MATRIX-ANALYTIC ANALYSIS OF A MAP/PH/1 QUEUE FITTED TO WEB SERVER DATA

ALMA RISKA
Dept. of Comp. Sci., College of William & Mary, Williamsburg, VA 23187, USA
E-mail: riska@cs.wm.edu

MARK S. SQUILLANTE, SHUN-ZHENG YU, ZHEN LIU, LI ZHANG
IBM Thomas J. Watson Research Center, Yorktown Heights, NY 10598, USA
E-mail: {mss,shuzheng,zhen,zhangl}@watson.ibm.com

We consider a MAP/PH/1 queue whose interarrival time and service time processes are fitted to measurement data from commercial Web sites. A methodology is developed for this fitting of Web data to various instances of Markovian Arrival Processes (MAP). Numerical experiments are used to evaluate our approach and to analyze several performance measures of the MAP/PH/1 queue.

1 Introduction

As the Internet continues to grow at an extremely rapid pace, Web servers are playing an ever increasing and important role in our daily life by providing access to a wide variety of commercial services. Critical issues for continued and successful growth concern the performance of such Web servers, which must provide reliable, scalable and efficient access to Internet applications and services.

A significant body of research has considered the network-packet or client-request patterns for different Web server environments (with greater attention given to the former), and has developed mathematical models to characterize these (packet or request) traffic patterns; e.g., see [1,2,3,4] and the references therein. Conversely, far less research has focused on analytic queueing-theoretic models of Web server performance based on such mathematical traffic models, and even less research has evaluated the quality of these traffic models with respect to various measures of Web server performance (as opposed to statistical tests of fit between the traffic data and the models). In fact, given the complexity of Web traffic patterns, it is commonly believed that Markovian methods cannot be used to model Web server traffic patterns or Web server performance with sufficient accuracy.

In this paper we present an initial study that develops a methodology for fitting measurement data from Web sites to the interarrival time and service time processes of a MAP/PH/1 queue used to model Web server performance.

Our study is based on an analysis of the client-request patterns found at real commercial Web sites from the retail industry, which demonstrates complex traffic patterns. We also observe from the measurement data that there are (peak and off-peak) traffic periods which appear to be stationary for relatively long intervals of time (on the order of five hours), and thus it is reasonable for us to focus on the steady-state performance measures that are of most interest in current Web server performance analysis and capacity planning studies. By modeling Web server performance as a MAP/PH/1 queue, we can further exploit matrix-analytic methods to obtain such stationary performance measures in a computationally efficient and numerically stable manner.

A considerable body of research has focused on developing strategies for fitting data sets to various stochastic models. Much of these efforts have examined the fitting of independent data sets to phase-type distributions (e.g., refer to [5,6,7,8] and the references cited therein) and the fitting of correlated data sets to MMPPs and BMAPs. In each case, the basic approach is based either on matching the moments of the data set with the moments of the stochastic model, or on using maximum likelihood estimators (MLEs). The moment matching approaches tend to be computationally more efficient, but the models tend to be somewhat more restrictive with respect to the number of states and the interactions among states that comprise the underlying Markov chain. Meier-Hellstern[9] uses the approach of MLEs for fitting data sets to 2-state MMPPs. Optimization methods have been used by Ryden[10] as part of an MLE-based approach for MMPP parameter estimation. Horvath et al.[11] develop a heuristic fitting method for MAPs based on the superposition of phase-type and interrupted Poisson processes. An estimation procedure for BMAPs based on the EM algorithm has been proposed by Breuer.[12,13] We refer the interested reader to [14], the above references, and the references therein for additional technical details.

An important distinction between the present study and previous related work is our focus here on client-request patterns (as opposed to network-packet patterns) found at real commercial Web sites. Our methodology must be sufficiently scalable (in time and space) to be able to handle the very large data sets of such environments, whose sizes exceed those of many previous studies by several orders of magnitude. Moreover, as demonstrated in a recent study,[15] an analysis of Web server performance based on the batch arrival process obtained from Web site data can significantly overestimate the response time and queue length performance measures of the corresponding Web server system. We therefore develop a hierarchical approach that first considers the batch arrival process at a relatively coarse time scale and then considers the underlying interarrival process at a finer time scale. In partic-

ular, we exploit standard Hidden Markov Model (HMM) methods to identify and characterize the dependence structure of, and some of the variability in, the batch process data set. This includes identifying the set of control states for the MAP and defining the interactions among these control states, at a coarser time scale than the batch process data set. Then we use an additional statistical analysis of the data set at a finer time scale to characterize the sojourn time and interarrival process for each control state. This includes exploiting the EM algorithm for fitting phase-type distributions to subsets of the data set. The result is a MAP that captures the complexities of client-request patterns found at the Web server environments of interest.

The remainder of the paper is organized as follows. After covering some technical preliminaries in Section 2, we then present our methodology for fitting Web server measurement data to instances of MAPs. Section 4 provides a representative set of results from a large number of numerical experiments. Our concluding remarks are given in Section 5.

2 Technical Preliminaries

2.1 Web Server Environments

We consider commercial Web sites from the retail industry whose identities will remain anonymous for obvious confidentiality restrictions. The Web sites generally consist of multiple single-server computing nodes to which incoming requests are routed by a set of front-end routers. Each router attempts to balance the Web site load across the set of single-server computing nodes by sending more traffic to less loaded nodes.

The access logs generated at one of the server nodes from each of these commercial Web sites are used as the basis for our study. Previous research has shown that the set of routers has the effect of smoothing out and equalizing (in a statistical sense) the arrival process observed at each of the server nodes when the client requests are relative small and have relatively low variability.[16,2] We have verified that the content served at the commercial Web sites of interest in our present study are consistent with the content served at the Web site considered in [16,2], at least in this respect, and thus our analysis of the access logs from one of the server nodes is assumed to be representative of the arrival process found at the other server nodes comprising the commercial Web sites of interest.

A large set of detailed measurements were performed on an isolated IBM SP2 uniprocessor node to obtain the resource requirements of each page. These measurement-based experiments reveal that the time to serve static

Web pages fits very well to a linear function of the page size, at least for the Web environments under consideration. We therefore use in our performance model the corresponding linear function fitted against the measurement data, $f_s(\cdot)$, in order to accurately estimate the resource requirements of each static page request based on the size of the request provided in the access log.

More information on the Web server environments considered in our study can be found in [16,2] and the references cited therein.

2.2 Web Server Data

Each access log contains several pieces of useful information about every client request served by the corresponding Web server node. This includes the time epoch of the n^{th} request, which we denote by A_n, and the number of bytes comprising the n^{th} request, which we denote by B_n, $n \in \mathbb{Z}^+$. The unit of time in the access logs available to us is one second, which is quite standard. There are typically tens or even hundreds of requests within a second for the commercial Web sites of interest. Thus, the access log data set directly provides us with the discrete-time batch process for the number of client requests per second, which we denote by $\mathcal{B}(t)$, $t \in \mathbb{Z}^+$.

There is an important problem with the coarse time granularity of this batch arrival process for our purposes. As demonstrated by a recent study,[15] the client-request response time and queue length measures obtained under the batch process $\mathcal{B}(\cdot)$ can significantly overestimate the performance measures in the real system (by more than an order of magnitude). To address this problem, we apply the methodology developed in [15] to construct the corresponding interarrival process at finer time scales from the batch process at the coarser time scale of 1 second. A discussion of this methodology, which basically exploits the statistical properties of the batch arrival process $\mathcal{B}(\cdot)$ to construct the interarrival process, is beyond the scope of the present paper and we refer the interested reader to [15] for the technical details.

The above procedure provides an accurate transformation from the coarse time-scale batch process data set to the interarrival process data set at a finer time scale (which is a millisecond for the data sets used in our study). We shall henceforth focus on the latter (finer time-scale) versions of the Web server data sets. Let $\mathcal{A}_n$ denote the time epoch of the n^{th} request in each of these data sets, $n \in \mathbb{Z}^+$. Let T_n and S_n respectively denote the interarrival time and the service time of the n^{th} request. We then have $T_n = \mathcal{A}_n - \mathcal{A}_{n-1}$ and $S_n = f_s(B_n)$ for $n \in \mathbb{Z}^+$, where $T_1 \equiv 0$.

Given the main objectives of our study, we identify and focus on sufficiently long stationary intervals of (peak and off-peak) traffic periods found in

each of the commercial Web site data sets of interest. Thus, the corresponding processes $\{T_n\}$ and $\{S_n\}$ derived from these data sets are stationary sequences in which each of the generic random variables $T \stackrel{\mathrm{d}}{=} T_n$ and $S \stackrel{\mathrm{d}}{=} S_n$ follow common marginal distributions $F_T(\cdot)$ and $F_S(\cdot)$, respectively. We have selected three representative data sets to be used in our study. The first, which we call *Trace A*, represents the peak traffic period for one of the Web sites of interest. This data set is long-range dependent with a Hurst parameter H_A of approximately 0.78. Another data set, called *Trace B*, represents the off-peak traffic period for one of the Web sites, which is long-range dependent with a Hurst parameter $H_B \approx 0.64$. The third data set, *Trace C*, is somewhat artificial but included here to represent a more extreme case where the Hurst parameter H_C is around 0.9. Each of these data sets is comprised of traffic periods whose lengths are on the order of five hours and consist of more than 500,000 data points.

Our analysis of the measurement data further suggests that the service time process has a negligible dependence structure. We shall therefore assume herein that the client-request service times are i.i.d. according to a phase-type distribution. This further explains our use of a MAP/PH/1 queue to model Web server performance.

2.3 MAP/PH/1 Queue

Following standard notation, let $(\mathbf{D}_0, \mathbf{D}_1)$ be the MAP descriptors of order m_A with mean $\lambda = (\mathbf{x}\mathbf{D}_1\mathbf{e})^{-1}$, where $\mathbf{e}$ is the column vector of appropriate dimension containing all ones and $\mathbf{x}$ is the invariant probability vector of the generator $\mathbf{D} = \mathbf{D}_0 + \mathbf{D}_1$, i.e., the (unique) solution of $\mathbf{x}\mathbf{D} = \mathbf{0}$ and $\mathbf{x}\mathbf{e} = 1$. Let $(\boldsymbol{\beta}, \mathbf{S})$ be the phase-type service time distribution parameters of order m_B with mean $\mu^{-1} = -\boldsymbol{\beta}\mathbf{S}^{-1}\mathbf{e}$. The infinitesimal generator matrix $\mathbf{Q}$ for the corresponding MAP/PH/1 queue has a structure given by

$$\mathbf{Q} = \begin{bmatrix} \mathbf{B}_1 & \mathbf{B}_0 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots \\ \mathbf{B}_2 & \mathbf{A}_1 & \mathbf{A}_0 & \mathbf{0} & \mathbf{0} & \cdots \\ \mathbf{0} & \mathbf{A}_2 & \mathbf{A}_1 & \mathbf{A}_0 & \mathbf{0} & \cdots \\ \mathbf{0} & \mathbf{0} & \mathbf{A}_2 & \mathbf{A}_1 & \mathbf{A}_0 & \cdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{A}_2 & \mathbf{A}_1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}, \tag{1}$$

where

$$\mathbf{B}_0 = \mathbf{D}_1 \otimes \boldsymbol{\beta}, \qquad \mathbf{B}_1 = \mathbf{D}_0, \qquad \mathbf{B}_2 = \mathbf{I}_{m_A} \otimes \mathbf{S}^0, \tag{2}$$

$$\mathbf{A}_0 = \mathbf{D}_1 \otimes \mathbf{I}_{m_B}, \qquad \mathbf{A}_1 = \mathbf{I}_{m_A} \otimes \mathbf{S} + \mathbf{D}_0 \otimes \mathbf{I}_{m_B}, \qquad \mathbf{A}_2 = \mathbf{I}_{m_A} \otimes \mathbf{S}^0\boldsymbol{\beta}, \tag{3}$$

$\mathbf{I}_k$ is the order k identity matrix, $\mathbf{S}^0 = -\mathbf{S}\mathbf{e}$, and $\otimes$ denotes the Kroneker product.

Assuming this QBD process to be irreducible and positive recurrent, then the components of its stationary probability vector $\boldsymbol{\pi}$ are given by

$$(\boldsymbol{\pi}_0, \boldsymbol{\pi}_1) \begin{bmatrix} \mathbf{B}_{00} & \mathbf{B}_{01} \\ \mathbf{B}_{10} & \mathbf{B}_{11} + \mathbf{R}\mathbf{A}_2 \end{bmatrix} = \mathbf{0}, \tag{4}$$

$$\boldsymbol{\pi}_{k+1} = \boldsymbol{\pi}_1 \mathbf{R}^k, \quad k \in \mathbb{Z}_+, \tag{5}$$

$$\boldsymbol{\pi}_0 \, \mathbf{e} + \boldsymbol{\pi}_1 (\mathbf{I} - \mathbf{R})^{-1} \mathbf{e} = 1, \tag{6}$$

where $\mathbf{R}$ is the minimal non-negative solution of the equation

$$\mathbf{R}^2 \mathbf{A}_2 + \mathbf{R}\mathbf{A}_1 + \mathbf{A}_0 = \mathbf{0}. \tag{7}$$

Define $\mathbf{A} \equiv \mathbf{A}_0 + \mathbf{A}_1 + \mathbf{A}_2$. We shall henceforth assume that the matrix $\mathbf{A}$ is irreducible, since this is indeed the case for all instances of the MAP/PH/1 queue considered in our study.

Given the components of the invariant vector $\boldsymbol{\pi}$, we can obtain various performance measures of interest. In particular, the tail distribution of the number of Web server requests in the system can be expressed as

$$\mathsf{P}[Q > x] = \sum_{k=x+1}^{\infty} \boldsymbol{\pi}_k \mathbf{e} = \boldsymbol{\pi}_1 \mathbf{R}^x (\mathbf{I} - \mathbf{R})^{-1} \mathbf{e}, \quad x \geq 0, \tag{8}$$

with the corresponding expectation given by

$$\mathsf{E}[Q] = \boldsymbol{\pi}_1 \sum_{k=0}^{\infty} \mathbf{R}^k \mathbf{e} + \boldsymbol{\pi}_1 \sum_{k=0}^{\infty} k \mathbf{R}^k \mathbf{e} = \boldsymbol{\pi}_1 (\mathbf{I} - \mathbf{R})^{-1} \mathbf{e} + \boldsymbol{\pi}_1 \mathbf{R} (\mathbf{I} - \mathbf{R})^{-2} \mathbf{e}. \tag{9}$$

The expected response time of Web server requests in the system can then be calculated using Little's law[17] and (9), which yields

$$\mathsf{E}[R] = \lambda^{-1} \left(\boldsymbol{\pi}_1 (\mathbf{I} - \mathbf{R})^{-1} \mathbf{e} + \boldsymbol{\pi}_1 \mathbf{R} (\mathbf{I} - \mathbf{R})^{-2} \mathbf{e} \right). \tag{10}$$

Let η denote the spectral radius of the matrix $\mathbf{R}$, which is often called the *caudal characteristic*.[18] In addition to providing the stability condition for the MAP/PH/1 queue, η is indicative of the tail behavior of the stationary queue length distribution. Let $\mathbf{u}$ and $\mathbf{v}$ be the left and right eigenvectors corresponding to η normalized by $\mathbf{u}\mathbf{e} = 1$ and $\mathbf{u}\mathbf{v} = 1$. Under the above assumptions, it is known that[19]

$$\mathbf{R}^x = \eta^x \, \mathbf{v} \cdot \mathbf{u} + \mathrm{o}(\eta^x), \quad \text{as} \quad x \to \infty,$$

which together with equation (5) yields

$$\boldsymbol{\pi}_x \mathbf{e} \;=\; \boldsymbol{\pi}_1 \mathbf{v}\, \eta^{x-1} \;+\; \mathrm{o}(\eta^{x-1}), \quad \text{as } x \to \infty. \tag{11}$$

It then follows that

$$\mathsf{P}[Q > x] \;=\; \frac{\boldsymbol{\pi}_1 \mathbf{v}}{1-\eta}\, \eta^x \;+\; \mathrm{o}(\eta^x), \quad \text{as } x \to \infty, \tag{12}$$

and thus

$$\lim_{x \to \infty} \frac{\mathsf{P}[Q > x]}{\eta^x} \;=\; \frac{\boldsymbol{\pi}_1 \mathbf{v}}{1-\eta}, \tag{13}$$

or equivalently

$$\mathsf{P}[Q > x] \;\sim\; \frac{\boldsymbol{\pi}_1 \mathbf{v}}{1-\eta}\, \eta^x, \quad \text{as } x \to \infty. \tag{14}$$

The caudal characteristic can be obtained without having to first solve for the matrix $\mathbf{R}$. We define the matrix $\mathbf{A}^*(s) = \mathbf{A}_0 + s\mathbf{A}_1 + s^2\mathbf{A}_2$, for $0 < s \leq 1$. Since the generator matrix $\mathbf{A}$ is irreducible, this matrix $\mathbf{A}^*(s)$ is irreducible with nonnegative off-diagonal elements. Let $\chi(s)$ denote the spectral radius of the matrix $\mathbf{A}^*(s)$. Then, under the above assumptions, η is the unique solution in (0,1) of the equation $\chi(s) = 0$. This solution can be directly computed, which is the approach taken for all of the experiments in Section 4. A more efficient method is developed in [20].

The maximum throughput of the Web server system is given by the maximum value of λ that yields a stable MAP/PH/1 queue. Since the matrix $\mathbf{A}$ is irreducible, the stability condition is given by[21]

$$\mathbf{x}\mathbf{A}_0\mathbf{e} \;<\; \mathbf{x}\mathbf{A}_2\mathbf{e}, \tag{15}$$

where $\mathbf{x}$ is the invariant probability vector of $\mathbf{A}$. This equation can be used to solve for the maximum throughput without having to first solve for the matrix $\mathbf{R}$. In this case we define $\widehat{\rho} = (\mathbf{x}\mathbf{A}_0\mathbf{e})/(\mathbf{x}\mathbf{A}_2\mathbf{e})$, where from equation (15) $\widehat{\rho} < 1$. Alternatively, the caudal characteristic η can be used to determine an effective measure of maximum throughput that is useful in practice.

3 Methodology

A primary objective of this paper is to develop a general parameter estimation methodology for fitting Web server data to MAPs as part of deriving a matrix-analytic analysis of Web server performance models. Our methodology can be viewed to be a hierarchical approach, based on the observation that a MAP essentially consists of a set of *control states* with arbitrary interactions among

them and a set of independent interarrival time processes, one for each of the control states. We first employ standard HMM methods to identify the set of control states and define the interactions among these states, at a coarser time scale than the data set, in an attempt to capture the correlations among the data set points as well as some of the variability. Then we either use these results directly to construct an MMPP, or we exploit these results together with an additional statistical analysis of the data set and the EM algorithm for fitting phase-type distributions in order to construct a more general MAP. The interarrival process for each control state is an i.i.d. stochastic sequence, which introduces additional states in the underlying Markov chain of the MAP process. We refer to these additional states as *phase-type states.*

An overview of the basic steps of our hierarchical fitting methodology is provided in Figure 1. Note that the algorithm input consists of the data set and a few assumptions and initial values (discussed in Section 3.1), but that our methodology does not place any restrictions on the structure of the underlying Markov chain of the MAP. The first step produces the set of outputs described in **1b** under the assumption of either exponential or hyperexponential sojourn time distributions for each control state. In the latter case, we have $\boldsymbol{\mu}_i = [\boldsymbol{\mu}_{i,1}, \ldots, \boldsymbol{\mu}_{i,m_H}]$ for each control state i, $1 \leq i \leq m_C$, where m_C denotes the number of control states and m_H denotes the number of phases in the hyperexponential distributions; otherwise, $\boldsymbol{\mu}_i$ is a scalar. The corresponding MMPP is constructed directly from the output of step **1**, whereas the remaining steps are used together with the output of step **1** to construct the corresponding MAP. In step **3**, the fitting of the data set sequence $\{\mathcal{S}_i\}$ for control state i to a Coxian distribution uses a slightly modified version of an implementation[22] of the EM algorithm developed in [5]. The remaining details of the basic steps in Figure 1 are explained in Sections 3.1 and 3.2.

3.1 Hidden Markov Model for Parameter Estimation

An HMM with explicit state duration is a doubly (embedded) stochastic process, whose intensity is controlled by a finite-state discrete-time Markov chain $\{J_n : n \in Z^+\}$ on the state space $\{i : 1 \leq i \leq m_C\}$ representing the set of control states. The amount of time that the process has been in the current state J_n at time n is denoted by τ_n, and the number of arrivals per unit time associated with state J_n is denoted by r_n, which is the observable output associated with state J_n. It is usually assumed that the control states $\{J_n\}$ and the observations $\{r_n\}$ are conditionally independent with the conditional distribution of r_n dependent on J_n only.

Since this semi-Markov process is not directly observable, the state se-

1. Use HMM methods to construct the control states and their interactions
  a. Input
    - data set with N data entries
    - desired number of control states
    - assumption for the arrival process per control state (coarse time scale)
      - Poisson
    - assumption for the sojourn times per control state
      - Exponential
      - Hyperexponential : desired number of phases
  b. Output
    - number of control states m_C
    - transition probability matrix $\mathbf{P} = [\mathbf{P}_{i,j}]_{1 \le i,j \le m_C}$ for the control states
    - mean interarrival times $\boldsymbol{\lambda}_i^{-1}$ for $1 \le i \le m_C$; $\boldsymbol{\lambda} = [\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_{m_C}]$
    - (vector of) mean sojourn times $\boldsymbol{\mu}_i^{-1}$ for $1 \le i \le m_C$; $\boldsymbol{\mu} = [\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_{m_C}]$
    - sojourn time probability vectors $\mathbf{p}_i^{\boldsymbol{\mu}}$ for $1 \le i \le m_C$, if using hyperexponential distribution; $\mathbf{p}^{\boldsymbol{\mu}} = [\mathbf{p}_1^{\boldsymbol{\mu}}, \ldots, \mathbf{p}_{m_C}^{\boldsymbol{\mu}}]$; $\mathbf{p}_i^{\boldsymbol{\mu}} = [\mathbf{p}_{i,1}^{\boldsymbol{\mu}}, \ldots, \mathbf{p}_{i,m_H}^{\boldsymbol{\mu}}]$
    - mapping $\{J_n\}$ of each data entry to its corresponding control state
2. Construct a data set sequence $\{\mathcal{S}_i\}$ per control state i using mapping $\{J_n\}$ from step **1** and the original data set
3. Feed each sequence $\{\mathcal{S}_i\}$ to EM algorithm to generate a Coxian distribution for the interarrival process of each control state
4. Compute the probability of state change upon arrival using mapping $\{J_n\}$
5. Construct $\mathbf{D}_0$ using:
  - transition probability matrix from step **1**
  - model for sojourn times from step **1**
  - Coxian model for interarrivals from step **3**
6. Construct $\mathbf{D}_1$ using:
  - Coxian model for interarrivals from step **3**
  - transition probability matrix from step **1**
  - model for sojourn times from step **1**
  - probability vector computed in step **4**

Figure 1. Overview of our MAP parameter estimation methodology.

quence $\{J_n, \tau_n\}$ and the model parameters (i.e., the transition probability matrix $\mathbf{P}$ for the control states, the mean interarrival times $\boldsymbol{\lambda}_i^{-1}$, the vector of mean sojourn times $\boldsymbol{\mu}_i^{-1}$, the sojourn time probability vector $\mathbf{p}_i^{\boldsymbol{\mu}}$ for each control state i, and the control state sequence $\{J_n\}$ of the data set) are estimated from the observed sequence $\{r_n\}$. The main steps of the standard recursive procedure for HMM with explicit state duration are summarized as

follows:

- Given an initial set of assumptions for the HMM model parameters, obtain refined maximum likelihood estimators for the model parameters by applying the *HMM re-estimation algorithms* with explicit state duration to the given observation sequence $\{r_n\}$.

- Apply one of the many *HMM forward-backward algorithms* with explicit state duration to find the maximum a posteriori state estimate, $\{J_n\}$, for the given observation sequences $\{r_n\}$.

We refer the interested reader to [23] for an overview of the details on these standard HMM algorithms. Additional technical details can be found in [24,25,26] and the references cited therein.

Following the above procedure, we can obtain the maximum likelihood model parameters for the given observation sequence $\{r_n\}$ and the state space $\{1, \ldots, m_C\}$. Let $\widehat{H}_i(\tau)$ denote the estimated non-parametric probability mass function for the sojourn time τ of state i, and let $\widehat{O}_i(r)$ denote the estimated non-parametric probability mass function for the observation r of state i.

The total number of model parameters can be reduced if the observation distribution or the state sojourn time distribution is approximated by some parametric distributions such as Gaussian, Poisson or gamma distributions.[27,28,29] In this case, one only needs to estimate a few parameters that specify the selected distribution functions. Ferguson[30] has shown that the parameters for the parametric sojourn time distribution $H_i(\tau)$ and the parametric observation distribution $O_i(r)$ for state i can be found by maximizing $\sum_\tau \widehat{H}_i(\tau) \ln(H_i(\tau))$ and $\sum_r \widehat{O}_i(r) \ln(O_i(r))$ subject to the stochastic constraints $\sum_\tau H_i(\tau) = 1$ and $\sum_r O_i(r) = 1$.

Under the assumptions that the arrival process for each control state (at a coarse time scale) is Poisson and that the per-control-state sojourn times follow a hyperexponential distribution, i.e.,

$$O_i(r) = \frac{(\boldsymbol{\lambda}_i t)^r}{r!} e^{-\boldsymbol{\lambda}_i t}, \tag{16}$$

$$H_i(\tau) = \sum_{j=1}^{m_H} \mathbf{p}_{ij}^{\boldsymbol{\mu}} \, \boldsymbol{\mu}_{ij} \, e^{-\boldsymbol{\mu}_{ij}(\tau-1)}, \tag{17}$$

where $\boldsymbol{\lambda}_i \geq 0$, $\boldsymbol{\mu}_{ij} \geq 0$ and $\sum_j \mathbf{p}_{ij}^{\boldsymbol{\mu}} = 1$, then the arrival rate $\boldsymbol{\lambda}_i$ of the Poisson process for state i can be estimated by $\widehat{\boldsymbol{\lambda}}_i = \sum_r \widehat{O}_i(r) r$.[30,27] The parameters

$\mathbf{p}_{ij}^{\boldsymbol{\mu}}$ and $\boldsymbol{\mu}_{ij}$ of the hyperexponential distribution can be determined numerically via equation (17).

Finally, as part of the initialization step, we set the total number of control states to a prespecified input parameter (which was analyzed and tuned in our experiments). When this parameter is a sufficiently large integer, then in the re-estimation procedure, the states that are never visited will be deleted from the state space, so that the value of m_C is reduced to a number of control states that match the data set. This led to a maximum of 20 control states for the data sets used in our study. We also need to initialize the elements of the transition probability matrix, the control state sojourn time distributions, and the initial control state probability vector. An often used choice is to assume that these initial values for the model parameters are uniformly distributed, which was the choice made in our study. In addition, we assume the initial values of the control-state arrival rates to be proportional to the state index, i.e., $\boldsymbol{\lambda}_i = r_{max}\frac{i}{m_C}$ where i is the index of the control state, r_{max} is the maximum value of r, and m_C is the total number of control states.

3.2 Generation of MAP from HMM output

The above HMM methods produce the output described in Figure 1, which includes the sequence $\{J_n\}$, $1 \leq n \leq N$, representing the mapping of the entries in the data set to the set of control states (at a coarse time scale). That is, as part of the HMM analysis, each interarrival time in the data set is assigned to one of the m_C control states. We first construct a new data set sequence $\{\mathcal{S}_i\}$, $1 \leq i \leq m_C$, that consists of all of the interarrival times from the data set where $J_n = i$, which are then combined in the same relative order as in the original data set. We also compute from the sequence $\{J_n\}$, $1 \leq n \leq N$, the probability vector $\widehat{\mathbf{p}}$ of dimension m_C where the element $\widehat{\mathbf{p}}_i$ denotes the probability that upon an arrival from control state i the process switches to another control state. Specifically, we have

$$\widehat{\mathbf{p}}_i \;=\; \frac{\sum_{j=2}^{N} I_{J_j \neq i,\, J_{j-1}=i}}{\sum_{j=2}^{N} I_{J_j=i,\, J_{j-1}=i} \;+\; \sum_{j=2}^{N} I_{J_j \neq i,\, J_{j-1}=i}}, \tag{18}$$

where I_A denotes the indicator function for event A having the value 1 if A occurs and the value 0 otherwise. Thus, with probability $1 - \widehat{\mathbf{p}}_i$ the process immediately returns to the same control state i.

To help facilitate the description of the procedure for generating MAPs from the above set of variables, we need the following definitions:

- $\boldsymbol{\Lambda} = \mathbf{diag}(\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_{m_C})$ is a diagonal matrix of order m_C whose diagonal elements are the elements of vector $\boldsymbol{\lambda}$;

- $\mathbf{P}_i^{\boldsymbol{\mu}}$ is the $m_H \times m_H$ matrix whose rows are all equal to $\mathbf{p}_i^{\mu}$, $1 \leq i \leq m_C$;
- $\boldsymbol{\Phi} = \mathbf{diag}(\boldsymbol{\Phi}_1, \ldots, \boldsymbol{\Phi}_{m_C})$ is a (block) diagonal matrix of order $m_C \cdot m_H$, where $\boldsymbol{\Phi}_i = \mathbf{diag}(\boldsymbol{\mu}_{i,1}, \ldots, \boldsymbol{\mu}_{i,m_H})$ is a diagonal matrix of order m_H;
- $\mathbf{col}(M, k, j)$ is a matrix function that partitions the columns of the matrix M into blocks of size k and then extracts the j^{th} such block of columns of size k from matrix M (the resulting matrix has the same number of rows as M and k columns);
- $\mathbf{row}(M, k, j)$ is a matrix function that partitions the rows of the matrix M into blocks of size k and then extracts the j^{th} such block of rows of size k from matrix M (the resulting matrix has k rows and the same number of columns as M);
- $\mathbf{V} = [\mathbf{V}_1 \, \mathbf{V}_2 \, \ldots \, \mathbf{V}_{m_C}]$, where $\mathbf{V}_i = \mathbf{col}(\mathbf{P}, 1, i) \otimes \mathbf{P}_i^{\boldsymbol{\mu}}$, $1 \leq i \leq m_C$.

The off-diagonal elements of the matrix $\mathbf{D}_0^{\text{MMPP}}$ and the matrix $\mathbf{D}_1^{\text{MMPP}}$ for an MMPP with exponential interarrival times and hyperexponential sojourn times per control state can be expressed as

$$\mathbf{D}_0^{\text{MMPP}} = \boldsymbol{\Phi}\mathbf{V}, \qquad \mathbf{D}_1^{\text{MMPP}} = \boldsymbol{\Lambda} \otimes \mathbf{I}_{m_H}. \tag{19}$$

Then the diagonal element of each row of $\mathbf{D}_0^{\text{MMPP}}$ is computed as the negative sum of the non-diagonal elements on the same row in the matrix $\mathbf{D}_0^{\text{MMPP}} + \mathbf{D}_1^{\text{MMPP}}$. The number of states in this MMPP is $m_C \cdot m_H$. During the numerical experiments we observed that some of the values of the vectors $\mathbf{p}_i^{\boldsymbol{\mu}}$, $1 \leq i \leq m_C$, are very small, i.e., less than a desired tolerance of accuracy. The states that are represented by these probabilities are removed from the set of states during the generation of $\mathbf{D}_0$ and $\mathbf{D}_1$ to avoid numerical problems in the solution of the MAP/PH/1 queues. We note that the size of the state space is decreased by up to 30% with this simple state reduction technique.

From the HMM output we can also construct a more general MAP by using the same matrix $\mathbf{D}_0$ from the above MMPP and modifying the matrix $\mathbf{D}_1$ by making use of the probability vector $\widehat{\mathbf{p}}$. Define $\widehat{\mathbf{P}} = \mathbf{diag}(\widehat{\mathbf{p}}_1, \ldots, \widehat{\mathbf{p}}_{m_C})$ to be the order m_C diagonal matrix corresponding to the vector $\widehat{\mathbf{p}}$. We then have

$$\mathbf{D}_0^{\text{MAP}} = \mathbf{D}_0^{\text{MMPP}}, \tag{20}$$

$$\mathbf{D}_1^{\text{MAP}} = \left(\mathbf{I}_{m_C \cdot m_H} - \mathbf{diag}(((\widehat{\mathbf{P}} \otimes \mathbf{I}_{m_H})\mathbf{V}\mathbf{e})^T) + (\widehat{\mathbf{P}} \otimes \mathbf{I}_{m_H})\mathbf{V}\right) \mathbf{D}_1^{\text{MMPP}}. \tag{21}$$

We compute from the sequence $\{J_n\}$, $1 \leq n \leq N$, only the probability of leaving a control state upon arrival. The probability of reaching any other

control state is not computed from this sequence, but rather from the probability transition matrix $\mathbf{V}$ since it is estimated using a similar analysis (which is reflected in the definition of matrix $\mathbf{D}_1^{\text{MAP}}$ in equation (21)).

Another set of MAP processes is obtained by incorporating the results of fitting the interarrival times in each of the data set sequences $\{\mathcal{S}_i\}$, $1 \leq i \leq m_C$, to a Coxian distribution using the EM algorithm.[5] This computes for each control state i the corresponding vector $\widehat{\boldsymbol{\alpha}}_i$ and matrix $\widehat{\mathbf{T}}_i$, $1 \leq i \leq m_C$, both of order m_X. We define

$$\mathbf{U}_i = \mathbf{row}(\mathbf{D}_0^{\text{MAP}}, m_H, i) \otimes \hat{\mathbf{T}}^i,$$
$$\mathbf{X}_i = \mathbf{row}(\mathbf{I}_{m_C m_H} - \mathbf{diag}(((\hat{\mathbf{P}} \otimes \mathbf{I}_{m_H})\mathbf{V}\mathbf{e})^T) + (\hat{\mathbf{P}} \otimes \mathbf{I}_{m_H})\mathbf{V}, m_H, i) \otimes \hat{\mathbf{T}}_i^o \hat{\boldsymbol{\alpha}}_i,$$
$$\mathbf{U} = \left[\mathbf{U}_1^T, \cdots, \mathbf{U}_{m_C}^T\right]^T, \qquad \mathbf{X} = \left[\mathbf{X}_1^T, \cdots, \mathbf{X}_{m_C}^T\right]^T.$$

Then the matrices $\mathbf{D}_0^{\text{MAP-C}}$ and $\mathbf{D}_1^{\text{MAP-C}}$, where MAP-C stands for the MAP with Coxian interarrival processes for each control state, can be expressed as

$$\mathbf{D}_0^{\text{MAP-C}} = \mathbf{U}, \qquad \mathbf{D}_1^{\text{MAP-C}} = \mathbf{X}. \tag{22}$$

The total number of states for this MAP is $m_C \cdot m_H \cdot m_X$. In the same manner as described above, the states with probabilities that are very close to 0 are removed from the final version of the MAP.

4 Numerical Results

Our objectives in this paper have been to develop a general methodology for fitting measurement data from Web sites to the interarrival time and service time processes of a MAP/PH/1 queue used to model Web server performance. To this end, a large number of numerical experiments have been performed where we apply the methodology of Section 3 to the commercial Web site data sets of Section 2, and then we solve the resulting MAP/PH/1 queue. These experiments are primarily used to explore two key issues: the accuracy of our methodology for fitting Web data to MAPs; and the performance characteristics of the corresponding MAP/PH/1 queues.

We consider fitting the interarrival times of each data set with a wide variety of models that can be obtained from our methodology as follows:

- an exponential (Exp) or Y-phase Coxian (Cox$Y$) distribution for the interarrival times associated with each control state;
- an exponential or Y-phase hyperexponential (HrY) distribution for the sojourn times of each control state.

To facilitate the presentation of results, we shall use the notation MMPP(s,d) and MAP(s,a,d) where s denotes the number of control states, d denotes the distribution of sojourn times for each control state, and a denotes the distribution that models the interarrival times associated with each control state.

In order to evaluate the accuracy of our approach, we compare various steady-state performance measures of the MAP/PH/1 queue against the corresponding measures obtained by simulating the well-known Lindley equation[31] under the input sequence $\{(T_n, S_n)\}$ from the Web data set. We assume that requests are served in an FCFS manner and that the server depletes work at rate $\mathcal{C}$, where $\mathcal{C}$ is a deterministic constant. By varying the parameter $\mathcal{C}$, different server loads $\rho \equiv \lambda \mathsf{E}[S]/\mathcal{C}$ can be considered, where $\lambda = \mathsf{E}[T]^{-1}$. For stability of the queue, we also assume $\rho < 1$. The value of $\mathcal{C}$ can only be reduced up to points that still maintain a stable regenerative G/G/1 queue (in the sense that the system empties with probability 1 and that there are a sufficiently large number of such regeneration points). Moreover, the spectrum of traffic intensities of interest from a practical perspective is well within the range over which the mean response time remains below $100 \times \mathsf{E}[S]$. We therefore restrict our attention to this spectrum of traffic intensities, which also ensures that the regenerative G/G/1 queue is stable for each of the cases considered.

Before presenting a representative sample of our results, it is important to point out that each Web data set available to us in our study represents a single stationary interval of traffic from a specific Web site, i.e., a single sample path of an underlying stationary stochastic process. We have considerable evidence suggesting that the statistical properties of each of these stationary sequences is representative of the per-weekday (peak and off-peak) traffic intervals found at the corresponding commercial Web site. However, our ongoing research is pursuing the use of multiple independent and statistically identical data sets from each of the specific Web sites as replicas in our simulation of the Lindley equation to compare against our results obtained via matrix-analytic methods, as well as the use of such i.i.d. data sequences in our fitting methodology.

4.1 Peak Traffic Period (Trace A)

We vary the number of control states used by the HMM algorithm as part of our methodology for fitting the interarrival times of Trace A to various MAPs. This makes it possible for us to examine the impact of the size of the underlying Markov chain on the accuracy of the model fitting. We start with 2 control states, and then increase to 5 and 10 states. The mean response

times under a small subset of these MAPs as a function of the traffic intensity ρ are plotted in Figure 2(a), together with the corresponding simulation results for Trace A. Our results clearly demonstrate that the accuracy of the fitting improves significantly with increases in the number of control states, as expected. (Note that the MAP(5,Cox2,Exp) results are somewhat more accurate than the MAP(2,Cox2,Exp) results, which are not shown.) This is because more control states in the underlying Markov chain provide greater flexibility which makes it possible to better capture not only the dependence structure but also the variability of the arrival process. The output of the HMM algorithm for larger numbers of control states exhibit small probabilities for entering a few of the control states (which represent extremes of the arrival rate values), together with small transition rates for leaving these control states once entered. This suggests that such control states improve the ability of the MAP to capture the tail of the interarrival process, and that the degree to which this is possible improves with increases in the number of control states.

In order to isolate, to some extent, the impact of the dependence structure on mean response time measures, we have ignored such dependencies and fitted the interarrival times of Trace A to a phase-type distribution. The mean response time measures for this PH/PH/1 queue are also provided in Figure 2(a). It can be clearly observed from these results that the PH/PH/1 fitting is very poor and, with the exception of very light traffic intensities, the PH/PH/1 queue is simply not capable of capturing the performance of the queueing system under Trace A. Conversely, the MAP models that capture the dependence structure do a much better job of matching the queueing system performance, particularly at heavier loads, where the accuracy increases with the complexity of the MAP.

In a similar manner, our methodology is used for fitting Trace A to various MMPPs, and the corresponding mean response times under a small subset of these MMPPs are plotted in Figure 2(b). The results from simulation are also included in the figure for comparative purposes. We continue to observe that the larger the number of control states, the more accurate the fitting. Once again, more control states in the underlying Markov chain provide greater flexibility that makes it possible to better capture both the dependence structure and the variability of the arrival process, for the reasons described above.

We observe that one of the MMPP models provides the most accurate results in comparison with simulation for light loads. Under heavier loads, however, one of the MAPs tends to provide the most accurate results. Specifically, MAP(10,Cox2,Exp) provides the best accuracy with a relative error always less than 20%. We further observe that the models using Coxian dis-

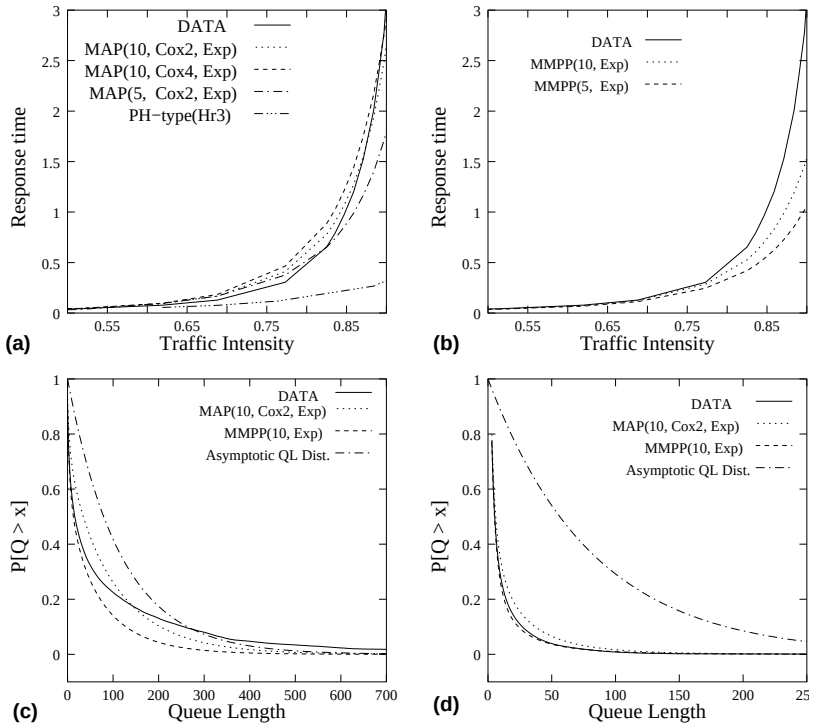

Figure 2. Response time as function of traffic intensity for (a) MAP models, (b) MMPP models; and Queue length tail distribution of fitted models for (c) moderate, (d) high system loads; all for Trace A.

tributions for the interarrival time process of each control state tend to provide better fits. Based on a simple statistical analysis of the sequence $\{J_n\}$ defined in Section 3.1, we further observe that the interarrivals for each control state are not exponential, which is why we use the Coxian distribution to model each of these control-state interarrival processes.

We also study the tail behavior of the queue length distribution, using equation (8), for the best MAP and MMPP models, i.e., MAP(10,Cox2,Exp) and MMPP(10,Exp). For comparison we also consider the asymptotic queue length tail distribution that corresponds to the MAP(10,Cox2,Exp) model, using equation (14) based on the caudal characteristic η, and the tail of the queue length distribution from the simulation of Trace A. Figure 2(c) plots these five queue length tail distributions for the traffic intensity $\rho = 0.77$,

which represents a case where the system is moderately loaded. The corresponding set of results for a traffic intensity of $\rho = 0.90$, which represents a heavily loaded system, are presented in Figure 2(d). Note that in each of these figures we only plot the asymptotic queue length tail distribution for the MAP model because the corresponding curve for the MMPP model is quite close to that of the MAP model.

We observe that the queue length tail distributions obtained under the MAP and MMPP models provide a reasonably close match with the corresponding tail distribution obtained from simulation over a relatively wide range. In the case of moderate load, i.e., $\rho = 0.77$, the tail distribution from the MMPP model closely matches the simulation results for small queue length values, which helps to explain the low relative error values at light to moderate loads. Conversely, the tail distribution from the MAP model overestimates the simulation results for all but very large queue length values. This causes the MAP to yield poorer relative errors at light to moderate loads, although still always less than 20%. In the case of heavier load, i.e., $\rho = 0.90$, the tail distribution from the MMPP model continues to provide a close match with the simulation results but only for very small queue length values. The tail distribution from the MAP model continues to overestimate the simulation results over a relatively large range of queue length values, crossing at around a queue length of 140. In fact, the accuracy of the expected response time under the MAP model is achieved in part by pushing this crossover point relatively far to the right (to larger queue lengths). This further explains why the expected response times under the MAP model underestimate those obtained from simulation at heavier loads. More accurate response times under the MAP model at heavier loads can be obtained by increasing the number of (control and/or phase-type) states in the underlying Markov chain.

We also observe that the asymptotic queue length tail distribution based on the caudal characteristic η dominates all other tail distributions, for both $\rho = 0.77$ and $\rho = 0.90$, across a relatively wide range of queue length values. The tail of the queue length distribution from simulation eventually crosses the asymptotic tail distribution, as expected due to the dependence structure and variability in Trace A, but these crossover points occur at queue length values greater than 250 which can be considered to be a relatively high value of queue length.

We note that the maximum response time value plotted for the MAP(10,Cox2,Exp) model represents $\rho = 0.9$, $\widehat{\rho} = 0.94$ and $\eta = 0.99$. This illustrates and quantifies the degree to which the latter two variables provide a better measure of effective system load than the standard traffic intensity ρ for Trace A, at least from a practical perspective.

4.2 Off-Peak Traffic Period (Trace B)

The same set of experiments and analysis as those described in Section 4.1 are performed on Trace B. Since the results in Section 4.1 demonstrate that a model with 10 control states fits the interarrival process much more accurately than the corresponding models with fewer control states, here we focus solely on models with 10 control states in the analysis of Trace B.

Based on equations (19), (20), (21), and (22), several different MAP and MMPP models are fitted to the interarrival process of Trace B. The mean response times under a small subset of these MAPs and MMPPs as a function of the traffic intensity ρ are respectively plotted in Figures 3(a) and 3(b). In Figure 3(a) we observe that the gap between the PH/PH/1 model and some of the MAP and MMPP models is smaller than the corresponding results of the previous section for Trace A. This is directly related to the weaker long-range dependence of the interarrival process of Trace B. From among the set of MAP models, MAP(10,Cox2,Exp) performs the best with a relative error always less than 12%. On the other hand, the MMPP(10,Exp) model performs slightly better with a worst case relative error of 10%. This supports the notion that the MMPP is a good model for data sets which have a relatively weak long-range dependence structure, or a short-range dependence structure.

The queue length tail distributions for the models MMPP(10,Exp) and MAP(10,Cox2,Exp), as well as the asymptotic behavior (as characterized by equation (14) in terms of the caudal characteristic η) for the model MAP(10,Cox2,Exp), are compared with the queue length tail distribution obtained from the simulation of Trace B in Figure 3(c) for a traffic intensity of 0.88. The corresponding curves for a traffic intensity of 0.95 are shown in Figure 3(d). These plots illustrate that for both moderate and relatively heavy traffic intensities MMPP(10,Exp) is a slightly better fit than MAP(10,Cox2,Exp). However, for heavier traffic intensities, the MAP(10,Cox2,Exp) curve follows the simulation curve for larger values of queue length than does the MMPP(10,Exp) curve.

Note that the maximum response time value plotted for the MAP(10,Cox2,Exp) model represents $\rho = 0.97$, $\widehat{\rho} = 0.99$ and $\eta = 0.99$.

4.3 Strong Long-Range Dependence (Trace C)

We perform the same set of experiments and analysis on the Trace C data set as those considered in Sections 4.1 and 4.2. Figure 4(a) presents the mean response time as a function of the traffic intensity for a small subset of the fitted MAP models in comparison with the corresponding simulation curve for Trace C. We note that the MAPs do a very good job of accurately capturing

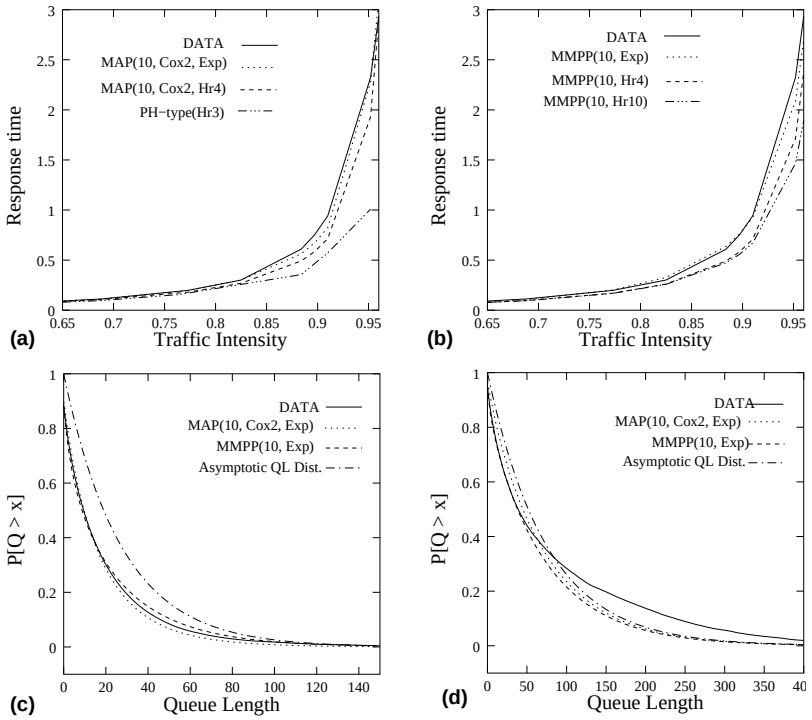

Figure 3. Response time as function of traffic intensity for (a) MAP models, (b) MMPP models; and Queue length tail distribution of fitted models for (c) moderate, (d) high system loads; all for Trace B.

the queueing system performance even though the dependence structure of Trace C is much stronger than that found in Traces A and B. It is this strong long-range dependence in Trace C that causes the MAP models with hyper-exponential sojourn time distributions for the control states to perform much better than the cases where the sojourn times are assumed to be exponentially distributed. Moreover, unlike the 2-phase Coxian distributions that are used to fit the interarrival process for each of the control states for Traces A and B, we choose a 4-phase Coxian distribution for fitting the interarrival process of each control state for Trace C. These 4-phase Coxian distributions are essentially Erlang distributions because we find that the coefficient of variation for each of the constructed control-state trace sequences (see Section 3.2) is less than 0.5.

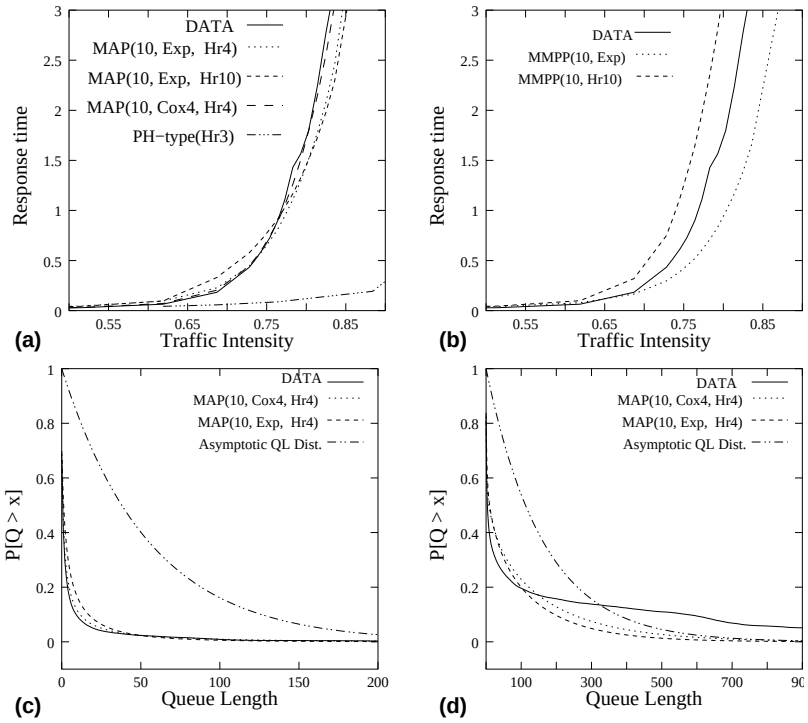

Figure 4. Response time as function of traffic intensity for (a) MAP models, (b) MMPP models; and Queue length tail distribution of fitted models for (c) moderate, (d) high system loads; all for Trace C.

Figure 4(b) is similar to Figure 4(a) but for a small subset of the fitted MMPP models. These results show that the MMPP models only roughly approximate the simulation curve. The best fit for Trace C is provided by the MAP(10,Cox4,Hr4) model with relative error less than 12%.

The queue length tail distributions for the two best fitting models, MAP(10,Cox4,Hr4) and MAP(10,Exp,Hr4), are plotted in Figure 4(c) for the moderate traffic intensity of 0.69, and in Figure 4(d) for the high traffic intensity of 0.83. These plots illustrate that the tail of the queue length distribution for Trace C obtained by simulation is rather heavy and that the MAP(10,Cox4,Hr4) model does a much better job of capturing the characteristics of this heavy tail up to a relatively large queue length value under both moderate and high traffic intensities. However, at high traffic intensi-

ties, the heavier tail of the simulation results eventually crosses from below that of the MAP(10,Cox4,Hr4) model, beyond which it decays much more slowly than all other tail distribution curves. This further explains why the expected response times under the MAP model underestimate those obtained from simulation at heavier loads. On the one hand, as previously noted, the range of traffic intensities that cover mean response time values from $\mathsf{E}[S]$ to $100 \times \mathsf{E}[S]$ represent by far the set of traffic intensities that might be of interest in practice. On the other hand, more accurate response times under the MAP model at heavier loads can be obtained by increasing the number of (control and/or phase-type) states in the underlying Markov chain.

We note that the maximum response time value plotted for the MAP(10,Cox4,Hr4) model represents $\rho = 0.83$, $\widehat{\rho} = 0.84$ and $\eta = 0.99$. This illustrates and quantifies the degree to which the latter two variables provide a better measure of effective system load than the standard traffic intensity ρ for Trace C, at least from a practical perspective.

5 Conclusions

In this paper we presented an initial study that considers a MAP/PH/1 queue whose interarrival time and service time processes are fitted to measurement data from actual commercial Web sites. Our fitting methodology is based on the use of standard HMM methods to identify and characterize the dependence structure of, and some of the variability in, these large data sets, together with additional statistical analysis of the data sets and the EM method for fitting phase-type distributions to construct different instances of MAPs.

The results of our many numerical experiments are quite promising, demonstrating that, contrary to common beliefs, Markovian models and matrix-analytic methods can be used to efficiently and accurately capture the complexities of commercial Web site data sets that exhibit a wide range of dependence structures and variabilities. Moreover, the size of the MAP models required to achieve such results over the spectrum of system loads of interest from a practical perspective is relatively small and manageable, at least for the Web data sets considered in our study.

Finally, the fitting methodology presented in this paper simply leverages standard HMM methods in a direct manner, essentially treating these procedures as a "black box". We are currently exploring the extension and tailoring of these methods from first principles for our specific purposes.

Acknowledgments

This work was performed during a summer internship by Alma Riska at the IBM T.J. Watson Research Center. The software tools used in our study were developed by Alma Riska under grants CCR-0098278 and EIA-9974992. We thank Jim Challenger for performing the detailed measurements described in Section 2.1. We further acknowledge the use of an implementation[22] of the EM algorithm developed in [5], and we thank the authors for making their code publically available via the World Wide Web. Lastly, we thank the referees for their helpful comments on an early draft of this paper.

References

1. Walter Willinger, Murad S. Taqqu, Robert Sherman, and Daniel V. Wilson. Self-similarity through high-variability: Statistical analysis of Ethernet LAN traffic at the source level. *IEEE/ACM Transactions on Networking*, 5(1):71–86, February 1997.
2. Mark S. Squillante, David D. Yao, and Li Zhang. Web traffic modeling and web server performance analysis. In *Proceedings of the IEEE Conference on Decision and Control*, December 1999.
3. Bo Friis Nielsen. Modelling long-range dependent and heavy-tailed phenomena by matrix analytic methods. In *Advances in Algorithmic Methods for Stochastic Models,* G. Latouche and P. Taylor (eds.), pages 265–278. Notable Publications, 2000.
4. Zhen Liu, Nicolas Niclausse, and Cesar Jalpa-Villanueva. Traffic model and performance evaluation of Web servers. *Performance Evaluation*, 46:77–100, October 2001.
5. Soren Asmussen, Olle Nerman, and Marita Olsson. Fitting phase-type distributions via the EM algorithm. *Scandinavian Journal of Statistics*, 23:419–441, 1996.
6. Soren Asmussen. Phase-type distributions and related point processes: Fitting and recent advances. In Srinivas R. Chakravarthy and Attahiru S. Alfa, editors, *Matrix-Analytic Methods in Stochastic Models*, pages 137–149. Marcel Dekker, 1997.
7. Andreas Lang and Jeffrey L. Arthur. Parameter approximation for phase-type distributions. In Srinivas R. Chakravarthy and Attahiru S. Alfa, editors, *Matrix-Analytic Methods in Stochastic Models*, pages 151–206. Marcel Dekker, 1997.
8. Andras Horvath and Miklos Telek. Approximating heavy tailed behaviour with phase type distributions. In *Advances in Algorithmic Methods for*

Stochastic Models, G. Latouche and P. Taylor (eds.), pages 191–214. Notable Publications, 2000.
9. Kathleen S. Meier-Hellstern. A fitting algorithm for Markov-modulated Poisson processes having two arrival rate. *European Journal of Operations Research*, 29:370–377, 1987.
10. Tobias Ryden. Parameter estimation for Markov modulated Poisson processes. *Communications in Statistics-Stochastic Models*, 10(4):795–829, 1994.
11. A. Horvath, G. Rozsa, and M. Telek. A MAP fitting method to approximate real traffic behaviour. In *Proceedings of the IFIP Workshop on Performance Modelling and Evaluation of ATM & IP Networks*, pages 32/1–12, Ilkley, UK, 2000.
12. Lothar Breuer. Parameter estimation for a class of BMAPs. In *Advances in Algorithmic Methods for Stochastic Models,* G. Latouche and P. Taylor (eds.), pages 87–97. Notable Publications, 2000.
13. Lothar Breuer. An EM algorithm for batch Markovian arrival processes and its comparison to a simpler estimation procedure. Preprint, 2001.
14. Tobias Ryden. Statistical estimation for Markov-modulated Poisson processes and Markovian arrival processes. In *Advances in Algorithmic Methods for Stochastic Models,* G. Latouche and P. Taylor (eds.), pages 329–350. Notable Publications, 2000.
15. Cathy H. Xia, Zhen Liu, Mark S. Squillante, Li Zhang, and Naceur Malouch. Traffic modeling and performance analysis of commercial web sites. Preprint, 2001.
16. Arun K. Iyengar, Mark S. Squillante, and Li Zhang. Analysis and characterization of large-scale web server access patterns and performance. *World Wide Web*, 2, June 1999.
17. John D. C. Little. A proof of the queuing formula $L = \lambda W$. *Operations Research*, 9:383–387, 1961.
18. Marcel F. Neuts. The caudal characteristic curve of queues. *Advances in Applied Probability*, 18:221–254, 1986.
19. E. Seneta. *Non-Negative Matrices and Markov Chains.* Springer Verlag, New York, second edition, 1981.
20. Nigel G. Bean, Jian-Min Li, and Peter G. Taylor. Caudal characteristics of qbds with decomposable phase spaces. In *Advances in Algorithmic Methods for Stochastic Models,* G. Latouche and P. Taylor (eds.), pages 37–55. Notable Publications, 2000.
21. Marcel F. Neuts. *Matrix-Geometric Solutions in Stochastic Models: An Algorithmic Approach.* The Johns Hopkins University Press, 1981.
22. Marita Olsson. The EMpht-programme. Technical report, Depart-

ment of Mathematics, Chalmers University of Technology, June 1998. http://www.maths.lth.se/matstat/staff/asmus/pspapers.html.
23. L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, February 1989.
24. B. Sin and J. H. Kim. Nonstationary hidden Markov model. *Signal Processing*, 46:31–46, 1995.
25. S.V.Vaseghi. State duration modeling in hidden Markov models. *Signal Processing*, 41:31–41, 1995.
26. Y. K. Park, C. K. Un, and O. W. Kwon. Modeling acoustic transitions in speech by modified hidden Markov models with state duration and state duration-dependent observation probabilities. *IEEE Transactions on Speech and Audio Processing*, 4(5):389–392, September 1996.
27. M. J. Rusell and R. K. Moore. Explicit modeling of state occupancy in hidden Markov models for automatic speech recognition. In *Proceedings of ICASSP85*, pages 5–8, 1985.
28. S. E. Levinson. Continuously variable duration hidden Markov models for automatic speech recognition. *Computer Speech and Language*, 1(1):29–45, 1986.
29. C. Mitchell and L. Jamieson. Modeling duration in a hidden Markov model with the exponential family. In *Proceedings of ICASSP93*, pages 331–334, 1993.
30. J. D. Ferguson. Variable duration models for speech. In *Proceedings of the Symposium on the Application of Hidden Markov Models to Text and Speech*, pages 143–179, October 1980.
31. Leonard Kleinrock. *Queueing Systems Volume I: Theory.* John Wiley and Sons, 1975.