Markovian Approach for Modeling IP Traffic Behavior on Several Time Scales

António Nogueira[†], Paulo Salvador[†], Rui Valadas[†], António Pacheco[‡]

[†]University of Aveiro / Institute of Telecommunications - Portugal
e-mail: nogueira@det.ua.pt; salvador@av.it.pt; rv@det.ua.pt
[‡]Instituto Superior Técnico - UTL / Department of Mathematics and CEMAT - Portugal
e-mail: apacheco@math.ist.utl.pt

ABSTRACT

In this paper we compare two traffic models based on Markov modulated Poisson processes (MMPPs), that were designed to capture self-similar behavior over multiple time scales. These models are both constructed by fitting the distribution of packet counts in a number of time scales. The first model is a superposition of MMPPs where each MMPP describes a different time scale. The second one is obtained as the equivalent to an hierarchical construction process that, starting at the coarsest time scale, successively decomposes MMPP states into new MMPPs to incorporate the characteristics offered by finner time scales. We evaluate the accuracy of the models by comparing the probability mass function at each time scale, as well as the loss probability and average waiting time in queue, corresponding to measured traces and to traces synthesized according to the proposed models. The analysis is based on three measured traffic traces exhibiting self-similar behavior: the well-known pOct Bellcore trace and two traces measured in a Portuguese ISP. Based on the obtained results, we conclude that both Markovian models have good and very similar performances in matching the characteristics of the data traces over the relevant time scales. However, one advantage of the hierarchical approach is that the number of states of the corresponding MMPP can be much smaller.

keywords: Traffic modeling, self-similar, time scale, Markov Modulated Poisson Process.

1. INTRODUCTION

The growing diversity of services and applications for IP networks is driving a strong requirement to make frequent measurements of packet flows and to describe them through appropriate traffic models. Since the work by Leland *et al.*[1] several studies have shown that network traffic may exhibit properties of self-similarity and/or long-range dependence (LRD),[1–5] which have significant impact on network performance. Self-similar traffic shows identical statistical characteristics over a wide range of time scales. In general, self-similarity implies long-range dependence, and vice-versa.

Several works have addressed the impact of LRD on network performance. Some of the works[5–8] study the case of a single queue and conclude that the buffer occupancy is not affected by autocovariance lags that are beyond the so-called critical time scale (CTS) or correlation horizon (CH), which depends on system parameters such as the buffer capacity. Similar conclusions are observed for the case of tandem queues.[9] Thus, matching the LRD is only required within the time scales specific to the system under study. One of the consequences of this result is that more traditional traffic models, such as Markov Modulated Poisson Processes (MMPPs), can still be used to model traffic exhibiting LRD. The use of MMPPs also benefits from the existence of several tools for calculating the queuing behavior and the effective bandwidth.

In this paper we describe and compare two traffic models based on MMPPs, together with their associated parameter fitting procedures, that are able to capture self-similar behavior over a range of time scales. We consider discrete-time MMMPs (dMMPPs) instead of continuous-time MMPPs, since they are more natural model for data corresponding to the number of arrivals in a sampling interval (the recording format that was used). Note that discrete-time and continuous-time MMPPs are basically interchangeable (through a simple parameter rescaling) as models for arrival processes, whenever the sampling interval used for the discrete-time version is small compared with the average sojourn times in the states of the modulating Markov chain. The first traffic model is constructed from the superposition of dMMPPs, where each dMMPP represents a specific time scale.[10] The parameter fitting procedure matches, at each time scale, a dMMPP to the empirical probability mass function characteristic of that time scale. The second traffic model is constructed using an hierarchical procedure: it starts at the largest time scale by inferring a dMMPP that matches the empirical PMF of this time

scale and then successively decomposes the dMMPP states into new dMMPPs, refining the traffic process by incorporating the characteristics offered by finer time scales.[11] In this way, a child dMMPP gives a more detailed description of its parent state PMF. The refinement process is iterated until a pre-defined number of time scales are integrated and, finally, a dMMPP incorporating this hierarchical structure is derived. The comparison of the fitting procedures is performed by applying them to several measured traffic traces that exhibit self-similar behavior: the well-known pOct Bellcore trace, a trace of aggregated IP WAN traffic, and a trace corresponding to the file sharing application Kazaa. We selected the Kazaa application given its present popularity in the Internet. We compare the probability mass function at each time scale, and the queuing behavior (as assessed by the loss probability and average waiting time in queue), corresponding to the measured and to synthetic traces generated from the inferred models.

Several fitting procedures have been proposed in the literature for estimating the parameters of MMPPs from empirical data, but they mainly concentrate on matching first- and/or second-order statistics, without addressing directly the issue of modeling on multiple time scales.[12–18] Yoshihara *et al.*[17] developed a fitting method for self-similar traffic based on the superposition of 2-MMPPs, that matches the variance at each time scale. In this way, the resulting MMPP reproduces the variance-scale curve characteristic of self-similar processes. Our proposals,[10,11] that we compare in this paper, are able to match the complete distribution at each time scale (and not only the variance) in order to reproduce accurately self-similar behavior.

The paper is organized as follows. Section 2 introduces the concept of self-similarity, motivates the need for a traffic model that matches the different time scales of the data and gives the required background on MMPPs. Section 3 presents the proposed traffic models and their associated parameter fitting procedures. Section 4 briefly describes the data traces used in the numerical evaluation and in Section 5 we discuss the obtained results. Finally, Section 6 presents the main conclusions.

2. SELF-SIMILARITY AND MARKOV MODULATED POISSON PROCESSES

The two inference procedures that will be compared are closely related to the notion of distributional self-similarity. Consider the continuous-time process $Y(t)$ representing the traffic volume (e.g. in bytes) from time 0 up to time t and let $X(t) = Y(t) - Y(t-1)$ be the corresponding increment process (e.g. in bytes/second). Consider also the sequence

$$X^{(m)}(k) = \frac{1}{m}\sum_{i=1}^{m} X((k-1)m+i), k = 1, 2, ... \tag{1}$$

obtained by averaging $X(t)$ over non-overlapping blocks of length m. $Y(t)$ is exactly self-similar when it is equivalent, in the sense of finite-dimensional distributions, to $a^{-H}Y(at)$, for all $t > 0$ and $a > 0$, where H $(0 < H < 1)$ is the Hurst parameter. Clearly, the process $Y(t)$ can not be stationary. However, if $Y(t)$ has stationary increments then again $X(k) = X^{(1)}(k)$ is equivalent, in the sense of finite-dimensional distributions, to $m^{1-H}X^{(m)}(k)$. This illustrates that a traffic model developed for fitting self-similar behavior must preferably enable the matching of the distribution on several time scales. Note also that, in general, self-similarity implies LRD, and vice-versa. An excellent overview of self-similarity and LRD can be found in Park and Willinger.[19]

An (homogeneous) Markov chain $(Y, J) = \{(Y_k, J_k),\ k = 0, 1, \ldots\}$ with state space $\mathbb{N}_0 \times S$ is a dMMPP if and only if Y has non-decreasing sample paths and

$$P(Y_{k+1} = m,\ J_{k+1} = j | Y_k = n,\ J_k = i) = p_{ij}\, e^{-\lambda_i}\lambda_i^{m-n}/(m-n)! \tag{2}$$

for $k = 0, 1, \ldots, m, n \in \mathbb{N}_0$ with $n \leq m$, and $i, j \in S$, where λ_i, $i \in S$ are nonnegative real constants and $\mathbf{P} = (p_{ij})$ is a stochastic matrix. In this case we say that (Y, J) is a dMMPP with set of modulating states S and parameter (matrices) $\mathbf{P}$ and $\mathbf{\Lambda}$, and write $(Y, J) \sim \text{dMMPP}_S(\mathbf{P}, \mathbf{\Lambda})$, where $\mathbf{\Lambda} = (\lambda_{ij}) = (\lambda_i \delta_{ij})$. The matrix $\mathbf{P}$ is the transition probability matrix of the modulating Markov chain J, whereas $\mathbf{\Lambda}$ is the matrix of Poisson arrival rates. If S has cardinality r, we say that (Y, J) is a dMMPP of order r (r-dMMPP). When, in particular, $S = \{1, 2, \ldots, r\}$ for some $r \in \mathbb{N}$, then we write simply that $(Y, J) \sim \text{dMMPP}_r(\mathbf{P}, \mathbf{\Lambda})$ and let $\pi = [\pi_1\, \pi_2, \ldots\, \pi_r]$ be the stationary distribution of J.

The superposition of independent dMMPPs is still an dMMPP. More precisely, if $(Y^{(l)}, J^{(l)}) \sim \text{dMMPP}_{r_l}(\mathbf{P}^{(l)}, \mathbf{\Lambda}^{(l)})$, $l = 1, 2, \ldots, L$, are independent, then their superposition $(Y, J) = (\sum_{l=1}^{L} Y^{(l)}, (J^{(1)}, J^{(2)}, \ldots, J^{(L)}))$ is a $\text{dMMPP}_S(\mathbf{P}, \mathbf{\Lambda})$ where $S = \{1, 2, \ldots, r_1\} \times \ldots \times \{1, 2, \ldots, r_L\}$,

$$\mathbf{P} = \mathbf{P}^{(1)} \otimes \mathbf{P}^{(2)} \otimes \ldots \otimes \mathbf{P}^{(L)} \qquad \text{and} \qquad \mathbf{\Lambda} = \mathbf{\Lambda}^{(1)} \oplus \mathbf{\Lambda}^{(2)} \oplus \ldots \oplus \mathbf{\Lambda}^{(L)} \tag{3}$$

with $\oplus$ and $\otimes$ denoting the Kronecker sum and product, respectively.

3. DESCRIPTION OF THE PROPOSED MARKOVIAN TRAFFIC MODELS

The proposed dMMPP models are able to capture self-similar behavior over a range of time scales. In order to incorporate the traffic characteristics over several time scales, we work with the probability mass function (PMF) of the number of arrivals at each time scale. The number of time scales, L, is fixed *a priori* and the time scales are numbered in an increasing way, from $l = 1$ (corresponding to the largest time scale) to $l = L$ (corresponding to the smallest time scale).

The difference between the two proposals lies essentially in the construction procedure of the equivalent dMMPP. The first traffic model is based on the superposition of dMMPPs, where each dMMPP represents a specific time scale, and will be designated by *superposition* model. Figure 1 illustrates the construction methodology of the dMMPP for the simple case of three time scales and two-state dMMPPs in each time scale. The dMMPP associated with time scale l is denoted by $\text{dMMPP}^{(l)}$ and the corresponding number of states by $N_{(l)}$.

The flowchart of the inference method is represented in Figure 3 where, basically, four steps can be identified:
(i) compute the data vectors (corresponding to the average number of arrivals per time interval) at each time scale;
(ii) calculate the empirical PMF at the largest time scale and infer the associated dMMPP;
(iii) for all other time scales (going from the largest to the smallest one), calculate the empirical PMF, deconvolve it from the empirical PMF of the previous time scale and infer a dMMPP that matches the resulting PMF;
(iv) calculate the final dMMPP through superposition of the dMMPPs inferred for each time scale.

The second traffic model is constructed using an hierarchical procedure, that successively decomposes dMMPP states into new dMMPPs, thus refining the traffic process by incorporating the characteristics offered by finer time scales. We start at the largest time scale by inferring a dMMPP that matches the PMF of this time scale. As part of the parameter fitting procedure, each time interval of the data sequence is assigned to a dMMPP state; in this way, a new PMF can be associated with each dMMPP state. At the next finer time scale, each dMMPP state is decomposed into a new dMMPP that matches the contribution of this time scale to the PMF of the state it descends from. In this way, a child dMMPP gives a more detailed description of its parent state PMF. This refinement process is iterated until a pre-defined number of time scales are integrated. Finally, a dMMPP incorporating this hierarchical structure is derived. This traffic model will be designated by *hierarchical* model.

The construction process of the hierarchical model can be described through a tree where, except for the root node, each tree node corresponds to a dMMPP state and each tree level to a time scale (Figure 2). A dMMPP state will be represented by a vector indicating the path in the tree from its higher level ancestor (i.e. the state it descends from at the largest scale, $l = 1$) to itself. Thus, a state at time scale l will be represented by some vector $\vec{s} = (s_1, s_2, ..., s_l)\,, s_i \in I\!N$. Each dMMPP will be represented by the state that generated it (i.e. its parent state). We let $\text{dMMPP}^{\vec{s}}$ denote the dMMPP generated by state $\vec{s}$ and $\{1, 2, \ldots, N_{\vec{s}}\}$ the corresponding states, where $N_{\vec{s}}$ is the number of states. The root node of the tree corresponds to a virtual state, denoted by $\vec{s} = \emptyset$, that is used to represent the dMMPP of the largest time scale, $l = 1$. This dMMPP will be called the root dMMPP. Thus, the dMMPP states in the tree are characterized by

$$\vec{s} = (s_1, s_2, ..., s_l)\,, l \in I\!N \tag{4}$$

with $s_{i+1} \in \left\{1, 2, \ldots, N_{\vec{s}_{i]}}\right\}, i = 0, 1, \ldots, l-1$; here, $\vec{s}_{j]}$ denotes the sub-vector of $\vec{s}$ given by $(s_1, s_2, ..., s_j)$, with $j < |\vec{s}|$, and $\vec{s}_{0]} = \emptyset$, where $|\vec{s}|$ denotes the length of vector $\vec{s}$. Note that, using this notation, a vector $\vec{s}$ can either represent state $\vec{s}$ or the dMMPP generated by $\vec{s}$. Also, the time scale of $\text{dMMPP}^{\vec{s}}$ is $|\vec{s}| + 1$.

Finally, let $E^{\vec{s}}$ denote the set of time intervals associated with state $\vec{s}$, i.e., with $\text{dMMPP}^{\vec{s}}$. Using this notation, the set associated with $\text{dMMPP}^{\emptyset}$ will be $E^{\emptyset} = \{1, 2, ..., N\}$, where N is the number of time intervals at the smallest time scale. Starting from $E^{\emptyset}$, the sets $E^{\vec{s}}$ are successively partitioned at each time scale in a hierarchical fashion. Thus, if states $\vec{s}$ and

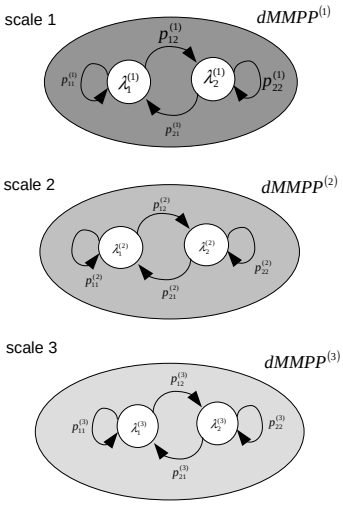

Figure 1. Construction methodology of the superposition dMMPP model.

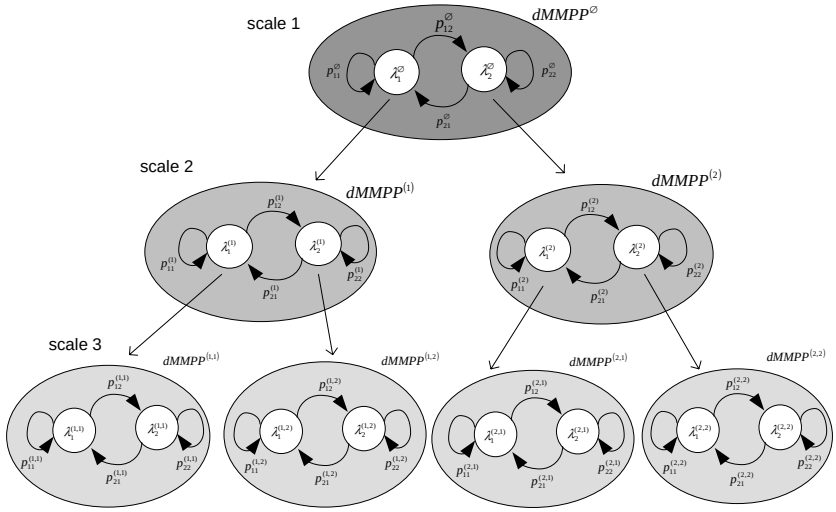

Figure 2. Construction methodology of the hierarchical dMMPP model.

$\vec{t}$ are such that $|\vec{s}| = \left|\vec{t}\right| = l$ and $\vec{s} \neq \vec{t}$, then $E^{\vec{s}} \cap E^{\vec{t}} = \emptyset$ and $\bigcup_{\vec{s}:|\vec{s}|=l} E^{\vec{s}} = E^{\emptyset}$. Moreover, if state $\vec{s}$ is a parent of state $\vec{t}$, that is $\vec{t} = (\vec{s}, j)$, then $E^{\vec{t}} \subseteq E^{\vec{s}}$ and $\bigcup_{j=1,\ldots,N_{\vec{s}}} E^{(\vec{s},j)} = E^{\vec{s}}$.

The inference procedure is represented schematically in the flowchart of Figure 4, where the following main steps can be identified:

(i) calculation of the data sequences (corresponding to the average number of arrivals per time interval) for each time scale, starting with the smallest one and going through an aggregation process up to the largest one.

(ii) inference of the dMMPP at the largest time scale, $l = 1$, that matches the empirical PMF at this time scale.

(iii) for all other time scales, in increasing order, $l = 2, ..., L-1$, and for each parent dMMPP state, identification of the time intervals assigned to the state, calculation of the corresponding PMF and inference of the dMMPP that matches the contribution of the time scale to the state PMF;

(iv) finally, calculation of matrices $\mathbf{\Lambda}$ and $\mathbf{P}$ of the dMMPP incorporating the previous hierarchical structure.

Note that in both traffic models the number of states of all dMMPPs are computed as part of the fitting procedure. The two inference procedures have some common steps. We will now describe in detail the common and the different steps of the inference methods. The main difference regarding the construction of the two models is that in the superposition model there is one dMMPP per time scale whereas in the hierarchical model there are several dMMPPs per time scale, one per state of the immediately higher time scale.

3.1. Aggregation process

This is a preliminary step in both inference procedures. Having defined the time interval at the smallest time scale, Δt, the number of time scales, L, and the level of aggregation, a, the aggregation process starts by computing the data sequence corresponding to the average number of arrivals in the smallest time scale, $D^{(L)}(k), k = 1, 2, \ldots, N$. Then, it calculates the data sequences of the remaining time scales, $D^{(l)}(k), l = L-1, ..., 1$, corresponding to the average number of arrivals in intervals of length $\Delta t a^{(L-l)}$. This is given by

$$D^{(l)}(k) = \begin{cases} \Psi\left(\frac{1}{a}\sum_{i=0}^{a-1} D^{(l+1)}(k + ia^{L-l-1})\right), & \frac{k-1}{a^{L-l}} \in I\!N_0 \\ D^{(l)}(k-1), & \frac{k-1}{a^{L-l}} \notin I\!N_0 \end{cases} \tag{5}$$

where $\Psi(x)$ represents round toward the integer nearest x. Note that the block length of equation (1) is related with a and l by $m = a^{L-l}$. Note also that all data sequences have the same length N and that $D^{(l)}(k)$ is formed by sub-sequences of

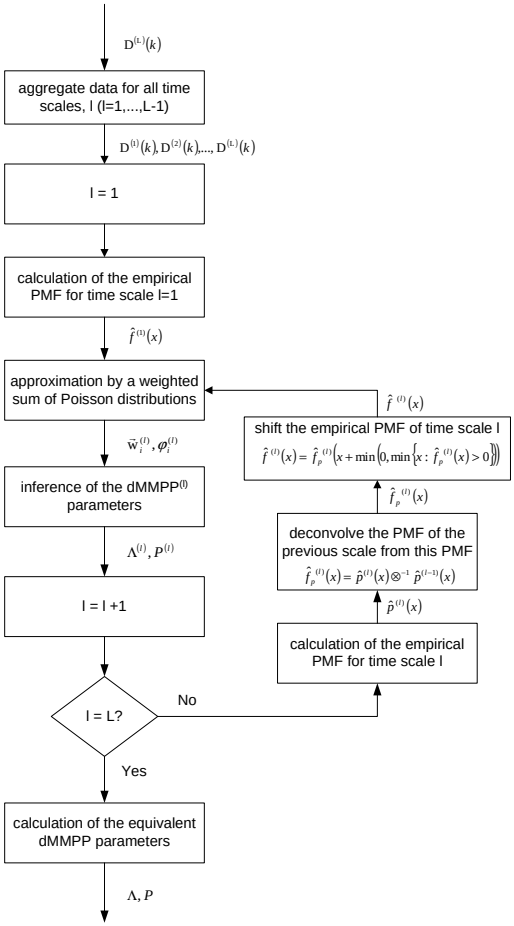

Figure 3. Flow diagram of the inference procedure of the superposition model.

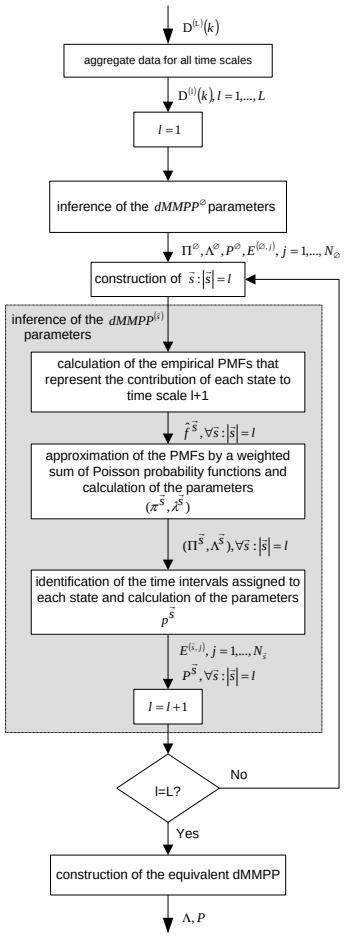

Figure 4. Flow diagram of the inference procedure of the hierarchical model.

a^{L-l} successive equal values; these sub-sequences will be called *l-sequences*. The empirical distribution of $D^{(l)}(k)$ will be denoted by $\hat{p}^{(l)}(x)$.

3.2. Inference of the partial dMMPPs

In both proposed traffic models, all partial dMMPPs are inferred in order to fit an empirical PMF. For the largest time scale, it is the PMF of the most aggregated data sequence, $D^{(1)}(k)$. For all other time scales l, $l = 2, ..., L$, the associated dMMPPs will model only the traffic components due to that scale, but at this point both models diverge in the way partial dMMPPs are calculated. In the superposition model, there will be only one dMMPP per time scale, inferred according to the flowchart of Figure 3. In the hierarchical model, a dMMPP will be inferred for each state of the immediately higher time scale. Thus, for each dMMPP and time scale the matched PMF represents the contribution of the time scale to the PMF of its parent state. The parameter fitting procedure of each tree dMMPP comprises several steps, highlighted in the flowchart of Figure 4 and explained in more detail in the next sub-sections.

3.2.1. Calculation of the empirical PMFs

The operations involved in this step are basically the same for both proposed traffic models, but the number of dMMPPs per time scale (and of PMFs) are different in both cases. Each dMMPP will be inferred from a PMF that represents its contribution to a particular time scale. For the largest time scale, this PMF is simply the empirical one for both traffic models. For the other time scales l, $l = 2, ..., L$, the PMFs calculated for each model are different.

In the superposition model, the traffic components due to time scale l, $l = 2, ..., L$, are obtained through deconvolution of the empirical PMFs of this and the previous time scales, i.e., $\hat{f}_p^{(l)}(x) = [\hat{p}^{(l)} \otimes^{-1} \hat{p}^{(l-1)}](x)$. However, this may result in probability mass at negative arrival rates for the dMMPP$^{(l)}$, which will occur whenever $\min\left\{x : \hat{p}^{(l-1)}(x) > 0\right\} < \min\left\{x : \hat{p}^{(l)}(x) > 0\right\}$.

In the hierarchical model, the contribution of a dMMPP at time scale l generated from state $\vec{s}$ corresponds also to the deconvolution of empirical PMFs, but now calculated over the set of time intervals $E^{\vec{s}}$, at this time scale $l = |\vec{s}| + 1$ and previous time scale $l - 1 = |\vec{s}|$, i.e., $\hat{f}_p^{\vec{s}}(x) = \left[\hat{p}^{\vec{s},|\vec{s}|+1} \otimes^{-1} \hat{p}^{\vec{s},|\vec{s}|}\right](x)$, where $\hat{p}^{\vec{s},l}$ represents the PMF obtained from the data sequence $D^l(k), k \in E^{\vec{s}}$. Note that the two empirical PMFs are obtained from the same set of time intervals but aggregated at different levels. Once again, these operations may result in probability mass at negative arrival rates for the dMMPP$^{\vec{s}}$, which will occur whenever $\min\left\{x : \hat{p}^{\vec{s},|\vec{s}|}(x) > 0\right\} < \min\left\{x : \hat{p}^{\vec{s},|\vec{s}|+1}(x) > 0\right\}$.

To correct these results, the dMMPP$^{\chi}$ will be fitted to

$$\hat{f}^{\chi}(x) = \hat{f}_p^{\chi}(x + e^{\chi}) \tag{6}$$

where $e^{\chi} = \min\left(0, \min\left\{x : \hat{f}_p^{\chi}(x) > 0\right\}\right)$, which assures $\hat{f}^{\chi}(x) = 0$, $x < 0$, considering that superscript χ replaces (l) in the superposition model description and $\vec{s}$ in the hierarchical model description. The additional factors that now introduced are removed in the final step of the inference procedure.

3.2.2. Inference of the parameters

In the next paragraphs we will maintain the notation introduced in the last sub-section, that is, superscript χ will denote (l) or $\vec{s}$, depending on whether we consider the superposition or the hierarchical traffic model.

The first step in the inference of the dMMPP$^{\chi}$ parameters is the approximation of $\hat{f}^{\chi}$ by a weighted sum of Poisson probability functions. This is based on an algorithm that progressively subtracts a Poisson probability function from $\hat{f}^{\chi}$.[18] The most important steps of this algorithm will be explained in the next paragraphs.

Let us represent the n^{th} Poisson probability function, with mean φ_n^{χ}, by $g_{\varphi_n^{\chi}}(x)$ and define $h_n^{\chi}(x)$ as the difference between $\hat{f}^{\chi}(x)$ and the weighted sum of Poisson probability functions at the n^{th} iteration. Initially, we set $h_1^{\chi}(x) = \hat{f}^{\chi}(x)$ and, in each step, we first detect the maximum of $h_n^{\chi}(x)$. The corresponding x-value, $\varphi_n = \arg\max_x h_n^{\vec{s}}(x)$, will be considered the n^{th} Poisson rate of the dMMPP$^{\chi}$. We then calculate the weights of each Poisson probability function, $\vec{w}_n^{\chi} = [w_{1n}^{\chi}, w_{2n}^{\chi}, ..., w_{nn}^{\chi}]$, through the following set of linear equations:

$$\hat{f}^{\chi}(\varphi_m^{\chi}) = \sum_{j=1}^{n} w_{jn}^{\chi} g_{\varphi_j^{\chi}}(\varphi_m^{\chi}) \tag{7}$$

for $m = 1, ..., n$. This assures that the fitting between $\hat{f}^{\chi}(x)$ and the weighted sum of Poisson probability functions is exact at φ_m^{χ} points, for $m = 1, 2, \ldots, n$. The final step in each iteration is the calculation of the new difference function

$$h_{n+1}^{\chi}(x) = \hat{f}^{\chi}(x) - \sum_{j=1}^{n} w_{jn}^{\chi} g_{\varphi_j^{\chi}}(x). \tag{8}$$

The algorithm stops when the maximum of $h_n^{\chi}(x)$ is lower than a pre-defined percentage of the maximum of $\hat{f}^{\chi}(x)$. At this point, the number of states of the dMMPP$^{\chi}$, N_{χ}, is made equal to n.

After N_{χ} has been determined, the parameters of the dMMPP$^{\chi}$, $\{(\pi_j^{\chi}, \lambda_j^{\chi}), j = 1, 2, \ldots, N_{\chi}\}$, are set equal to

$$\pi_j^{\chi} = w_{jN_{\chi}}^{\chi} \qquad \text{and} \qquad \lambda_j^{\chi} = \varphi_j^{\chi}. \tag{9}$$

Note that the number of states of each dMMPP depends on the level of accuracy employed in the approximation of f^{χ} by the weighted sum of Poisson probability functions.

The next step of the parameter inference procedure is to associate, at each time scale l, one of the dMMPP$^{\chi}$ states with each time interval of the arriving process. Recall that the data sequences aggregated at time scale l have a^{L-l} successive equal values called l-sequences. For the hierarchical model, the set of time intervals associated with dMMPP$^{\vec{s}}$ is $E^{\vec{s}}$ and the goal here is to partition $E^{\vec{s}}$ into subsets $E^{(\vec{s},j)}, j = 1, ..., N_{\vec{s}}$. The state assignment process considers only the first time interval of each l-sequence, defined by $i = a^{L-\chi'}(k-1)+1, k \in I\!N, i \in E^{\chi}$, where χ' equals $(|\chi|+1)$ for the hierarchical model and l for the superposition model and E^{χ} represents the set of time intervals associated with dMMPP$^{\chi}$. The state that is assigned to l-sequence i is calculated randomly according to the probability vector $\vec{\theta}^{\chi}(i) = \left\{\theta_1^{\chi}(i), \ldots, \theta_{N_{\chi}}^{\chi}(i)\right\}$, with

$$\theta_n^{\chi}(i) = \frac{g_{\lambda_n^{\chi}}\left(D^{\chi'}(i)\right)}{\sum_{j=1}^{N_{\chi}} g_{\lambda_j^{\chi}}\left(D^{\chi'}(i)\right)} \tag{10}$$

for $n = 1, ..., N_{\chi}$, where λ_j^{χ} represents the Poisson arrival rate of the j^{th} state of dMMPP$^{\chi}$, and $g_{\lambda}(y)$ represents a Poisson probability distribution function with mean λ. The elements of this vector represent the probability that the state j had originated the number of arrivals $D^{(l)}(k)$ at time interval k from time scale l.

After this step, we infer the dMMPP$^{\chi}$ transition probabilities, p_{od}^{χ}, with $o, d = 1, ..., N_{\chi}$, counting the number of transitions between each pair of states. If n_{od}^{χ} represents the number of transitions from state o to state d of the dMPPP$^{\chi}$, then

$$p_{od}^{\chi} = \frac{n_{od}^{\chi}}{\sum_{m=1}^{N_{\chi}} n_{om}^{\chi}}, o, d = 1, ..., N_{\chi} \tag{11}$$

The transition probability and the Poisson arrival rate matrices of the dMMPP$^{\chi}$ are then given by

$$\mathbf{P}^{\chi} = \begin{bmatrix} p_{11}^{\chi} & p_{12}^{\chi} & \cdots & p_{1N_{\chi}}^{\chi} \\ p_{21}^{\chi} & p_{22}^{\chi} & \cdots & p_{2N_{\chi}}^{\chi} \\ \cdots & \cdots & \cdots & \cdots \\ p_{N_{\chi}1}^{\chi} & p_{N_{\chi}2}^{\chi} & \cdots & p_{N_{\chi}N_{\chi}}^{\chi} \end{bmatrix} \quad \text{and} \quad \mathbf{\Lambda}^{\chi} = \begin{bmatrix} \lambda_1^{\chi} & 0 & \ldots & 0 \\ 0 & \lambda_2^{\chi} & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & \ldots & \lambda_{N_{\chi}}^{\chi} \end{bmatrix} + e^{\chi}\mathbf{I} \tag{12}$$

The diagonal matrix of the steady-state probabilities will be designated by $\mathbf{\Pi}^{\chi}$.

3.3. Construction of the equivalent dMMPP model

Due to the construction procedure inherent to each proposed traffic model, this step in completely different for both approaches. Taking the superposition model, for example, the equivalent dMMPP process is constructed using equation (3), where matrices $\mathbf{P}^{(l)}$ and $\mathbf{\Lambda}^{(l)}$, $l = 1, ..., L$, were calculated in the last subsection. However, the additional factors introduced in sub-section 3.2.1 must be removed. Thus, the final $\mathbf{\Lambda}^{(l)}$ will be given by

$$\mathbf{\Lambda} = \mathbf{\Lambda} - \sum_{l=2}^{L} e^{(l)} \cdot \mathbf{I} \tag{13}$$

where $\mathbf{I}$ is the identity matrix.

For the hierarchical model, we have to construct a dMMPP equivalent to the tree structure of dMMPPs derived in previous sections. The goal is to incorporate in the model the level of detail given by the finer time scale, so the equivalent dMMPP will have a number of states equal to the number of states in smallest time scale of the tree structure, L. These can be identified by $\vec{s} = (s_1, s_2, ..., s_L)$; each state is associated with its ancestor states $\vec{s}_{i+1]} = (s_1, s_2, ..., s_{i+1})$, $i = 0, 1, \ldots, L-1$ of the dMMPP$^{\vec{s}_{i]}}$.

Thus, the states of the equivalent dMMPP will have Poisson rates which are the sum of the Poisson rates of its ancestors in the tree structure, i.e.,

$$\lambda_{\vec{s}} = \sum_{j=0}^{L-1} \lambda_{s_{j+1}}^{\vec{s}_{j]}} \tag{14}$$

The transition between each pair of states is determined by the shortest path in the tree structure, going through the root dMMPP, that joins the two states. Any pair of states descend from one or more common dMMPPs. The first one, at the time scale with higher l, will be denoted by $\vec{s} \wedge \vec{t} = (s_1, s_2, ..., s_k)$ where $k = \max\{i : s_j = t_j, j = 1, 2, ..., i\}$.

We first consider the case of $\vec{s} \neq \vec{t}$. The probability of transition from $\vec{s}$ to $\vec{t}$, $p_{\vec{s},\vec{t}}$, is given by the product of three factors. The first factor accounts for the time scales where $\vec{s}$ and $\vec{t}$ have the same associated states and is given by

$$\phi_{\vec{s},\vec{t}} = \begin{cases} \prod_{j=0}^{|\vec{s}\wedge\vec{t}|-1} p_{s_{j+1},s_{j+1}}^{\vec{s}_{j]}}, |\vec{s} \wedge \vec{t}| \neq 0 \\ 1, |\vec{s} \wedge \vec{t}| = 0 \end{cases} \tag{15}$$

The second factor accounts for the transition in the time scale where $\vec{s}$ and $\vec{t}$ are associated to different states of the same dMMPP, which corresponds to $p^{\vec{s}\wedge\vec{t}}_{s_{|\vec{s}\wedge\vec{t}|+1},t_{|\vec{s}\wedge\vec{t}|+1}}$. The third factor accounts for the steady-state probabilities of states associated to $\vec{t}$ in the time scales that are not common to $\vec{s}$ and is given by

$$\psi_{\vec{s},\vec{t}} = \prod_{j=|\vec{s}\wedge\vec{t}|+1}^{L-1} \pi_{t_{j+1}}^{\vec{t}_{j]}} \tag{16}$$

where an empty product is equal to one.

Finally, for $\vec{s} \neq \vec{t}$,

$$p_{\vec{s},\vec{t}} = \phi_{\vec{s},\vec{t}} p^{\vec{s}\wedge\vec{t}}_{s_{|\vec{s}\wedge\vec{t}|+1},t_{\vec{s}\wedge\vec{t}+1}} \psi_{\vec{s},\vec{t}} \tag{17}$$

In case $\vec{s} = \vec{t}$, it is simply

$$p_{\vec{s},\vec{t}} = \phi_{\vec{s},\vec{t}} \tag{18}$$

4. OVERVIEW OF THE TRAFFIC TRACES

Two traces of aggregated IP traffic were selected to test the accuracy and compare both fitting procedures: (i) the well known and publicly available Bellcore pOct LAN trace[1] and (ii) a trace measured at the backbone of a Portuguese ISP ADSL network, characterizing the downstream Internet access traffic corresponding to approximately 65 simultaneous users. A third trace was also considered, including the downstream traffic from 10 users of the file sharing application Kazaa, a protocol running over TCP. This trace was measured at the premises of the same Portuguese ISP and its inclusion is due to the fact that an increasing percentage of the overall Internet traffic belongs to peer-to-peer protocols of the same type as Kazaa. For all our measurements, the traffic analyzer was a 1.2 GHz AMD Athlon PC, with 1.5 Gbytes of RAM and running WinDump, and recorded the arrival instant and the IP header of each packet. The main characteristics of all selected traces are described in Table 1.

All traces exhibit self-similar characteristics: taking trace Kazaa, for example, the analysis of its autocovariance function (Figure 5) lead us to suspect that it exhibits LRD behavior, due to the slow decay for large time lags. This is confirmed by the scaling analysis[20] (Figure 6), that looks for alignment in the so-called Logscale Diagram (LD), which is a log-log plot of the variance estimates of discrete wavelet transform coefficients, against scale, completed with confidence intervals about these estimates at each scale. Traffic is said to be LRD if, within the limits of the confidence intervals, the log of the variance estimates fall on a straight line, in a range of scales from some initial value j_1 (octave 8 in this case) up to the largest one present in data (octave 12). A similar analysis was made for the other traces, also revealing the same LRD behavior.

Trace name	Capture period	Trace size (pkts)	Mean rate (byte/s)	Mean pkt size (bytes)
pOct	Bellcore trace	0.5 million	322790	568
ISP	10.26pm to 10.49pm, October 18^{th}2002	0.5 million	583470	797
Kazaa	10.26pm to 11.31pm, October 18^{th}2002	0.5 million	131140	1029

Table 1. Main characteristics of measured traces.

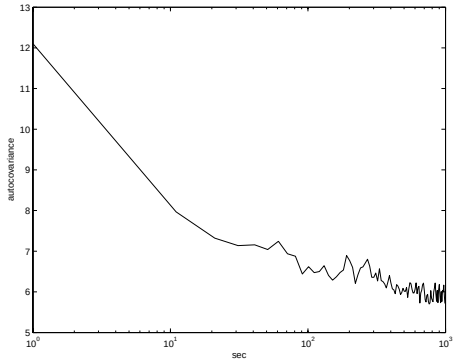

Figure 5. Autocovariance of packet counts, trace Kazaa.

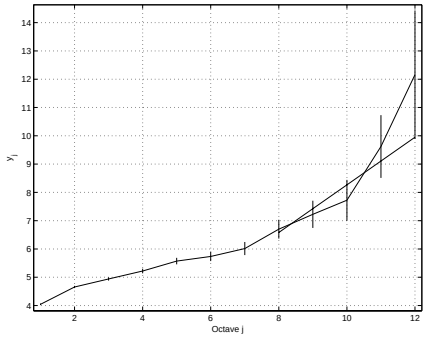

Figure 6. Second order Logscale Diagram, trace Kazaa.

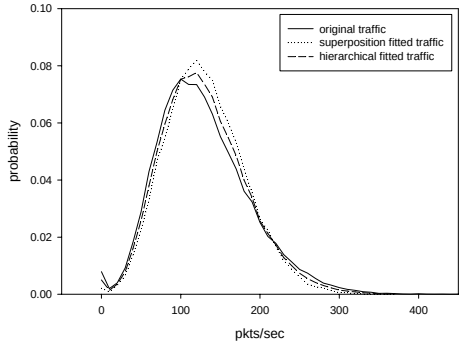

Figure 7. Probability mass function at the smallest time scale, trace Kazaa.

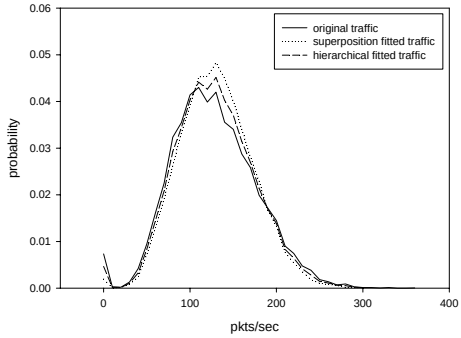

Figure 8. Probability mass function at the intermediate time scale, trace Kazaa.

5. NUMERICAL RESULTS

We assess the accuracy and compare the dMMPP fitting procedures using several criteria: (i) comparing the Hurst parameters of the original and synthesized (according to the parameters inferred for the resulting dMMPPs) data traces; (ii) comparing the PMFs of the average number of arrivals in different time scales, obtained for the original and synthesized traces and (iii) comparing the queuing behavior, in terms of packet loss ratio (PLR) and average waiting time in queue (AWT), through a trace-driven simulation using those traces as inputs. All simulations were carried out using a fixed packet length equal to the mean packet length of the trace. For all traces, the sampling interval of the counting process was chosen to be 0.1s and three different time scales were considered: 0.1s, 0.2s and 0.4s. Larger aggregation levels were also considered, with good fitting results. For each trace, the estimation procedure took less than 1 minute for the superposition model and 2 minutes for the hierarchical model, using a MATLAB implementation running in the PC described above, which shows that both procedures are computationally very efficient.

In order to verify that both fitting approaches capture the traffic LRD behavior, we compare in Table 1 the Hurst parameters estimated (using the LD estimator) for the original and fitted traffic, for both traffic models and for each one of the three selected data traces. Table 1 also includes the range of time scales where the wavelet coefficients follow a straight line, written in parenthesis near to the corresponding Hurst parameter value. As we can see, there is a very good agreement between the Hurst parameter values of the original and fitted (according to both inference procedures) traffic, so LRD behavior seems to be well captured by the fitting approaches.

The next evaluation criteria is based on the comparison between the PMFs of the original and dMMPP fitted traces, for different time scales. Considering trace Kazaa, we can see from figures 7, 8 and 9 that there is a good agreement between all the PMFs corresponding to the original and dMMPP fitted traces, for the smallest, intermediate and largest time scales. This is achieved with the resulting dMMPPs having about 288 states in the superposition model and 38 states in the hierarchical model. For traces pOct and ISP similar analysis were made, also revealing very good agreements between the

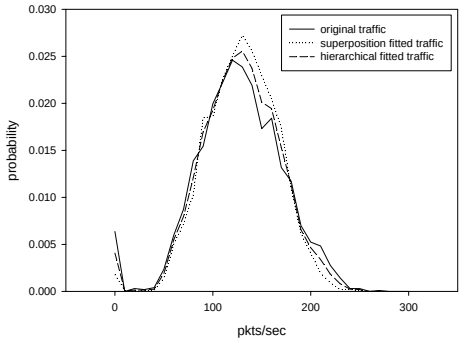

Figure 9. Probability mass function at the largest time s
trace Kazaa.

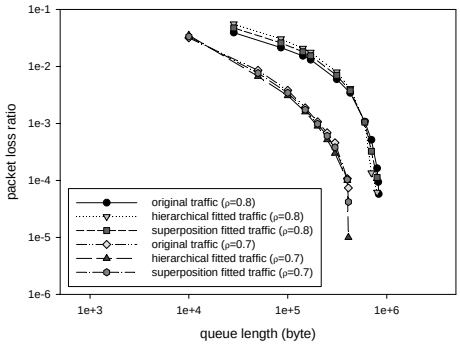

Figure 10. Packet loss ratio, trace pOct.

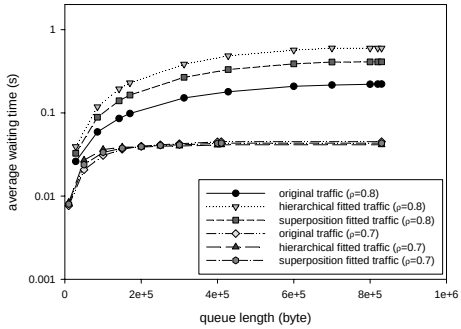

Figure 11. Average waiting time in queue, trace pOct.

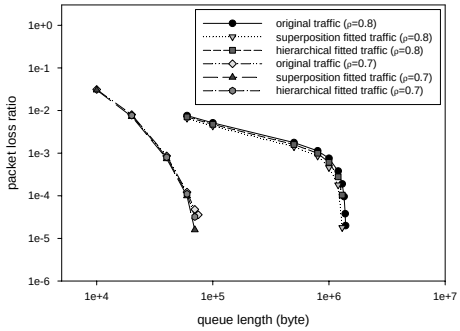

Figure 12. Packet loss ratio, trace Kazaa.

PMFs of the original and dMMPP fitted traces, for both traffic models and all the considered time scales. The performance of both models is very similar but the resulting number of states is much higher in the superposition model case: for trace pOct the number of states was equal to 344 for the superposition model and 81 for the hierarchical model, whereas for trace ISP the inferred dMMPPs had 245 states using the superposition approach and 74 sates using the hierarchical approach.

We now verify if the close match obtained in the Hurst parameter values and in the PMFs at each time scale is enough to guarantee a similar queuing behavior between the original traffic and the fitted models. Considering queuing performance, for each selected trace we compare the PLR and AWT values obtained through a trace-driven simulation of the original and dMMPP fitted traces. Two different sets of utilization ratios were used in the simulations: for traces pOct and Kazaa, we used $\rho = 0.7$ and $\rho = 0.8$ and for trace ISP the selected values were $\rho = 0.8$ and $\rho = 0.9$. This is due to the lower burstiness of the ISP traffic, which leads to lower packet losses for the same link utilization. From figures 10 and 11 it is possible to see that, for trace pOct, PLR behavior is very well approximated by the equivalent dMMPPs for both utilization ratios, while the agreement of the AWT curves is less accurate specially for higher utilization ratios. For trace Kazaa the results are depicted in figures 12 and 13 and for trace ISP the results are illustrated in figures 14 and 15. For both traces, the agreement between the PLR curves corresponding to the original and fitted traces is good. However, as the utilization ratio increases the deviation slightly increases, because the sensitivity of the metrics variation to a slight difference in the compared traces is higher. Regarding AWT, the agreement between the curves corresponding to the original and fitted

Trace	original	superposition fitted	hierarchical fitted
pOct	0.846 (4,11)	0.859 (4,11)	0.851 (4,11)
ISP	0.954 (4,10)	0.956 (4,10)	0.952 (5,10)
Kazaa	0.917 (8,12)	0.897 (6,12)	0.901 (6,12)

Table 2. Comparison between Hurst parameter values

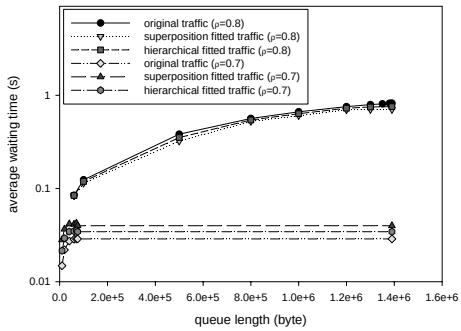

Figure 13. Average waiting time in queue, trace Kazaa.

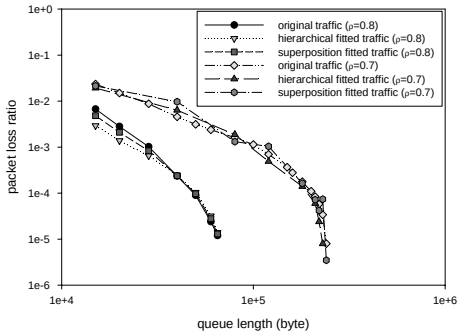

Figure 14. Packet loss ratio, trace ISP.

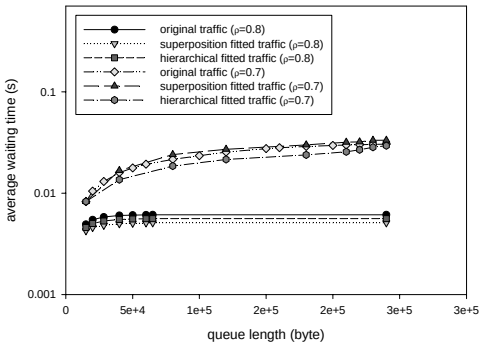

Figure 15. Average waiting time in queue, trace ISP.

traces is also good, for both utilization ratios.

Thus, we conclude that the proposed fitting approaches provide a close match of the Hurst parameters and PMFs at each time scale, and this agreement reveals itself sufficient to drive a good queuing performance in terms of packet loss ratio and average waiting time in queue. Moreover, the computational complexity of both fitting methods is small. Note that this complexity, as well as the number of states of the resulting dMMPPs, is directly related to the level of accuracy used to approximate the empirical PMFs at each time scale by weighted sums of Poisson probability functions

The performance of both inference procedures is very similar. Thus, it is not easy to recommend one of approaches over the other based solely on their associated performances. One argument that clearly favors the hierarchical approach is that the numbers of states of the resulting dMMPPs are smaller than the corresponding numbers for the superposition approach. This may be due to the fact that in the hierarchical approach and as the time scale increases, dMMPPs are fitted to successively smaller sets of intervals whose arrivals characteristics tend to increase in homogeneity and, thus, tend to have associated a smaller number of states than the dMMPP fitted through the superposition approach for the same time scale. However, the contribution of each time scale for the characterization of the aggregate traffic characteristics is interpreted in an easier and more natural way through the superposition approach. Note also that, for the same number of states, a smaller number of dMMPPs and corresponding parameters tend to be needed to compute the final dMMPP using the superposition approach than using the hierarchical approach.

6. CONCLUSIONS

In this paper we compared two traffic models based on Markov Modulated Poisson Processes (MMPPs), which are able to capture self-similarity over a range of time scales. It is known that this characteristic has significant impact on network performance; therefore, it must be taken into account for an accurate prediction of network performance. The first model is a superposition of MMPPs, where each MMPP describes a different time scale, and the second one is obtained as the equivalent to an hierarchical construction process that, starting at the coarsest time scale, successively decomposes MMPP states into new MMPPs in order to incorporate the characteristics offered by finner time scales. The accuracy of the models

was evaluated by comparing the probability mass function at each time scale, as well as the loss probability and average waiting time in queue, corresponding to measured traces and to traces synthesized according to the proposed models. The analysis was based on three measured traffic traces exhibiting self-similar behavior and the results show that both MMPPs have good and very similar performances in matching the characteristics of the data traces over the time scales of interest. However, one advantage of the hierarchical approach is that the number of states of the corresponding MMPP can be much smaller.

Acknowledgements: This research was supported in part by Fundação para a Ciência e a Tecnologia, the project POSI/42069/CPS/2001, and the grant BD/19781/99.

REFERENCES

1. W. Leland, M. Taqqu, W. Willinger, and D. Wilson, "On the self-similar nature of Ethernet traffic (extended version)," *IEEE/ACM Transactions on Networking* **2**, pp. 1–15, Feb. 1994.
2. J. Beran, R. Sherman, M. Taqqu, and W. Willinger, "Long-range dependence in variable-bit rate video traffic," *IEEE Transactions on Communications* **43**(2/3/4), pp. 1566–1579, 1995.
3. M. Crovella and A. Bestavros, "Self-similarity in World Wide Web traffic: Evidence and possible causes," *IEEE/ACM Transactions on Networking* **5**, pp. 835–846, Dec. 1997.
4. V. Paxson and S. Floyd, "Wide-area traffic: The failure of Poisson modeling," *IEEE/ACM Transactions on Networking* **3**, pp. 226–244, June 1995.
5. B. Ryu and A. Elwalid, "The importance of long-range dependence of VBR video traffic in ATM traffic engineering: Myths and realities," *ACM Computer Communication Review* **26**, pp. 3–14, Oct. 1996.
6. D. Heyman and T. Lakshman, "What are the implications of long range dependence for VBR video traffic engineering?," *IEEE/ACM Transactions on Networking* **4**, pp. 301–317, June 1996.
7. A. Neidhardt and J. Wang, "The concept of relevant time scales and its application to queuing analysis of self-similar traffic," in *Proceedings of SIGMETRICS'1998/PERFORMANCE'1998*, pp. 222–232, 1998.
8. M. Grossglauser and J. C. Bolot, "On the relevance of long-range dependence in network traffic," *IEEE/ACM Transactions on Networking* **7**, pp. 629–640, Oct. 1999.
9. A. Nogueira and R. Valadas, "Analyzing the relevant time scales in a network of queues," in *SPIE's International Symposium ITCOM 2001*, Aug. 2001.
10. A. Nogueira, P. Salvador, R. Valadas, and A. Pacheco, "Modeling self-similar traffic through markov modulated poisson processes over multiple time scales," in *Proceedings of the 6th IEEE International Conference on High Speed Networks and Multimedia Communications (HSNMC'03)*, July 2003.
11. A. Nogueira, P. Salvador, R. Valadas, and A. Pacheco, "A hierarchical approach based on MMPPs for modeling self-similar traffic over multiple time scales," in *Proceedings of the First International Working Conference on Performance Modeling and Evaluation of Heterogeneous Networks (HET-NETs'03)*, July 2003.
12. K. Meier-Hellstern, "A fitting algorithm for Markov-modulated Poisson process having two arrival rates," *European Journal of Operational Research* **29**, 1987.
13. P. Skelly, M. Schwartz, and S. Dixit, "A histogram-based model for video traffic behaviour in an ATM multiplexer," *IEEE/ACM Transactions on Networking* , pp. 446–458, Aug. 1993.
14. S. Li and C. Hwang, "On the convergence of traffic measurement and queuing analysis: A statistical-match and queuing (SMAQ) tool," *IEEE/ACM Transactions on Networking* , pp. 95–110, Feb. 1997.
15. A. Andersen and B. Nielsen, "A Markovian approach for modeling packet traffic with long-range dependence," *IEEE Journal on Selected Areas in Communications* **16**, pp. 719–732, June 1998.
16. C. Nunes and A. Pacheco, "Parametric estimation in MMPP(2) using time discretization," *Proceedings of the 2nd International Symposium on Semi-Markov Models: Theory and Applications* , Dec. 1998.
17. T. Yoshihara, S. Kasahara, and Y. Takahashi, "Practical time-scale fitting of self-similar traffic with Markov-modulated Poisson process," *Telecommunication Systems* **17**(1-2), pp. 185–211, 2001.
18. P. Salvador, R. Valadas, and A. Pacheco, "Multiscale fitting procedure using Markov modulated Poisson processes," *Telecommunications Systems* **23**, pp. 123–148, June 2003.
19. K. Park and W. Willinger, "Self-similar network traffic: an overview," in *Self-Similar Network Traffic and Performance Evaluation*, K. Park and W. Willinger, eds., Wiley-Interscience, 2000.
20. D. Veitch and P. Abry, "A wavelet based joint estimator for the parameters of LRD," *IEEE Transactions on Information Theory* **45**, Apr. 1999.