RESEARCH ARTICLE

Accuracy vs Efficiency of Hyper-exponential Approximations of the Response Time Distribution of MMPP/M/1 Queues

Paolo Romano[a∗], Bruno Ciciani[b], Andrea Santoro[c], Francesco Quaglia[b]

[a] *INESC-ID, Rua Alves Redol, 9, 1000-029 Lisbon, Portugal*;
[b] *Dipartimento di Informatica e Sistemistica, Sapienza Università di Roma, Via Ariosto 25, 00185 Rome, Italy*
[c] *Ente per le Nuove Tecnologie, l'Energia e l'Ambiente, Via Enrico Fermi, 45, 00044 Frascati, Italy*

(*Submitted on the 31st of May 2008*)

Keywords: performability evaluation; Markov Modulated Poisson Process; queuing theory; hyperexponential approximation;

The Markov Modulated Poisson Process (MMPP) has been shown to well describe the flow of incoming traffic in networked systems, such as the Grid and the WWW. This makes the $MMPP/M/1$ queue a valuable instrument to evaluate and predict the service level of networked servers. In a recent work we have provided an approximate solution for the response time distribution of the $MMPP/M/1$ queue, which is based on a weighted superposition of $M/M/1$ queues (i.e. a hyper-exponential process). In this article we address the tradeoff between the accuracy of this approximation and its computational cost. By jointly considering both accuracy and cost, we identify the scenarios where such approximate solution could be effectively used in support of network servers (dynamic) configuration and evaluation strategies aimed at ensuring agreed dependability levels in case of, e.g., request redirection due to faults. Finally the effectiveness of the proposed approximate solution method is evaluated for a real-world case study relying on a trace based traffic characterization of a Grid server.

[∗] Corresponding author. Email: romanop@gsd.inesc-id.pt

1. Introduction

Workload characterization studies of networked systems, such as the Grid and the WWW, have shown that the incoming traffic behavior should rely on models more complex than the traditional Poisson process [3, 8, 15, 19, 26, 30]. Specifically, relevant features of incoming traffic, such as self-similarity and burstiness (e.g. due to request derouting in case of failures) can be effectively captured by the Markov Modulated Poisson Process (MMPP) [25, 27–29, 32], which is a Poisson process whose arrival rate changes according to the evolution of a Markov Chain [10]. On the other hand, in scenarios where relationships between service providers and customers are based on a Service-Level-Agreement (SLA), it would be important to determine the output statistics of $MMPP/M/1$ queues (possibly used as the base for modeling networked servers), via computationally efficient techniques. This might be due to the need for on-line evaluating the effects of (dynamic) system reconfiguration strategies on the achievable level of service [9], in order to still provide agreed dependability levels (e.g. in terms of availability and performance) in the presence of adverse events, such as faults.

In a recent work [5] we have presented a technique allowing analytical approximations of the output distributions (i.e. response time and queue length distributions) of the $MMPP/M/1$ queue. The proposed approximate solution method is based on a weighted superposition of the output distributions of different $M/M/1$ queues. It follows that the approximate response time distributions are expressed as hyper-exponentials obtained by an ad-hoc (linear) combination of the exponential distributions characterizing the response time of $M/M/1$ queues.

Compared to several other works addressing exact solution techniques for the $MMPP/M/1$ queue (based either on matrix geometric methods [2, 24], or on generating functions [12], or on spectral expansion [6, 13], or even on a combination of these approaches [14, 34]), some of the approximations in [5] do not require iterative or numerical methods, e.g., for the determination of matrix eigenvalues/eigenvectors. Hence they can provide benefits for the latency of the analysis. As sketched above, such a computational efficiency might be useful in the context of, e.g., real-time decision making processes aimed at reconfiguring server platforms (for example via request redirection towards a different server instance) in order to ensure adequate service levels [4, 20].

In this paper we aim at studying the accuracy and the actual efficiency (in terms of performance) of those approximation techniques in a wide range of settings. These issues were not addressed (or only partially) in [5] ([1]). To this end, we first draw analytical considerations on the role played by the $MMPP/M/1$ queue parameters in determining the approximation error. On the basis of these considerations and of an experimental sensitivity analysis, we identify the regions within the parameters space where the considered approximations introduce limited or even negligible error. Next, via an experimental study based on diversified implementations of both approximate and exact solution methods, we locate the bottlenecks of exact solution methods and quantitatively demonstrate the superior scalability of the hyper-exponential approximations. Overall, combining accuracy and performance efficiency results, we allow the identification of scenarios where such approximate models could be effectively used in support of network servers evaluation and (dynamic) configuration activities. Finally, we jointly evaluate the accuracy and efficiency of the proposed approximate solution method in a real

[1] Concerning the accuracy, the work in [5] only evaluates the approximation errors for a single trace-based case study. On the other hand, performance efficiency was not explicitly evaluated.

world case study in which we analyze the performance of a Grid system. This study is based on the exploitation of traces of incoming traffic to Grid servers, which have been shown to match the MMPP model [22].

The remainder of this paper is structured as follows. In Section 2, we shortly recall the hyper-exponential approximation technique. In Section 3, we provide analytic insights on the factors affecting the approximation accuracy. The results of an experimental sensitivity analysis aimed at quantifying the impact of the different queue parameters on the approximation error are presented in Section 4. Section 5 is devoted to comparing the performances of the exact and approximate solution methods as a function of the number of states in the MMPP, when considering a wide range of synthetic scenarios. In Section 6 we evaluate the effectiveness of the proposed solution method in a case study based on the traffic characterization of a real Grid server. Section 7 concludes the paper.

2. Recall on the Approximation Technique

The work in [5] derives stochastic processes which approximate the behavior of the $MMPP/M/1$ queue by exploiting well known results in the context of $M/M/1$ queues. Actually, two types of approximations are provided, namely *lower/upper bound* and *unbiased* approximations, the latter being the simplest type (not requiring complex solution methods) on which we focus the present accuracy/efficiency study.

As in [5], we denote with $(S_1 \dots S_H)$ the H states composing the MMPP, and we use the notation $M_i/M/1$ to refer to a $M/M/1$ queue whose arrival rate is the λ_i associated with the generic state S_i of the MMPP. Also, let the service rate μ be a constant value among all the states S_i. The unbiased approximation is based on pinning the response time and queue length of the $MMPP/M/1$ queue to the corresponding steady state values of the $M_i/M/1$ queue as long as the MMPP stays in state S_i. In other words, the approximation is based on a weighted superposition of classical $M/M/1$ queues.

As an example, let us consider a MMPP composed by two states. The mean number of resident requests at time t in the $MMPP/M/1$ queue is shown in Figure 1.a. The instants T_k, T_{k+1} and T_{k+2} correspond to state transitions in the MMPP. Therefore the evolution of the mean queue length value can be described as follows: each time a state transition occurs there is a transient phase (t_{12} for a transition from S_1 to S_2 and t_{21} for a transition from S_2 to S_1) after which the mean queue length may reach the steady state, if any, of the corresponding $M_i/M/1$ queue. As soon as another state transition occurs, a new transient phase starts.

Actually, the $M/M/1$ queue never reaches steady state, but merely approaches it asymptotically, hence in [5] the queue is considered to have reached steady state when the difference between the mean queue length at time t and its theoretical value at steady state differ by no more than an arbitrarily small value ϵ. Also, given that the approximation technique is constructed on the basis of the steady state statistics of $M/M/1$ queues in different time phases (each one representative of the permanence in a different S_i state of the MMPP arrival process), it is necessary to assume that the values of the involved parameters in each state S_i match steady state analysis assumptions for the corresponding $M/M/1$ queue. This means that the server utilization factor must be less than one in every S_i state. Although this assumption might not be suited for $MMPP/M/1$ queues when framed in generic modeling contexts, its feasibility for the evaluation of, e.g., Quality-of-Service (QoS) oriented networked systems has been justified in [5] by the fact that the related capacity planning methods typically entail admission control strategies [7] aimed

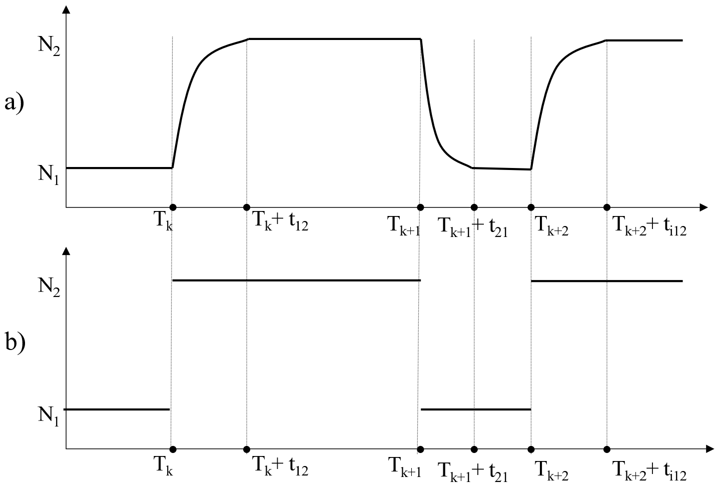


Figure 1. a) Real $MMPP/M/1$ behavior; b) Behavior as modeled by the hyper-exponential approximation.

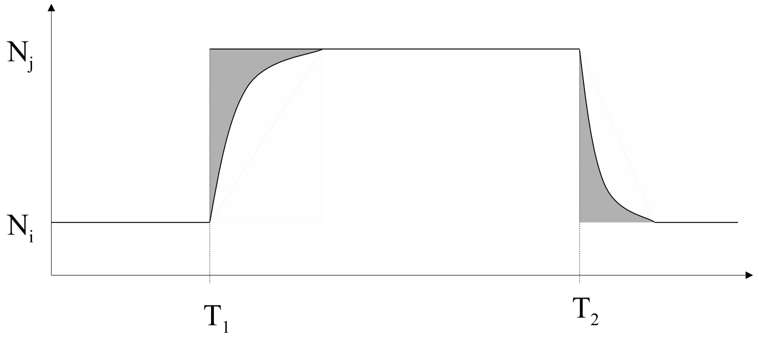


Figure 2. Mean queue length difference between the approximation and the original $MMPP/M/1$ process.

precisely at preventing the utilization factor from exceeding specific bounds.

Given the previous considerations, the unbiased approximation has been derived by adopting the probabilities for the MMPP to stay in each state S_i as the weights of the superposition. Specifically, denoting with Q_i the steady state queue length of $M_i/M/1$ and with p_i the probability for the MMPP to stay in state S_i, the mean queue length of the $MMPP/M/1$ queue can be approximated as $Q = \sum_{i=1}^{H} p_i Q_i$, which would correspond to the case shown in Figure 1.b. A similar technique has been applied in [5] to derive the mean response time of the $MMPP/M/1$ queue, but with a variation. Specifically, the mean response time of the $MMPP/M/1$ queue can still be a weighted sum of the mean response times of $M_i/M/1$ queues, expressed as $R = \sum_{i=1}^{H} w_i R_i$. However, the weights w_i do not simply correspond to the state probabilities of the MMPP (as in the case of the average queue length). They must be scaled to take into account the different arrival rate per each state, which reflects the fact that we are evaluating the response time over samples at discrete time points. Hence $w_i = \frac{p_i \lambda_i}{\sum_{j=1}^{H} p_j \lambda_j}$.

Finally, as far as the cumulative distribution and probability density functions of queue length and response time are concerned, they can also be derived as weighted superpositions of the corresponding functions of the different $M_i/M/1$ queues, the weights being those described above, respectively. Given that by well known queuing theory results the response time of a $M/M/1$ queue is exponentially distributed, this approach gives rise to a hyper-exponential approximation of the distribution function of the response time of the $MMPP/M/1$ queue.

3. Factors Affecting the Approximation Error

The error due to the aforementioned approximation essentially depends on the relative duration of transients periods, caused by a switch from some state S_i to some state S_j of the MMPP arrival process, compared to the permanence time in state S_j. This is because, during those transient periods, the $MMPP/M/1$ queue is characterized by a corresponding steady state representation based on a traditional $M/M/1$ queue.

Going back to the example shown in Figure 1, which depicts the evolution of the mean queue length of the $MMPP/M/1$ queue over time, the error can be represented via the grayed out areas associated with both ramp-up and ramp-down periods caused by state transitions of the MMPP arrival process (see Figure 2). We note that the shown case refers to a typical scenario where the variation of the mean queue length suffers from no "elongation" phenomenon [16], which would lead to non-monotonic ramp-up/ramp-down of the average number of queued requests when an increase/decrease of the arrival rate occurs. However, even in case such an unlikely phenomenon occurs, a similar reasoning could be applied, which would express the fact that the error during the transient ramp-up (resp. ramp-down) phase could switch from positive to negative (resp. from negative to positive) value.

We use the notation T_{S_j} to denote the permanence time in state S_j, and $T_{tr_{i,j}}$ to identify the "hypothetical" duration of the transient period when switching from state S_i to state S_j, namely the time frame starting upon the occurrence of a transition from state S_i to state S_j, which ends as soon as the queue output statistics become time-independent (i.e. they converge towards the statistics of the corresponding $M_j/M/1$ queue). This duration is hypothetical since we might have a new transition in the MMPP arrival process before time independent behavior is reached. However, the length of the hypothetical transient period, as defined above, is representative of the error in the approximation since slow convergence (reflected by a longer hypothetical transient period) means higher impact of, e.g., the grayed out areas representing the error in the evaluation of the mean queue length.

Given that the behavior of the $MMPP/M/1$ queue is approximated with the steady state behavior of the $M_j/M/1$ queue while the MMPP arrival process is in state S_j, the function f defining the error value can be expressed as:

$$Error \simeq f(\frac{T_{tr_{1,2}}}{T_{S_2}}, \frac{T_{tr_{2,1}}}{T_{S_1}}, \ldots, \frac{T_{tr_{H,H-1}}}{T_{S_{H-1}}}, \frac{T_{tr_{H-1,H}}}{T_{S_H}}) \tag{1}$$

The average permanence time T_{S_j} in state S_j can be straightforwardly computed once the transition rate from state S_j to whichever state S_k, denoted as $\alpha_{j,k}$, is known [17]. Specifically, it can be expressed as:

$$T_{S_j} = \frac{1}{\sum_{k \neq j}^{H} \alpha_{j,k}} \tag{2}$$

In order to determine $T_{tr_{i,j}}$ we can compare the mean queue length $N(t)$ at time t (where $t = 0$ is used to represent the occurrence time of the transition towards S_j) with the steady state mean queue length of the $M_j/M/1$ queue associated with state S_j, which we denote as N_{S_j}. Since $N(t)$ converges to N_{S_j} only after infinite time (assuming that no transition in the arrival process takes place in the meanwhile), we consider the output statistics of $MMPP/M/1$ queue to have become time-independent at time $T_{tr_{i,j}}$, where:

$$T_{tr_{i,j}} = min\{t \in \mathrm{R}^+ : |N(t) - N_{S_j}| < \epsilon\} \tag{3}$$

with ϵ arbitrarily small.

By basic queuing theory results [17], $N_{S_j} = \frac{\rho_j}{1-\rho_j}$, where $\rho_j = \lambda_j/\mu$. Also, the average queue length $N(t)$ during the transient period can be computed as $N(t) = \sum_{k=0}^{\infty} P_k(t)k$, where $P_k(t)$ is the probability to have k requests in the queue at time t. To compute $P_k(t)$, we exploit the results in [1, 18], which derive analytical expressions for the probability $P_{h,k}(t)$, namely the probability for the queue to contain k requests at time t given that the queue contained h requests at time $t = 0$. Below we report the expression of $P_{h,k}(t)$, as obtained from [18], with just a few minor differences in notation:

$$P_{h,k}(t) = \rho_{_j}^{\frac{(k-h)}{2}} e^{-(\rho_j+1)\mu t}(I_{_{h-k}} - I_{_{h+k}}) + \rho_j^{-h-1} P_{0,h+k+1}(t) \tag{4}$$

where I_n is the modified Bessel function of first kind whose argument is $2\mu t\sqrt{\rho_j}$. By equation (4), we can express $P_k(t)$ as:

$$P_k(t) = \sum_{h=0}^{\infty} P_h(0) P_{h,k}(t) \tag{5}$$

where $P_h(0)$ represents the probability of h queued requests at the time the transition from state S_i to state S_j occurs (in fact $t = 0$ has been taken as the reference time for that transition). We note that expression (5) could be solved iteratively by computing the values of $P_h(0)$ on the basis of a previously occurred transition towards state S_i, and on the elapsed time since the transition occurrence. Anyway, in case the transition from state S_i to state S_j occurs when the $MMPP/M/1$ queue already reached the steady state behavior of the corresponding $M_i/M/1$ queue, then, at the time of the switch to state S_j, the queue contains h requests with time independent probability $P_h = {\rho_i}^h(1-\rho_i)$. In this case, $P_k(t)$ can be evaluated with no need for iterating on the evaluation of $P_h(0)$ values via the following expression:

$$P_k(t) = \sum_{h=0}^{\infty} {\rho_i}^h (1 - \rho_i) P_{h,k}(t) \tag{6}$$

Consequently we get the following expression:

$$|N(t) - N_{S_j}| = \left| \left[\sum_{k=0}^{\infty} k \sum_{h=0}^{\infty} {\rho_i}^h (1 - \rho_i) P_{h,k}(t) \right] - \frac{\rho_j}{1 - \rho_j} \right| \tag{7}$$

Analytically determining the transient duration, $T_{tr_{i,j}}$ would require identifying a closed form for expression (7), which, at the best of our knowledge, is currently not known. However, since expression (7) depends exclusively on ρ_i, ρ_j and μ, we have conducted a sensitivity analysis, based on a numerical solution approach, which allowed us to highlight the impact of the above parameters on transient duration. In Figure 3 we plot the value of $T_{tr_{i,j}}$ as a function of μ, and consider a set of $\Delta\rho = |\rho_i - \rho_j|$ values which allow us to widely span in the interval [0,1). Note that the plots (which were obtained by setting $\epsilon = 0.1 \times |N_{S_i} - N_{S_j}|$) report, for each $\Delta\rho$ value, the time needed to reach the steady state behavior after the MMPP

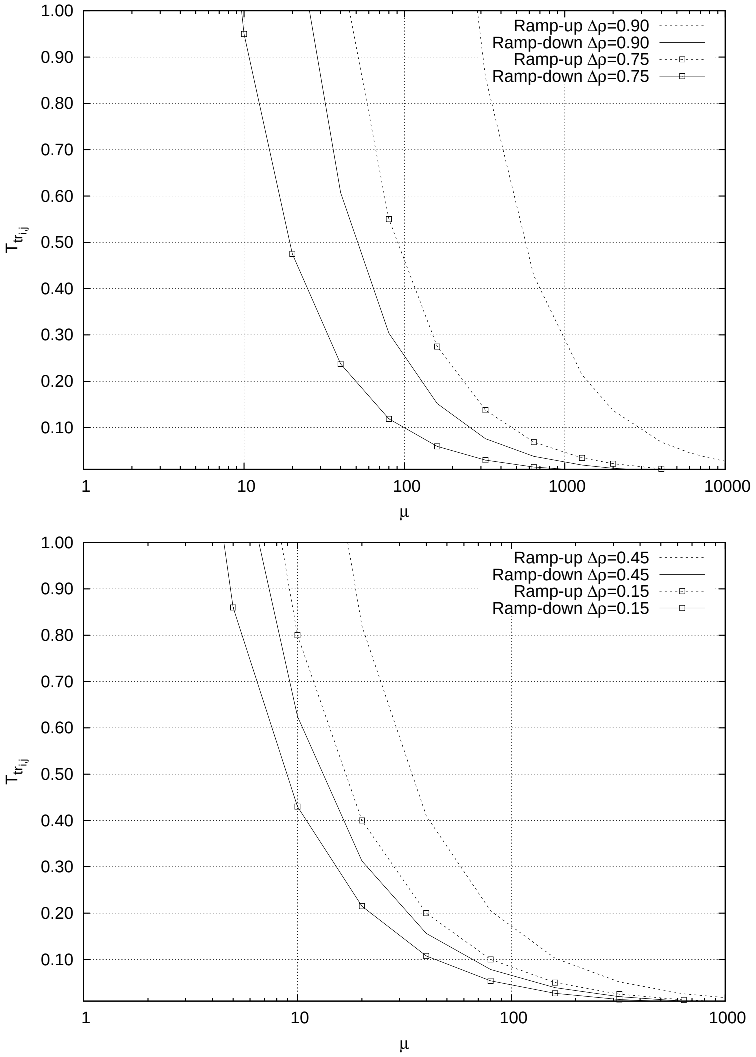


Figure 3. Steady state behavior convergence delay when the MMPP arrival process switches from S_i to S_j.

arrival process has switched to S_j for both ramp-up and ramp-down transitions. From these plots we can deduce the following three considerations:

- Once the values of ρ_i and ρ_j are fixed, the duration of transient periods (in absence of further transitions in the arrival process) rapidly decreases as μ increases. More precisely, we found that $T_{tr_{i,j}}$ can be very closely fitted by means of a hyperbola of equation $\frac{k}{\mu}$, whose k parameter depends exclusively on ρ_i and ρ_j. ([1]). Such a fitting is shown in Figure 4. This leads to the following expression:

$$T_{tr_{i,j}} \approx \frac{k}{\mu} \propto \frac{1}{\mu} \tag{8}$$

which, together with expression (2), allows us to re-write expression (1) as fol-

[1] Fitting, i.e. determining the value of the hyperbola's k parameter, was obtained via the nonlinear least-squares (NLLS) Marquardt-Levenberg algorithm, which converged after very few iterations and showed negligible residual error.

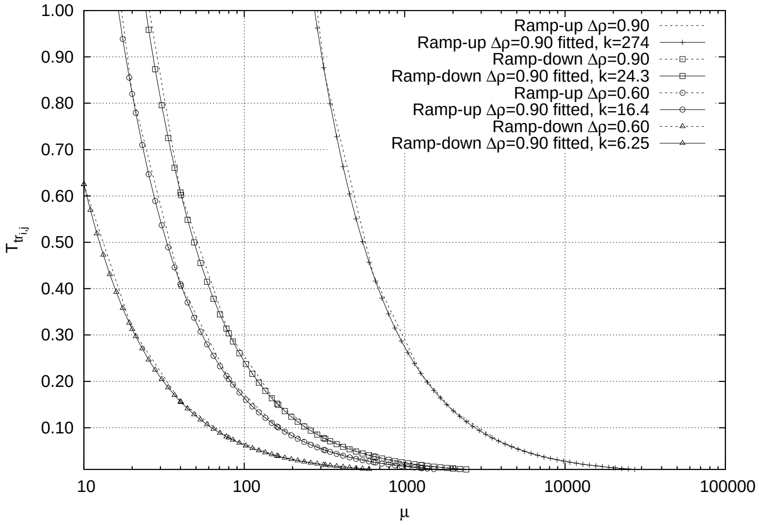


Figure 4. Fitting $T_{tr_{i,j}}$ through hyperbolas of equation $\frac{k}{\mu}$.

lows:

$$Error \simeq f(\frac{\sum_{k\neq 1}^{H} \alpha_{1,k}}{\mu}, \ldots, \frac{\sum_{k\neq H}^{H} \alpha_{H,k}}{\mu}) \tag{9}$$

Therefore, without loss of generality, in the quantitative study we will evaluate the approximation error as a function of the ratio $\frac{\sum_{k\neq j}^{H} \alpha_{j,k}}{\mu}$.

- Concerning the effects of the values of two generic utilization factors ρ_i and ρ_j when a switch from S_i to S_j occurs in the MMPP arrival process, their difference directly influences the distance among the expected queue lengths in states S_i and S_j. As a consequence, it also has an impact on the duration of the corresponding transient periods. Therefore, utilization factors will be treated as independent parameters in the quantitative study, so to evaluate the approximation error in different settings for what concerns the load associated with the different states of the MMPP arrival process.
- Convergence towards steady state in ramp-up transitions is slower than in the opposite ramp-down transitions. This is true independently of the considered $\Delta\rho$ and μ values. At the light of such an observation, since the error due to the approximation leads to an overestimation of queue length and response time during ramp-up transitions, while it leads to an underestimation during ramp-down transitions, the overall approximation error tends towards overestimation.

4. Evaluation of the Approximation Accuracy

This section aims at identifying the regions within the $MMPP/M/1$ parameters space where the approximate solution recalled in Section 2 actually provides negligible error. On the basis of the discussions and deductions in Section 3, we have conducted the analysis while varying the parameters μ, ρ_i and $\alpha_{i,j}$, as well as the number of states composing the MMPP arrival process. Also, the approximation error is evaluated by comparing the output statistics of the approximate model with those obtained by the implementation of the exact solution technique in [14].

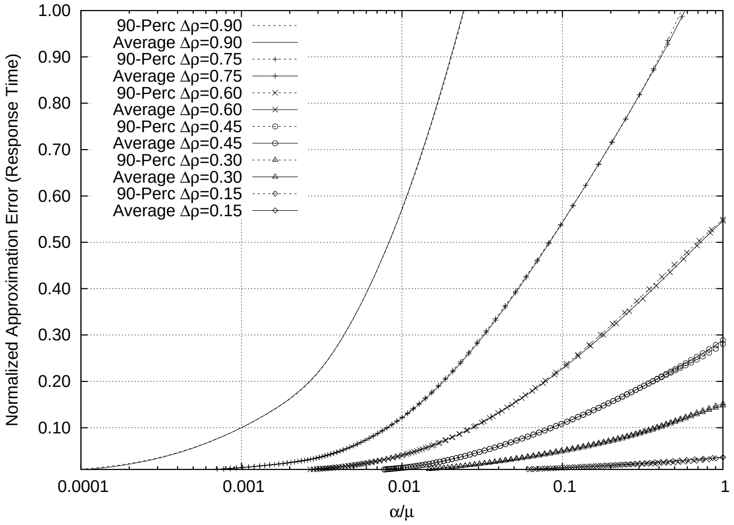


Figure 5. Approximation error on response time and 90-th percentile.

4.1 Two-state MMPP

We start our analysis by focusing on a simple MMPP having two states. Actually, a two-state MMPP can be viewed as the fundamental building block for more complex configurations. Hence, understanding the variation of the approximation error in such a basic configuration is functional to the extension of our analysis to more complex MMPPs.

In Figure 5 we report the normalized error value due to the approximation for both the average response time and the 90-th percentile of the response time distribution of the $MMPP/M/1$ queue while varying both the ratio α/μ and the value of $\Delta\rho$. We set $\alpha = \alpha_{1,2} = \alpha_{2,1}$ in order not to favor stability of the MMPP arrival process in one of the two possible states. Indeed, once fixed the transient duration, such a choice represents a worst case scenario for the hyper-exponential approximation. Specifically, configurations such as, e.g., $\alpha_{1,2} > \alpha_{2,1}$ would have increased the average permanence time in state S_2, thus reducing the error associated with transitions to such a state.

In the plots, the percentage error is identical when considering, for a given configuration, average response time and 90-th percentile of the response time distribution. Also, if the ratio between the value of α and the value of μ is on the order of 10^{-3} or less, then the approximate model ensures a percentage error on the order of 10% or less, independently of the range of variation $\Delta\rho$ of the utilization factor. Anyway, for variations bounded by 0.75 or less, the error rapidly decreases. This occurs also for lower values of the ratio between α and μ. We note that the case of $\alpha/\mu = 0.01$ is representative of a scenario where, assuming expected service time of 1 second, then the arrival process remains stable for a time interval of at least 100 seconds, which, in the context of networked systems, might be usual even in rapidly changing load situations. As an example, the trace based analysis in [22] has shown how, in the context of a Grid system, the arrival process can be modeled via a two-state MMPP, and the frequency of change in the request arrival pattern is at least two orders of magnitudes lower than the frequency of job completion. In such a scenario, the error would be bounded by 10% as soon as the utilization factor variation is bounded by 0.75. We omit reporting data concerning the error due to the approximate solution on the average queue length and on the queue length distribution, since they are very similar to the ones shown in Figure 5 for the response time.

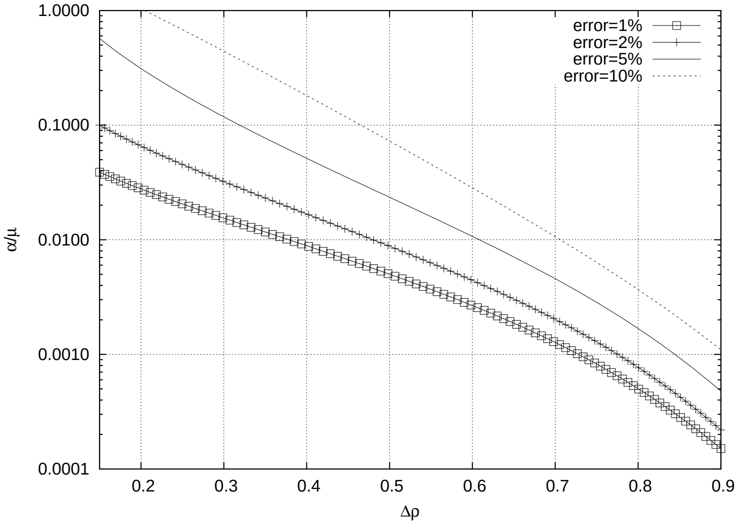


Figure 6. Iso-error curves vs α/μ and $\Delta\rho$.

Table 1. Utilization factors associated with each state of the MMPP.

States	ρ_1	ρ_2	ρ_3	ρ_4	ρ_5	ρ_6	ρ_7	ρ_8
3	0	0.45	0.90	-	-	-	-	-
4	0	0.30	0.60	0.90	-	-	-	-
8	0	0.13	0.26	0.39	0.52	0.65	0.78	0.9

For the reader's convenience we provide in Figure 6 a different data representation. Specifically, we plot iso-error curves associated with specific error values ranging from 1% to 10%. In these curves, we see again that, when $\alpha/\mu = 0.01$, the error value is on the order of 10% for excursion of the utilization factor values up to 0.75. However, in case of a decrease of the ratio α/μ by an additional order of magnitude, the error rapidly decreases towards 1% even for such a large excursion of the utilization factor.

4.2 MMPPs with More than Two States

In this section we extend our study to cover the common scenarios of arrival processes modeled by means of MMPPs with more than two states. For this purpose we consider three different MMPPs with 3, 4 and 8 states, respectively. The corresponding utilization factors, reported in Table 1, are equally distributed in the interval [0,0.9], so to widely span the range of plausible values.

We consider the case of fully connected MMPPs where, from any state, it is possible to switch to any other state. Such a choice allows us to evaluate the tightness of the approximation in the presence of "critical transitions" involving states with largely different utilization factors (as highlighted in the previous sections, the approximation error increases vs larger $\Delta\rho$ values). As in the previous case study, the transition rates $\alpha_{i,j}$ are set to the same value for each state.

In Figure 7 we plot the approximation error on the average response time as a function of α/μ for MMPPs with 3, 4 and 8 states, as well as for the aforementioned two-state case. We omit plotting the error on response time percentiles, since it exhibits very similar behavior. The plots show that, as the number of states in the MMPP increases, the approximation accuracy increases. This can be confirmed also by noting that the curve associated with the two-state MMPP with $\Delta\rho = 0.9$ represents an upper bound on the approximation error for the case of MMPPs with a larger number of states. This can be explained by considering that, as the number of states of the MMPP increases, the average $\Delta\rho$ between the states decreases.

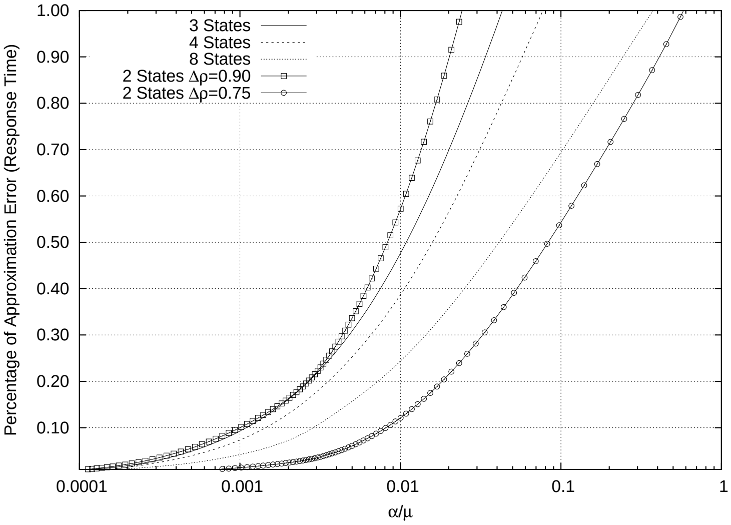


Figure 7. Approximation error on the average response time for MMPPs with more than two states.

Correspondingly, we have a reduced impact of the transition characterized by the maximum excursion of the utilization factor (i.e. $\Delta\rho = 0.9$), which is the one causing the largest part of the approximation error.

Finally, in Figure 8 we show the cumulative distribution function (CDF) of the response time obtained via the analytical approximation and via the exact solution for (i) the 2-state MMPP with $\Delta\rho = 0.9$, (ii) the 4-state MMPP, and (iii) the 8-state MMPP. The curves were obtained considering the value $\alpha/\mu = 10^{-3}$, which, as discussed in Section 4.1, is expected to ensure error less than 10% even for very large values of $\Delta\rho$. The distance between each pair of curves, corresponding to the approximation error, decreases as the number of states in the MMPP grows, thus confirming the deductions derived from the analysis of Figure 7. Further, it is worthy underlining that in every analyzed scenario the CDF obtained via the hyper-exponential approximation represents a consistent underestimation for the actual CDF, thus confirming our intuition in Section 3 concerning the trend of this approximation technique towards the overestimation of the response time (analogous considerations hold for the queue length, even though we omit plotting the corresponding CDFs due to space constraints).

5. Efficiency of the Approximation

In this section we present a performance study aimed at assessing the actual performance gains achievable through the hyper-exponential approximation with respect to the most efficient (at the best of our knowledge) existing exact solution technique, namely the one in [14], which has been already mentioned in the previous section. To this end, we first recall the main algorithmic steps required by the two compared methods. Our focus here, rather than on a detailed description of the two techniques, is on the identification of their main sources of computational cost, which is functional to the subsequent analysis of experimental data concerning the execution times of the two approaches.

Denoting again with H the number of states of the MMPP, the main computational steps required by the exact solution in [14] are the following:

(1) Determine the MMPP equilibrium probabilities by numerically solving a system of $H+1$ linear equations.

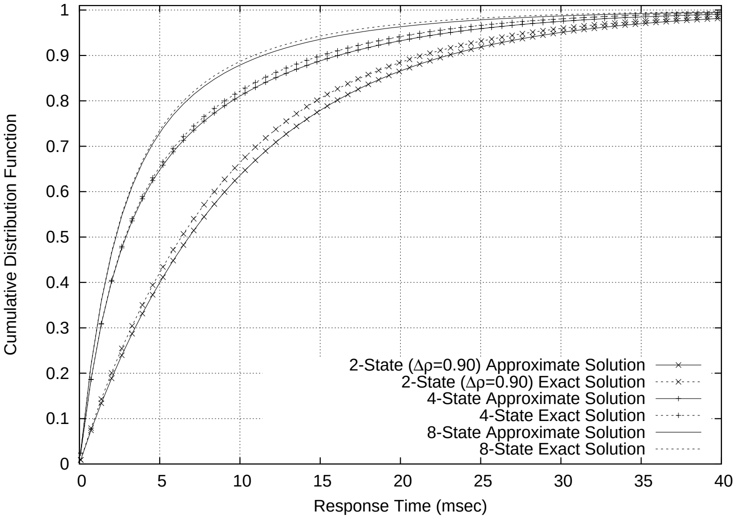


Figure 8. Cumulative distribution function of the response time obtained via the analytical approximation and via exact solution.

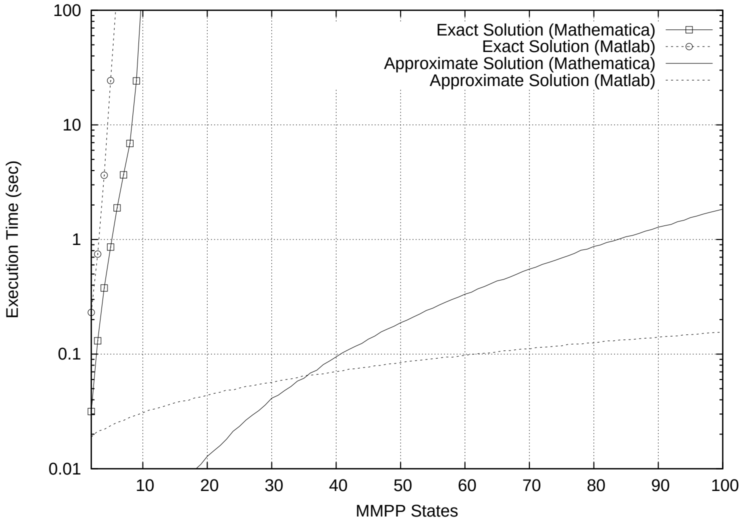


Figure 9. Execution time for the approximate and the exact solution methods vs number of states of the MMPP.

(2) Use the spectral expansion method [21] to derive the steady-state probability distribution of the queue length. This implies (1) numerically computing the eigenvalues/eigenvectors of a sparse square matrix having size $2H \times 2H$, and (2) numerically solving a linear equations system of size $(2H + 1) \times (2H + 1)$

(3) Compute the Laplace transform of the response time distribution. This requires (1) $O(H)$ symbolic operations (i.e. additions and multiplications), as well as two symbolic inversions, involving $H \times H$ polynomial matrices, and (2) symbolically reducing the Laplace transform of the response time distribution into partial fractions.

(4) Pattern match each term resulting from partial fraction decomposition in order to constructively compute the Laplace anti-transform and obtain the response time distribution in the time domain. This can be done via a single iteration over the $O(H^2)$ terms deriving from the partial fraction decomposition.

On the other hand, the main computational steps required by the hyper-

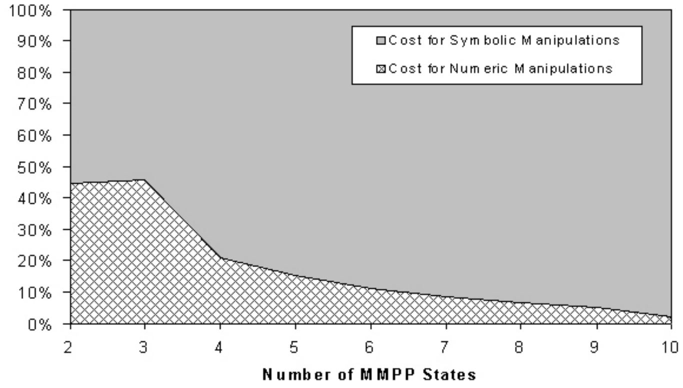


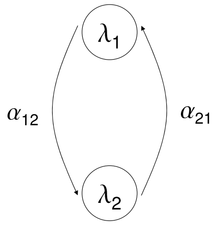
Figure 10. Profiling of the execution costs for the exact solution method (Mathematica-based implementation).

exponential approximation method, as recalled in Section 2, are the following:

(1) Determine the MMPP equilibrium probabilities by numerically solving a system of $H + 1$ linear equations.
(2) Compute the utilization factors ρ_i in each MMPP state as well as the weights w_i for the superposition of the response time distributions of $M/M/1$ queues. This requires a very low number of plain numerical multiplications involving scalars as well as diagonal matrices and vectors of size at most H.

In the light of the above descriptions, the approximate solution method only requires numeric matrix manipulations, whereas the exact solution technique implies potentially onerous steps involving both symbolic and numeric domains. Unfortunately, commercially available mathematical software typically excels only in one of these two domains. Hence, in order to fairly quantify the performance of the exact solution method we developed two implementations relying on diverse mathematical engines, Mathematica 6 [33] and MATLAB 7 [31] specialized in the symbolic and the numeric domain, respectively. The experiments were all carried out on top of a multi-processor equipped with 4 Xeon 2.0 GHz CPUs, 4GB of RAM, running Windows 2003 Server.

In Figure 9 we plot the average execution times for the exact and approximate solution methods while varying the number of states of the MMPP. The curves were obtained by considering a number of randomly generated $MMPP/M/1$ queues, large enough to ensure a confidence interval of 10% around the mean at the 95% confidence level. These curves allow us to draw several considerations. First, they highlight the limited scalability of the exact solution technique in comparison to the hyper-exponential approximation approach. In fact, the execution times of the most efficient exact solution implementation, i.e. the Mathematica-based one, rapidly grow over 100 seconds as the number of states of the MMPP approaches 10. Conversely, the most efficient approximate solution implementation, i.e. the MATLAB-based one, exhibits execution times on the order of a hundred milliseconds even for MMPPs having hundreds of states. These experimental data clearly demonstrate how the approximate approach represents the only viable technique for supporting real-time "what-if" analysis or on-line evaluations of complex $MMPP/M/1$ queuing systems. Note also that, concerning the approximate solution method, its MATLAB-based implementation is favored by the underlying optimized numerical engine, which allows the achievement of execution times one order of magnitude lower in the presence of very large MMPPs. The situation is



MMPP Parameters	
λ_1	22.1
λ_2	7.16
α_{12}	0.17
α_{21}	0.08

Figure 11. MMPP model for each job source (parameter values from [22]).

opposite when comparing exact solution implementations. In this case, the efficient Mathematica symbolic engine allows achieving relatively better performance compared to the MATLAB-based implementation. The latter, in fact, exhibits execution times exceeding 100 seconds much earlier than the Mathematica-based one, i.e. as soon as the MMPP has more than 5 states. In figure 10 we further analyze the sources of latency of the most efficient exact technique implementation, i.e. the Mathematica-based one, identifying the costs imputable to numeric operations (namely steps 1, 2 and 4 of the above description) rather than to symbolic ones (namely step 3 of the above description). **At this end we instrumented our Mathematica-based implementation of the exact solution technique in order to extract detailed profiling information regarding the execution latencies of each of the aforementioned computational steps prescribed by the exact solution approach.** This allowed us to determine that the dominating cost for the exact solution method, while the number of states of the MMPP grows, is associated with the polynomial matrix symbolic manipulations required to derive the Laplace transform of the response time distribution.

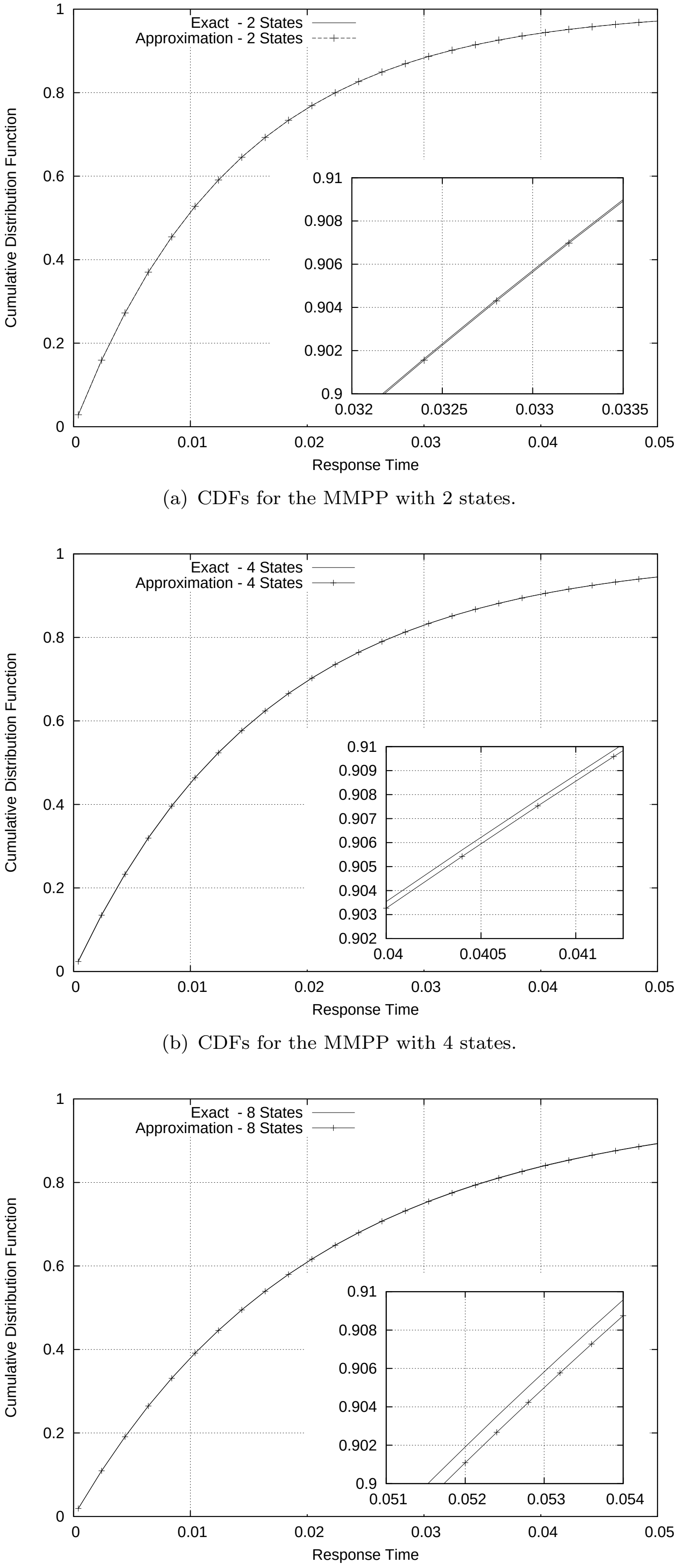
6. A Case Study Based on the Traces of a Grid Server

In this section we aim at jointly evaluating the accuracy and the performance benefits of the proposed model solving approach in realistic settings for what concerns the parameters space of a $MMPP/M/1$ queue.

To this end, we consider three different MMPP arrival processes based on the results reported in [22]. This work has shown, via real traces analysis, the feasibility to model incoming traffic to a Grid server just by means of the MMPP model. Specifically, according to the data reported in [22], the incoming traffic of the analyzed Grid server can be modeled via a two-state MMPP, whose parameters are reported in Figure 11.

On the basis of these parameters we build a test scenario where the response time of the Grid server is evaluated when considering (i) a single source of jobs, (ii) two uncorrelated job sources, and (iii) three uncorrelated job sources. In all the cases, each job source is described on the basis of the previously mentioned trace based study. Also, in terms of MMPP arrival process, the aforementioned cases correspond to situations where the number of MMPP states is equal to 2, 4 and 8, respectively. Note that, while case (i) represents a basic performance analysis scenario, case (ii) and case (iii) may be representative of more critical scenarios where different job sources need to be de-routed to a single Grid site due to critical events (e.g., failures) in the Grid infrastructure. Finally, the Grid server request processing rate has been set to achieve a scenario where the server capacity is saturated at the 75% when the three job sources simultaneously exhibit their peak rate (i.e. $\mu = 88.4$).

The plots in Figure 12 contrast the response time CDF for the three consid-



(a) CDFs for the MMPP with 2 states.

(b) CDFs for the MMPP with 4 states.

(c) CDFs for the MMPP with 8 states.

Figure 12. Evaluating the accuracy of the approximate solution method.

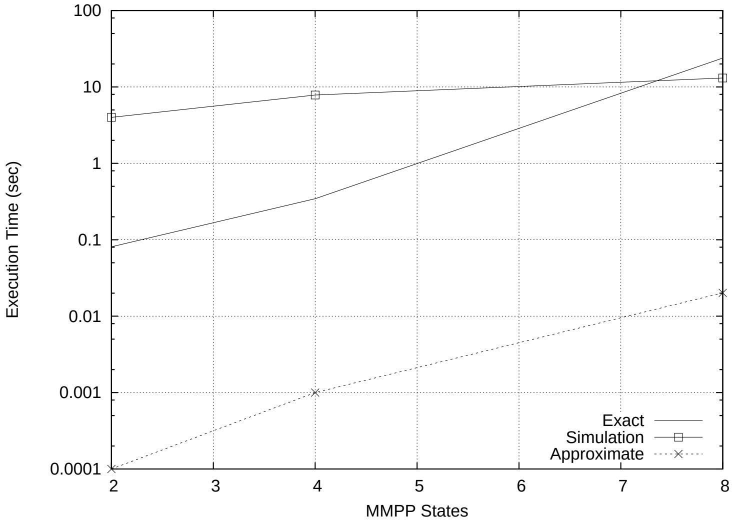


Figure 13. Execution times for the exact and the approximate solution methods.

ered MMPP arrival processes when computed via the approximate, rather than the exact, solution method. For each scenario, we also plot, as an inner figure, a zoom of the response time CDF centered around the 90-th percentile, as this metric is commonly employed, in practice, to establish SLAs. The plots highlight that the approximation error remains always negligible, even in correspondence of the peak arrival rate scenario, where three job sources are simultaneously routed to the Grid server. This confirms the accuracy of the proposed approximation solution technique when employed to assess the performability of realistic computing infrastructures, as well as the validity of the results on the approximation's accuracy derived in the previous section. In fact, in the considered case study, the ratio between the maximum α_{ij} and the server's μ is around 0.002. Also, the maximum utilization factor, which is obtained in the case of three simultaneous job sources producing arrivals at rate λ_1, is equal to 0.75. Then, according to the iso-error curves plotted in Figure 6, we can expect the approximation error not to exceed the 2%. Such an expectation is indeed widely confirmed as in this case study the approximation error on both the 90-th percentile and the expected value of the response time is less than 1% in all the considered scenarios.

In Figure 13, on the other hand, we report the execution times required to determine the response time's CDFs via the approximate and the exact solution methods, as well as, for completeness, via a simulation-based approach. To this end, we have developed an optimized discrete event simulation program for the $MMPP/M/1$ queue, exclusively relying on C technology, whose execution latency was determined by stopping the run as soon as the incrementally computed statistics on the simulated response time CDF vary by no more than 1%. Concerning the exact and the approximate solution methods, the plots were obtained by considering the Mathematica based implementation, namely the implementation that revealed to be more efficient for the exact solution technique (see Section 5). Note that since the MATLAB based implementation of the approximate method is slightly more efficient than the Mathematica based one, this corresponds to slightly favoring the exact solution technique.

Also for this realistic case study, as in the previous performance study in Section 5, the performance benefits achievable through the employment of the approximate solution method are strongly evident, with

execution latencies about three orders of magnitude smaller than for the exact solution technique or the simulation approach. It is worthy highlighting that the solution time of the the exact method (resp. simulation approach) for the 8 states MMPP is about 23 seconds (resp. about 13 seconds), whereas it takes just less than 20 milliseconds for the approximate solution to be computed. These experimental results underline the unfeasibility of employing exact solutions techniques in the context of real-time performability assessment of complex systems, which are, conversely, timely supported by the proposed approximate solution technique.

7. Conclusions

$MMPP/M/1$ queues represent a valuable modeling tool for performance and dependability due to their ability to realistically capture typical features of network traffic, such as self-similarity, burstiness, and long range dependency. The focus of this paper is on investigating the accuracy and computational efficiency of a recently proposed approach [5], which allows to derive a hyper-exponential approximation of the response time distribution of $MMPP/M/1$ queues. The paper's contributions can be summarized as follows.

First, we provided analytical insights on the causes of errors introduced by such an approximate solution technique, which allowed us to identify a few relevant parameters, namely the MMPP state transition rates and the queue service rate, having a major effect on the approximation accuracy.

Next, on the basis of the results of a sensitivity analysis, we isolated the regions within the $MMPP/M/1$ parameters space where the hyper-exponential approximate solution generates minor, or even negligible, deviations with respect to exact solutions.

Then, through the quantitative analysis of the execution times of a number of diverse implementations of both exact and approximate solution methods, we experimentally demonstrated the superior scalability of the proposed approximate approach.

Finally, we demonstrated the effectiveness of the proposed solution method in a realistic case study based on the traffic characterization of a real GRID server.

References

[1] J. Abate and W. Whitt, "Transient Behavior of the M/M/1 Queue Via Laplace Transforms", Advances in Applied Probability, Vol.20, No.1, 1988, pp.145-178.

[2] D. Bini, G. Latouche and B. Meini, "Solving Matrix Polynomial Equations Arising in Queueing Problems", Linear Algebra and its Applications, Vol.340, 2002, pp.225-244.

[3] L. Breslau, P. Cao, L. Fan, G. Phillipps and S. Shenker, "Web Caching and Zipf-like Distributions: Evidence and Implications", Proc. of IEEE INFOCOM, 1999.

[4] V. Cardellini, E. Casalicchio, M. Colajanni and S. Tucci, "Mechanisms for Quality of Service in Web Clusters", Computer Networks, Vol.37, No.6, 2001, pp.761-771.

[5] B. Ciciani, A. Santoro and P.Romano, "Approximate Analytical Models for Networked Servers Subject to MMPP Arrival Processes", Proc. 6th IEEE International Symposium on Network Computing and Applications (NCA), July 2007.

[6] R. Chakka. "Performance and Reliability Modeling of Computer Systems Using Spectral Expansion", PhD thesis, University of Newcastle upon Tyne, 1995.

[7] X. Chen, H. Chen and P. Mohapatra, "ACES: An efficient admission control scheme for QoS-aware web servers", Computer Communications, Vol.26, No.14, 2003, pp.1581-1593.

[8] M. Crovella and A. Bestavros, "Self-similarity in World-Wide-Web traffic: Evidence and possible causes.", IEEE/ACM Transactions on Networking, Vol.3, No.3, Jun. 1994, pp.226-244.

[9] Y. Diao, B. Ciciani and C. H. Crawford, "Enforcing Quality of Service Using Decentralized Runtime Feedback Control", Proc. of the 29th International Computer Measurement Group Conference, 2003, pp.627-638

[10] W. Fischer and K. Meier-Hellstern, “The Markov-modulated Poisson process (MMPP) cookbook”, Performance Evaluation, Vol.18, No.2, Sep. 1993, pp.149-171.

[11] Y. Fujita, M.Murata and H. Miyahara, “Analysis of Web Server Performance Toward Modeling and Performance Evaluation of Web Systems”, Proc. of IEEE SICON, 1998.

[12] H.R. Gail, S.L. Hantler and B.A. Taylor, “Spectral Analysis of M/G/1 type Markov chains”, RC17765, IBM Research Division, 1992.

[13] P.G. Harrison and Y. Zhang, “Delay Analysis of Priority Queues with Modulated Traffic”, Proc. of th 13th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), 2005, pp.280-287.

[14] P.G. Harrison and H. Zatschler, “Sojourn Time Distributions in Modulated G-Queues with Batch Processing”, International Conference on Quantitative Evaluation of Systems (QEST), 2004, pp.90-99.

[15] A. Horvath and M. Telek, “A Markovian Point Process Exhibiting. Multifractal Behavior and Its Application To Traffic Modeling”, Proc. of MAM4, Adelaide, Australia, 2002.

[16] W. D. Kelton and A. M. Law, “The Transient Behavior of the M/M/s Queue, with Implications for Steady-State Simulation”, Operations Research, Vol.33, No.2, 1985, pp.378-396.

[17] L. Kleinrock, “Queuing Systems”, Volume I: Theory, John Wiley & Sons, 1975.

[18] W. Leguesdron, J. Pellaumail, G. Rubino and B. Sericola, “Transient analysis of the M/M/1 queue”, Advances in Applied Probability, No.25, 1993.

[19] W. Leland, M. S. Taqqu, W. Willinger and D. V. Wilson, “On the self-similar nature of the Ethernet traffic (extended version)”, IEEE/ACM Transactions on Networking, Vol.2, No.1, Feb. 1994, pp.1-15.

[20] D.A. Menasce, “Automatic QoS Control”, IEEE Internet Computing, Vol.7, No.1, 2003, pp.92-95.

[21] I. Mitrani,“Spectral Expansion Solutions for Markov-Modulated Queues”, Performance Evaluation of Complex Systems: Techniques and Tools, Performance Tutorial Lectures, Springer-Verlag, 2002, pp.17–35.

[22] H. Li, M. Muskulus and L. Wolters, “Modeling Job Arrivals in a data-intensive Grid”, Proc. 12th Workshop on Job Scheduling Strategies for Parallel Processing, 2006.

[23] Z. Liu, N. Niclausse and C. Jalpa-Villanueva, “Traffic Model and Performance Evaluation of Web Servers”, Performance Evaluation Journal, 46(2-3), pp.77-100, 2001.

[24] M.F. Neuts, “Matrix Geometric Solutions in Stochastic Models”, John Hopkins Press, 1981.

[25] A. Nogueira, P. Salvador, R. Valadas and A. Pacheco, “Fitting self-similar traffic by a superposition of MMPPs modeling the distribution at multiple time scales”, IEICE Transactions, Vol.E87-B, No.3, 2004, pp.678-688.

[26] V. Paxson and S. Floyd, “Wide Area Traffic: the Failure of Poisson Modeling”, IEEE/ACM Transactions on Networking, Vol.3, No.3, pp.226-244, 1995.

[27] A. Riska, M. Squillante, S. Yu, Z. Liu and L. Zhen, “Matrix-Analytic Analysis of a MAP/PH/1 Queue Fitted to Web Server Data”, Proc. of Int. Conference on Matrix Analytic Methods in Stochastic Models, July 2002.

[28] P. Rodriguez, C. Spanner and E.W. Biersack, “Analysis of Web Caching Architectures: Hierarchical and Distributed Caching”, IEEE/ACM Transactions on Networking, Vol.9, No.4, Aug. 2001, pp.404-418.

[29] P. Salvador, R. Valadas and A. Pacheco, “Multiscale Fitting Procedure using Markov Modulated Poisson Processes”, Telecommunication Systems, Springer, Vol.23, No.1-2, 2003, pp.123-148.

[30] W. Willinger, M.S. Taqqu, R. Sherman and D.V. Wilson, “Self-similarity through high-variability: Statistical analysis of Ethernet LAN traffic at the source level”, IEEE/ACM Transactions on Networking, Vol.5, No.1, Feb. 1997, pp.71-86.

[31] The MathWorks, “MATLAB 7”, 2007.

[32] T. Yoshihara, S. Kasahara and Y. Takahashi, “Practical Time-Scale Fitting of Self-similar traffic with Markov Modulated Poisson process”, Telecommunication Systems, Vol.17, No.1-2, 2001, pp.185-211.

[33] Wolfram Research Inc., “Mathematica Edition: Version 6.0”, 2007.

[34] H. Zatschler, “Performance and Reliability Modeling of Computer Systems Using Spectral Expansion”, PhD thesis, University of London Imperial College of Science, 2004.