behavior by comparing the packet loss ratio, obtained through trace-driven simulation, using two types of input traffic: (i) the original trace and (ii) the trace generated according to the fitted stochastic L-System. Trace generation has two steps: (i) first the sequence of arrival rate and mean size pairs is constructed, according to the inferred L-System; (ii) second, within each time interval, the packet arrival instants are spaced uniformly (the number of arrivals is set according to the arrival rate of the interval) and all packets are assigned the same size (equal to the mean packet size of the interval).

Our comparisons are extended to a double conservative cascade model, that uses two inde-

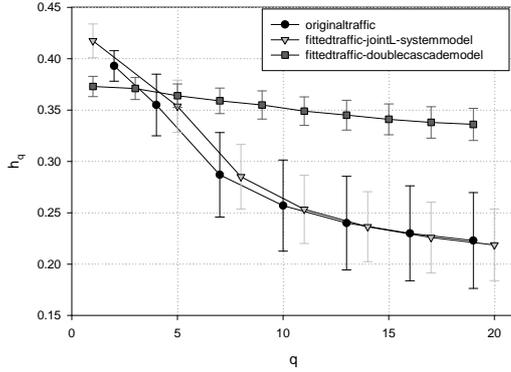


Figure 4. Linear multiscale diagram, packet arrival rates of trace UA.

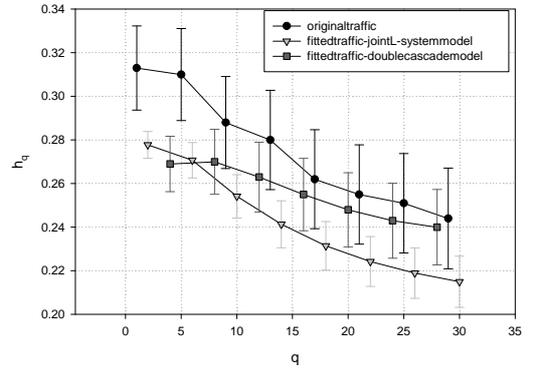


Figure 5. Linear multiscale diagram, packet mean sizes of trace UA.

pendent conservative cascades for the characterization of packet arrival rates and mean packet sizes. In conservative cascades, the traffic process is constructed iteratively from the largest to the smallest time scale, as in L-Systems. A mass (which can be interpreted as number of arrivals or bytes) is assigned to the largest time scale. Then, the mass is redistributed in each time scale such that the mass of each parent interval is distributed to a left and a right child intervals, under the control of a random variable W , called the generator, that takes values in $(0, 1)$, has mean $1/2$ and is symmetric about its mean. If the mass of the parent interval is Q , the mass of the left child interval will be QW and that of the corresponding right child interval will be $Q(1 - W)$. As a result, the mass at all stages will be (exactly) M , if the initial mass is M . This is the mass conservation property which also occurs in our stochastic L-System constructions. In this work, each cascade model was inferred by fitting the generators at each time scale to truncated normal distributions, with mean always $1/2$ but variance adjusted individually at each time scale. We will refer to the proposed model as the joint L-System model.

Multifractality is assessed using a linear multiscale diagram [5], which plots $h_q = \alpha_q/q$ against q ; α_q represents the q^{th} order scaling exponent, estimated in the q^{th} order logscale diagram. Multifractal scaling behavior is detected when there is no horizontal alignment (within the limits of confidence intervals). Figure 4 and Figure 5 show that the packet arrival and packet size processes of the UA trace, and of the traces generated using the L-System and conservative cascade models, all have multifractal scaling behavior. A similar analysis carried out on the pOct.TL trace revealed that this trace is not multifractal.

In the case of the probability mass function (Figure 9), both the conservative cascade and the L-System models fitted very well the UA trace. The performance of the conservative cascade model was not so good in the case of the pOct.TL trace (Figure 6). This can be explained by the symmetry imposed on the generators W , which makes it difficult for conservative cascades to fit asymmetric probability distributions. In the case of the autocovariance function (Figure 7 and Figure 10), the fitting performance achieved by the models was good in the case of the UA trace, but clearly inferior for conservative cascades in the case of the pOct.TL trace.

To assess the queuing behavior the buffer size was varied from 100 Kbytes to 20 Mbytes. The service rate was 518 Kbytes/s for the pOct.TL trace (corresponding to an utilization of 0.7) and 512 Kbytes/s for the UA trace (corresponding to an utilization of 0.9). Figure 8 and Figure 11 show that, for both traces, the fitting of the queuing behavior was very good for the joint L-System model but significant differences occurred with the double conservative cascade model.

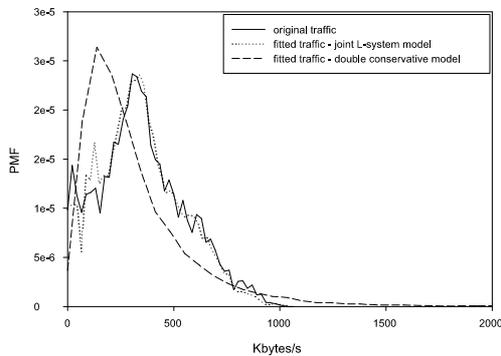


Figure 6. Probability mass function of bytes-per-time, trace pOct.TL.

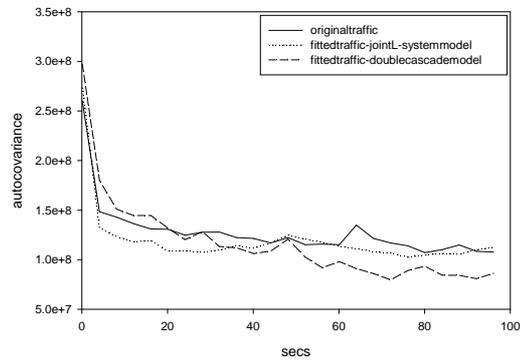


Figure 7. Autocovariance of bytes-per-time, trace pOct.TL.

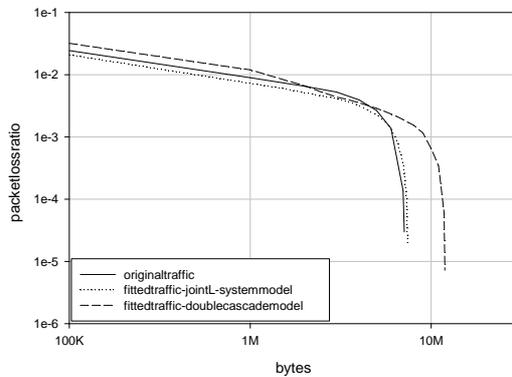


Figure 8. Packet loss ratio versus buffer size, trace pOct.TL.

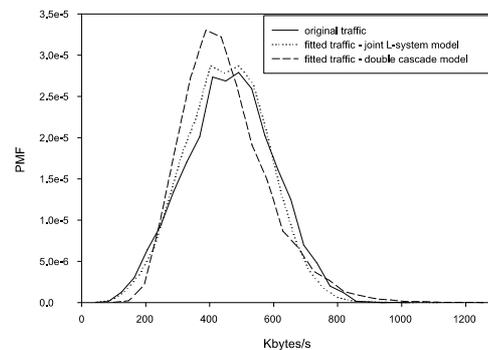


Figure 9. Probability mass function of bytes-per-time, trace UA.

It is interesting to note that, in the case of the UA trace, these differences occurred despite of a very good fitting in the first and second order statistics. This clearly illustrates that first and second order statistics are not sufficient to characterize multifractal traffic.

We also have analyzed a number of other traces. Our results show that, in general, joint L-Systems achieve an almost perfect matching of the queuing behavior, due to their ability to characterize correlations between packet arrivals and sizes. Joint L-Systems also show better performance than the conservative cascade model, which can be explained as follows. First, joint L-Systems allow the mass redistribution to depend on the mass itself, a feature that is present in real observed data. Second, the generator in conservative cascades is assumed to be symmetric which restricts the fitting of the probability mass function.

6. Conclusions

In this paper we proposed a new multifractal modeling approach, and an associated parameter fitting procedure, which characterizes the joint process of packet arrivals and packet sizes. The model is based on stochastic L-Systems, which were introduced by biologist A. Lindenmayer as a method to model plant growth. We work with a single L-System alphabet and production rule, where the alphabet is a set of pairs, and each pair element represents a packet arrival rate and a packet mean size. In this way, the traffic model is able to capture correlations between arrivals

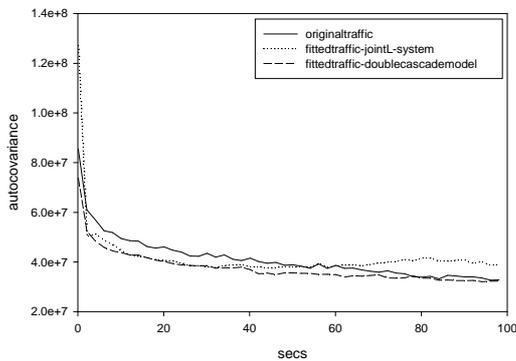


Figure 10. Autocovariance of bytes-per-time, trace UA.

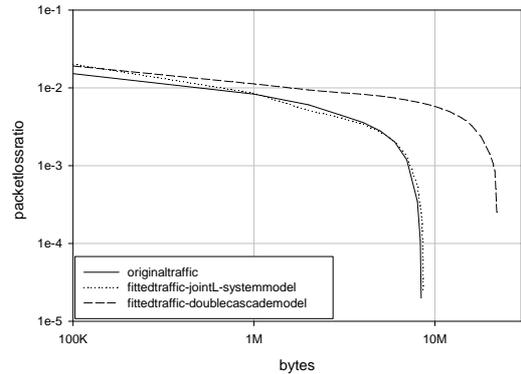


Figure 11. Packet loss ratio versus buffer size, trace UA.

and sizes, leading to an accurate prediction of the queuing behavior. We provided a detailed comparison with a related multifractal model based on conservative cascades. Our results, that include applying the fitting procedure to real observed data with multifractal scaling behavior on both the packet arrival and packet size processes, showed that our L-System based model can achieve excellent fitting performance in terms of first and second order statistics and queuing behavior.

REFERENCES

1. B. Mandelbrot, Intermittant turbulence in self-similar cascades: Divergence of high moments and dimensions of the carrier, *Journal of Fluid Mechanics* 62 (1974) 331–358.
2. R. Riedi, J. L. Véhel, Multifractal properties of TCP traffic: a numerical study, Technical Report No 3129, INRIA Rocquencourt, France Available at www.dsp.rice.edu/~riedi.
3. A. Feldmann, A. Gilbert, W. Willinger, Data networks as cascades: Investigating the multifractal nature of internet WAN traffic, in: *Proceedings of SIGCOMM, 1998*, pp. 42–55.
4. R. Riedi, M. Crouse, V. Ribeiro, R. Baraniuk, A multifractal wavelet model with application to network traffic, *IEEE Transactions on Information Theory* 45 (4) (1999) 992–1018.
5. P. Abry, P. Flandrin, M. Taqqu, D. Veitch, Wavelets for the analysis, estimation and synthesis of scaling data, in *Self-Similar Network Traffic Analysis and Performance Evaluation*, K. Park and W. Willinger Eds .
6. J. Gao, I. Rubin, Multifractal modeling of counting processes of long-range-dependent network traffic, *Journal of Computer Communications* 24 (2001) 1400–1410.
7. P. Salvador, A. Nogueira, R. Valadas, Modeling multifractal traffic with stochastic L-Systems, in: *Proceedings of GLOBECOM 2002, 2002*.
8. A. Lindenmayer, Mathematical models for cellular interactions in development II. Simple and branching filaments with two-sided inputs, *Journal of Theoretical Biology* 18 (1968) 300–315.
9. H. Peitgen, H. Jurgens, D. Saupe, *Chaos and Fractals: New Frontiers of Science*, Springer-Verlag, 1992.