Multi-server preemptive priority queue with general arrivals and service times

Alexandre Brandwajn, Thomas Begin

Multi-server preemptive priority queue with general arrivals and service times

Alexandre Brandwajn
Baskin School of Engineering
University of California Santa Cruz
USA
alexb@soe.ucsc.edu

Thomas Begin
LIP UMR CNRS - ENS Lyon - UCB Lyon 1 - INRIA 5668
France
thomas.begin@ens-lyon.fr

ABSTRACT

We present a simple approximate solution for preemptive-resume queues with multiple servers, general (phase-type) service and general (phase-type) interarrival time distributions. In our solution, priority levels are solved one at a time in the order of decreasing priorities. Each priority level is solved approximately using a reduced state description. The complexity of our approximate solution in terms of the number of equations solved grows linearly with the number of servers and priority levels.

We studied a large number of numerical examples with a range of values for mean service times and offered loads across priority levels, varying the number of servers from 8 to 48. Discrete-event simulation was used to assess the accuracy of our approximate solution. Overall, in the case of Poisson and quasi-Poisson arrivals, expected relative error for the mean number of customers in the system was below 2% while the corresponding median relative error was below 0.25%. The good accuracy of our approximation appears to extend to the case of phase-type times between arrivals, with expected relative errors for the mean number in system below 5% even for a Pareto-like distribution of interarrival times with a large coefficient of variation. Our numerical results indicate that the proposed approximation provides a relatively simple and generally accurate approach to preemptive-resume queues with larger numbers of servers and general distributions of service and interarrival times.

Keywords: Multiple servers, priority, preemptive-resume, general service, general arrivals, *Ph/Ph/c/N* queue, reduced-state approximation, linear complexity.

1. INTRODUCTION

Systems with multiple servers in which customers are served according to different priorities can be found in many areas, such as customer service centers [GAN03], airport security checkpoints [DEL13], hospital emergency rooms [LIN14], cloud computing systems [ELL12] or processor management in certain computer Operating Systems [STA04].

Despite the large number of systems that can be viewed as instances of a priority queue with multiple servers, the literature devoted to their theoretical analysis appears rather moderate as the inherent complexity of these queues hinders their analysis. In fact, even the accurate approximate analysis of a queue with multiple servers and general service times without priorities remained an open issue for several decades (cf. Gupta et al. [GUP07]).

With the preemptive-resume service discipline (in which a higher priority customer can interrupt the service of a lower priority customer and the interrupted service resumes from the point of interruption when a server becomes available) a simple exact analytical solution is known in the particular case when there is only one server for all customers [TAK91, ALL90]. To the best of our knowledge, few exact results exist in the case of priority queues with multiple servers even under the simplest assumption of exponentially distributed service times (e.g., service rates for all priority levels must be identical in the solutions proposed by Davis [DAV66] and Kella et. al, [KEL85]).

However, noteworthy progress has been made in the analysis of priority queue with multiple servers over the last decade or so. Recently, Wang et al [WAN15] proposed a novel exact analytical solution for the particular case of two priority levels and exponential service times (*M/M/c*). They obtain the generating function of the number of customers at the lower priority level and their solution becomes cumbersome when the number of servers exceeds 2.

A direct numerical solution of the balance equations of an *M/M/c* queue with priorities quickly becomes unmanageable as the number of priority levels increases. Clearly, general service times can only compound the problem, especially with higher numbers of servers, as illustrated by the difficulty of solving even the much simpler top priority level, which is just an *M/G/c* queue [BRA14]. This drives the development of efficient approximate solutions.

In 2004, Zeltyn et al. [ZEL04] studied the *M/M/c* queue with L mixed priority classes (some preemptive, others non-preemptive). They were able to derive exact and approximate solutions under the constraint that the servers have exponential service times, identical for all priority classes.

Since then, several interesting approaches have been proposed aiming to lift the restriction of identical and exponentially distributed service times in the solution of priority *M/M/c* queues. More recently, Al Hanbali et al. [ALH15] removed the constraint of exponential service times by proposing an approximate solution to evaluate the first two moments of the waiting time in a non-preemptive *M/G/c* priority queue with identical service time distributions over all the classes. In the same year, as mentioned above, Wang et al. [WAN15] relaxed the constraint of having identical service rate over all the classes by providing the exact analysis of a preemptive *M/M/c* queue with two priority classes having different

service rates. However, none of these two approaches handles the case of a priority queue having both non-identical service rates and general service times.

In 2005, Harchol-Balter et al. [HAR05] made a significant step in the analysis of the general case of a priority multi-server queue, by considering servers that combine both non-exponential service times and non-identical service rates over all the classes. Their approach relies on reducing the dimensionality of the underlying Markov chain into a one-dimension Markov chain without truncations. Their results, spanning a range of loads and variability of the service times, show good accuracy. Their solution seems best applicable to systems with a moderate number of servers and classes. Indeed, although in theory their method can handle systems with any number of servers and any number of priority classes, the authors develop another approximation when the number of classes, L, is large by approximating the L-priority system with a two-class priority system. Besides, all numerical results presented in their paper pertain to systems with only two servers.

We have mentioned in this introduction only prior work that appears most relevant to this paper. The interested reader may refer to the paper of Harchol-Balter et al. [HAR05] for a thorough and insightful review of the literature prior to 2005.

In this paper, we focus on preemptive-resume multi-server queues and we propose a conceptually simple approximate solution for such a preemptive priority system with general interarrival and service times. In our solution, priority levels are solved one at a time in the order of decreasing priorities. This makes the computational complexity of our solution linear in the number of levels. Each priority level is solved approximately using a reduced state description so that the complexity of the solution grows linearly with the number of servers.

Thus, the contribution of this paper is to introduce a simple approximate solution for preemptive queues with multiple servers, general service and interarrival times that is computationally scalable both in the number of servers and the number of priority levels. Our approach can accommodate times between arrivals and service times that depend on the number of customers at a given priority level. Additionally, although we focus in this work on preemptive-resume priorities, the proposed approach can be readily applied to multi-server queues with preemptive-restart priority.

This paper is organized as follows. We start by the case of memoryless (i.e., Poisson or quasi-Poisson) arrivals. In Section 2 we describe in detail the system considered and define the main symbols used in the sequel. Section 3 outlines the proposed approximate solution and clearly identifies the approximations made. Section 4 is devoted to the numerical results illustrating the accuracy of our approximation in the case of memoryless arrivals. Section 5 presents the extension of our method to general arrivals. Finally, Section 6 concludes this paper.

2. SYSTEM CONSIDERED - CASE OF MEMORYLESS TIMES BETWEEN ARRIVALS

With memoryless arrivals, the priority queueing system considered is shown in Figure 1. It comprises C homogeneous servers (agents) and arriving customers are divided into L priority classes (levels), numbered $1,...,L$ where level 1 is the highest. Customers of class ℓ arrive according to a quasi-Poisson

process with rate $\lambda_\ell(n_\ell)$ where n_l is the number of level ℓ customers currently in the system. An arriving customer who finds its level queue empty interrupts (preempts) the service of a lower-priority customer, if any. The interrupted customer may resume its service at the point of interruption (preemptive-resume) or restart a whole new service period (preemptive-restart). We do not consider the case in which the interrupted customer repeats an identical service period (preemptive-repeat identical). The number of customers at each priority level is limited to N_ℓ. Customers arriving to find their respective queue at capacity are simply lost.

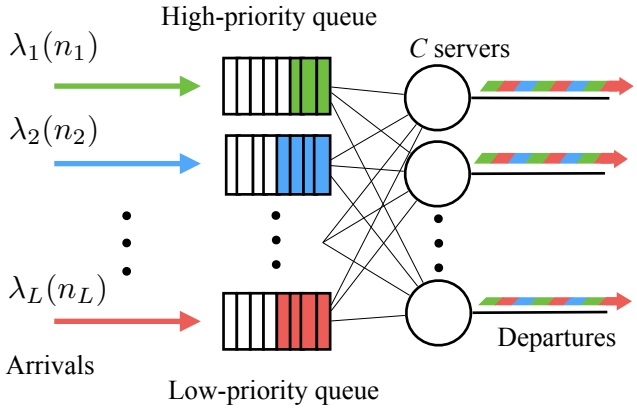

Figure 1. The priority queueing system with L priority levels and C servers.

The service time distribution at each priority level can be different. Customers at a given level are assumed to be statistically identical and the queueing discipline is assumed to be FCFS within each class. The service times at level ℓ ($\ell = 1,...,L$) are distributed according to a phase-type distribution (see Figure 2) with a total of b_ℓ phases. Referring to level ℓ, we denote by σ_{i_ℓ} the probability that service starts in phase i and by μ_{i_ℓ} the intensity of the corresponding phase. The probability that the service proceeds in phase j following the completion of phase i is given by q_{ij_ℓ} ($i = 1,...,b_\ell$, $j = 1,...,b_\ell$) and the probability that the service ends with the completion of phase i is denoted by $\hat{q}_{i_\ell}$. The principal notation used in this paper is summarized in Table 1.

Note that any distribution can be represented arbitrarily closely by a phase-type distribution [BOL05]. If only the first two moments of a distribution are known, and if the distribution's squared coefficient of variation is greater than 0.5, a phase-type distribution with only two phases ($b_\ell = 2$) suffices to match the known first two moments. If more moments are known or one is matching a whole (theoretical or empirical) distribution, a good fit will typically require many more than two phases. Also, distributions with a squared coefficient of variation below 0.5 require more than 2 phases. (Recall that the coefficient of variation of a distribution is defined as the ratio of the standard deviation to the mean.) Readily available tools exist to effect such distribution fitting (e.g. [BOB05, OSO06]).

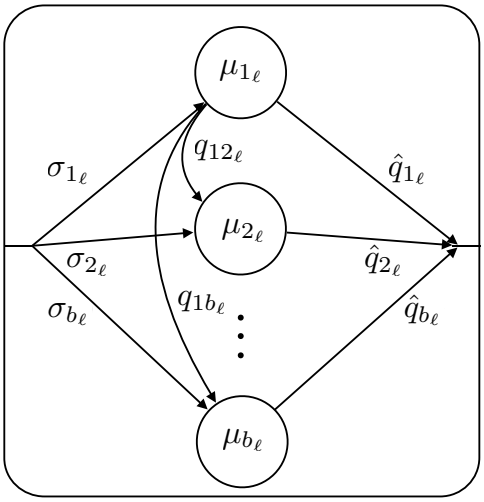

Figure 2. The phase-type distribution with b_ℓ phases for service times of priority level ℓ.

The performance indices of interest include customary performance metrics such as the mean number of customers at each level, the mean response (sojourn) time, loss probabilities, the server utilization or customer throughput for each class, etc.

In the next section, we outline an efficient approximate solution to compute the performance metrics of interest.

3. EFFICIENT APPROXIMATE SOLUTION

A classical state description for the priority system considered is the joint probability of the numbers of customers at each priority level and the number of customers in each phase of service (including customers whose service has been suspended due to preemption by higher priority classes). It allows one to generate directly the full balance equations of the system. However, the number of states in this full description grows combinatorially with the number of priority classes and servers in the systems, making such a direct description unmanageable. Since, with the preemptive priority considered, a given priority level is only affected by higher priority levels, we elect to look at a single level at a time, starting from the top level.

The top level is simply an instance of the *M/Ph/c/N* queue and we solve it approximately using the reduced state description (n_1, i_1) where n_1 is the current number of customers at this level and i_1 describes the current service phase of a selected service position (see [BRA14]). For a system with preemptive-resume priority, we describe the state of a priority level ℓ ($\ell = 2,\ldots,L$) by the triple (n_ℓ, m_ℓ, i_ℓ) where n_ℓ is the current number of customers at this level, m_ℓ is the number of servers (agents) currently unavailable at this level (busy serving higher priority customers) and i_ℓ describes the progress of the service at the current level. Following the idea of reduced state description [BRA14], we describe explicitly the progress (phase number) of the service of only one arbitrarily selected service position, so that $i_\ell = 1,\ldots,b_\ell$ if the service position is currently active at this level (i.e., an agent is serving

the level ℓ customer at this position), $i_\ell = -1,...,-b_\ell$ if the level ℓ customer at this position is currently suspended (i.e., the level ℓ customer at this position is preempted by a higher level customer) , and we use the value $i_\ell = 0$ to describe a service position without a level ℓ customer. The possible values for the number of unavailable servers are $m_\ell = 0,...,C$, and for the current number of level ℓ customers in the system $n_\ell = 0,...,N_\ell$. Thus, the number of states in our state description for level ℓ is at most $(N_\ell + 1)(2b_l + 1)(C + 1)$ since not all values of i_ℓ are feasible for all sets of n_ℓ and m_ℓ . The set of values for i_ℓ corresponds to the case of preemptive-resume priority discipline. If the service discipline is preemptive-restart, there is no need to keep track of the service phase in which the customer was preempted, so that one "suspended" state suffices and the total number of possible values for i_ℓ is limited to $(b_\ell + 2)$.

As an example, in a system with $C = 6$ agents (servers), considering lower priority level $\ell > 1$, the state $(n_\ell = 4, m_\ell = 4, i_\ell > 0)$ implies that there are a total of 4 customers at level ℓ , the number of servers available for customers at this level is $C - m_\ell = 2$ and the selected service position is currently active, the customer at this service position being in phase i_ℓ of its service. If $i_\ell = 0$, this implies that the selected service position is one of the $\max(C - n_\ell, 0) = 2$ unoccupied service positions. If $i_\ell < 0$, this means in our example that the customer at the selected service position is one of the $\min(n_\ell - C + m_\ell, m_\ell) = 2$ customers currently suspended due to unavailability of servers (preempted by customers at higher priority levels).

Note that in our description each priority level has a total of C service positions and any available server (agent) may serve any service position occupied by a customer. Since there is no affinity between service positions and agents, a customer interrupted by higher priority customers will in general resume its service with a different server than before preemption. At the top level all servers are always available ($m_1 = 0$) and thus we must have $i_1 \geq 0$.

Figure 3 shows the saving in terms of the number of states with the proposed reduced-state description as compared to full-state description for a preemptive-resume queue. In Figure 3a, with a system with $C = 8$, $N_\ell = 64$ for $\ell = 1,...,L$, and $b_\ell = 4$ for $\ell = 1,...,L$, we let the number of priority levels vary between $L = 2$ and 10 . We observe that the reduced-state description leads to several hundreds of states while the full-state description results in several tens of thousands. In Figure 3b, the system under consideration has a fixed number of priority levels of $L = 4$, and $N_\ell = 128$ for $\ell = 1,...,L$, and $b_\ell = 4$ for $\ell = 1,...,L$. Depending on the specific number of servers C , we observe that the difference between the total number of states considered in the reduced-state description and the full-state description amounts to roughly two and three orders of magnitude.

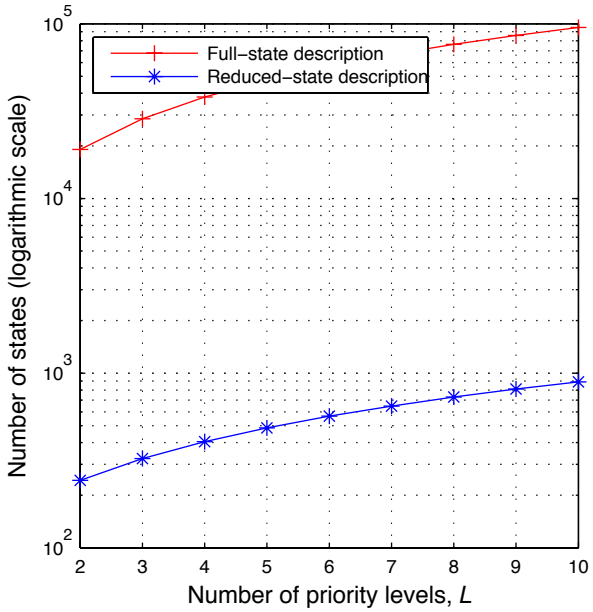

Figure 3a. Saving in number of states using the reduced state description with $C=8$, $N_\ell=64$ and $b_\ell=4$.

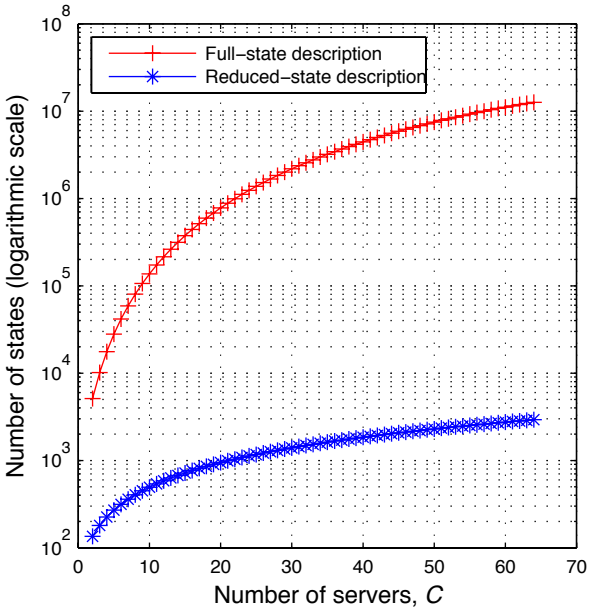

Figure 3b. Saving in number of states using the reduced state description with $L=4$, $N_\ell=128$ and $b_\ell=4$.

Let $p(n_\ell,m_\ell,i_\ell)$ be the steady-state probability of the retained state description at level ℓ. From the perspective of the given priority level, the influence of higher priority levels can be viewed simply as servers (agents) disappearing and reappearing with some rates (corresponding to preemptions and higher priority levels becoming idle). It is a straightforward (albeit somewhat tedious) matter to derive the balance equations for $p(n_\ell,m_\ell,i_\ell)$. As an example, for the case where $n_\ell>C$ and $m_\ell<C$ with $i_\ell=1,\ldots,b_\ell$ we have

$$\begin{aligned}
&p(n_\ell,m_\ell,i_\ell)[\lambda_\ell(n_\ell)+\mu_{i_\ell}+\nu_\ell(n_\ell,m_\ell,i_\ell)+\alpha_\ell(n_\ell,m_\ell,i_\ell)+\beta_\ell(n_\ell,m_\ell,i_\ell)]=\\
&p(n_\ell-1,m_\ell,i_\ell)\lambda_\ell(n_\ell-1)+\sum_{j_\ell=1}^{b_\ell}p(n_\ell,m_\ell,j_\ell)\mu_{j_\ell}q_{j_{i_\ell}}+\sum_{j_\ell=1}^{b_\ell}p(n_\ell+1,m_\ell,j_\ell)\mu_{j_\ell}\hat{q}_{j_\ell}+\\
&p(n_\ell,m_\ell+1,i_\ell)\beta_\ell(n_\ell,m_\ell+1,i_\ell)+p(n_\ell,m_\ell+1,-i_\ell)\beta_\ell(n_\ell,m_\ell+1,-i_\ell)\frac{1}{m_\ell+1}+\\
&p(n_\ell,m_\ell-1,i_\ell)\alpha_\ell(n_\ell,m_\ell-1,i_\ell)\frac{C-m_\ell}{C-m_\ell+1}+p(n_\ell+1,m_\ell,i_\ell)\nu_\ell(n_\ell,m_\ell,i_\ell)
\end{aligned}\qquad(1)$$

Equation (1) involves the known parameters for level ℓ customers as wells as new quantities viz., $\nu_\ell(n_\ell,m_\ell,i_\ell)$, $\alpha_\ell(n_\ell,m_\ell,i_\ell)$ and $\beta_\ell(n_\ell,m_\ell,i_\ell)$. The first quantity $\nu_\ell(n_\ell,m_\ell,i_\ell)$ represents the rate of completions of customers at service positions other than the chosen one (the service progress of the latter is described by i_ℓ) given the current state (n_ℓ,m_ℓ,i_ℓ). $\alpha_\ell(n_\ell,m_\ell,i_\ell)$ denotes the rate with which servers disappear from level ℓ given the current state, and $\beta_\ell(n_\ell,m_\ell,i_\ell)$ is the rate with which servers reappear given that the current state is (n_ℓ,m_ℓ,i_ℓ).

Analogous equations can be obtained for all other values of n_ℓ, m_ℓ and i_ℓ. We must have $\sum_{n_\ell=0}^{N_\ell}\sum_{m_\ell=0}^{C}\sum_{i_\ell=-b_\ell}^{b_\ell} p(n_\ell,m_\ell,i_\ell)=1$. The number of equations in the resulting set of equations is moderate (depending on the value of N_l) and grows only linearly as the number of servers C increases.

Of course, we need to know the rates $\nu_\ell(n_\ell,m_\ell,i_\ell)$, $\alpha_\ell(n_\ell,m_\ell,i_\ell)$ and $\beta_\ell(n_\ell,m_\ell,i_\ell)$ to solve this set of equations. If we had the exact values for these rates, the solution of our set of equations would give us the exact probabilities $p(n_\ell,m_\ell,i_\ell)$. We are not able to obtain the exact values for the unknown rates but we can obtain good approximations by assuming that certain variables in each of these rates are more important than others.

We start by $\alpha_\ell(n_\ell,m_\ell,i_\ell)$, the rate at which servers disappear given (n_ℓ,m_ℓ,i_ℓ). It seems logical to assume that the current number of users at the given level and the service progress at the selected service position have much less influence on the rate α_ℓ than the number of servers already unavailable so that

$$\alpha_\ell(n_\ell,m_\ell,i_\ell)\approx\alpha_\ell(m_l),\quad m_l=0,...,C-1. \tag{2}$$

We make similar assumptions for the rate at which servers reappear at level ℓ given (n_ℓ,m_ℓ,i_ℓ)

$$\beta_\ell(n_\ell,m_\ell,i_\ell)\approx\beta_\ell(m_\ell),\quad m_\ell=1,...,C. \tag{3}$$

Clearly, we have $\alpha_\ell(C)=\beta_\ell(0)=0$.

Note that the solution of the top priority level produces the steady-state probability $p(n_1,i_1)$ since there are no unavailable servers at level 1. The probability that there are a total of n_1 customers at level 1 is given by $p_1(n_1)=\sum_{i_1=0}^{b_1} p(n_1,i_1)$. Since the servers and customers at a given level are assumed to be statistically identical, we readily obtain the overall rate of completions given that the current number of customers is n_1, denoted by $u_1(n_1)$, as

$$u_1(n_1)=C\sum_{i_1=1}^{b_1} p(n_1.i_1)\mu_{i_1}\hat{q}_{i_1}/\,p_1(n_1) \tag{4}$$

The rate of server disappearance for the following level, $\alpha_2(m_2)$, is simply

$$\alpha_2(m_2)=\lambda_1(n_1=m_2)\ \text{ for }\ m_2=0,...,C-1. \tag{5}$$

The rate with which servers reappear at level 2 is given by

$$\beta_2(m_2)=\begin{cases}u_1(n_1=m_2), & m_2=1,...,C-1\\ u_1(n_1=C)p_1(n_1=C)/\sum\limits_{n_1\geq C}p_1(n_1), & m_2=C\end{cases} \tag{6}$$

At level 2, the rate of completions of the selected service position when there are n_2 customers and m_2 servers (agents) unavailable given that the position is not idle can be expressed as

$$\xi_2(n_2,m_2)=\sum_{i_2=1}^{b_2}p(n_2,m_2,i_2)\mu_{i_2}\hat{q}_{i_2}\Big/\sum_{i_2=1}^{b_2}p(n_2,m_2,i_2), \quad n_2=1,...,N_2;\ m_2=0,...,C-1 \tag{7}$$

The rate of completions by service positions other than the selected one can be approximated in terms of $\xi_2(n_2,m_2)$

$$\nu_2(n_2,m_2,i_2)\approx\begin{cases}\min(n_2,C-m_2)\xi_2(n_2,m_2), & i_2\leq 0\\ [\min(n_2,C-m_2)-1]\xi_2(n_2,m_2), & i_2>0\end{cases} \tag{8}$$

At this stage we can solve the balance equations for level 2 to obtain the steady-state probabilities $p(n_2,m_2,i_2)$. Note that, because the rates $\nu_2(n_2,m_2,i_2)$ are effectively expressed in terms of $p(n_2,m_2,i_2)$, the system of equations to solve becomes non-linear. The steady-state probabilities that there are n_2 customers at this level is given by $p_2(n_2)=\sum\limits_{m_2=0}^{C}\sum\limits_{i_2=-b_2}^{b_2}p(n_2,m_2,i_2)$.

Having solved level 2 we use the probabilities $p(n_2,m_2,i_2)$ to assess the rates of server disappearance and reappearance for the immediately following priority level. The rate of server disappearance for level 3 can be obtained as

$$\alpha_3(m_3)=\sum_{m_2=0}^{m_3}\sum_{i_2=-b_2}^{b_2}p(n_2=m_3-m_2,m_2,i_2)[\alpha_2(m_2)+\lambda_2(n_2)]/P_2(m_3),\ m_3=0,...,C-1 \tag{9}$$

where $P_2(m_3)$ denotes the probability that a total of m_3 servers are unavailable for the following level, i.e. busy with customers at level 1 and 2.

The rate of server reappearance at level 3 can be expressed as

$$\beta_3(m_3)=\sum_{m_2=0}^{m_3}\sum_{i_2=-b_2}^{b_2}p(n_2=m_3-m_2,m_2,i_2)[\beta_2(m_2)+n_2\xi_2(n_2,m_2)]/P_2(m_3),\ m_3=1,...,C. \tag{10}$$

The probability $P_2(m_3)$ is given by

$$P_2(m_3)=\begin{cases}\sum\limits_{m_2=0}^{m_3}\sum\limits_{i_2=-b_2}^{b_2} p(n_2=m_3-m_2,m_2,i_2), & m_3=0,...,C-1\\ \sum\limits_{m_2=0}^{C}\sum\limits_{n_2=C-m_2}^{N_2}\sum\limits_{i_2=-b_2}^{b_2} p(n_2,m_2,i_2), & m_3=C\end{cases}. \tag{11}$$

The rates of completions by service positions other than the selected one $\nu_3(n_3,m_3,i_3)$ are computed using formulas analogous to (7) and (8), and the solution of level 3 becomes then possible. We proceed in this way level by level. The rates of server disappearance and reappearance are evaluated at each priority level (except the lowest level) for the solution of the level immediately below it using formulas directly analogous to formulas (9) and (10).

The analysis of priority level ℓ ($\ell=1,...,L$) yields $p_\ell(n_\ell)$ the steady-state probability that there are n_ℓ customers at this level. The mean number of customers at level ℓ is given by $\overline{n}_\ell=\sum\limits_{n_\ell=1}^{N_\ell} n_\ell p(n_\ell)$, the attained throughput of customers at this level can be expressed as $\theta_\ell=\sum\limits_{n_\ell=0}^{N_\ell-1}\lambda_\ell(n_\ell)p(n_\ell)$ and the loss probability can be written as $\xi_\ell=\lambda_\ell(N_\ell)p(N_\ell)/\sum\limits_{n_\ell=0}^{N_\ell}\lambda_\ell(n_\ell)p(n_\ell)$. Knowing the mean number of customers and the throughput, it is easy to obtain the corresponding mean sojourn time and server utilization.

Algorithm 1 summarizes our approach.

Algorithm 1. Solving preemptive-resume queues with multiple servers, general (phase-type) service times and Poisson or quasi-Poisson arrivals.

Step 1. Consider level 1

- Solve the top level to obtain $p(n_1,i_1),\ n_1=0,...,N_1,\ i_1=0,...,b_1$, and $p_1(n_1)$.
- Evaluate performance indices of interest pertaining to level 1.
- Compute $\alpha_2(m_2),\ m_2=0,...,C-1$, and $\beta_2(m_2),\ m_2=1,...,C$ for use in the solution of level 2 (formulas (5) and (6)).

Step 2. Consider levels $\ell=2,...,L$ in the order of decreasing priority. At level ℓ

- Solve the balance equations for the given level using approximation formula (8) to obtain $p(n_\ell,m_\ell,i_\ell)$ and $p_\ell(n_\ell)$.
- Evaluate performance indices of interest pertaining to level ℓ.

- If $\ell < L$, compute $\alpha_{\ell+1}(m_{\ell+1})$ and $\beta_{\ell+1}(m_{\ell+1})$ using formulas (9) and (10).

Note that the proposed approximate solution replaces the solution of a single system of balance equations whose complexity grows combinatorially with the number of priority levels and the number of servers by the solution of L systems of equations whose complexity (in terms of the number of equations) grows only linearly in the number of servers (Figure 3 compares the complexity of the two approaches). We solve the equations for each priority level numerically.

The next section presents numerical results to illustrate the accuracy of the proposed approach in the case of memoryless arrivals.

C	Number of servers (agents)
L	Number of priority levels (classes)
n_ℓ	Number of level ℓ customers currently in the system
$\lambda_\ell(n_\ell)$	Arrival rate of level ℓ customers given there are n_l customers in the system
N_ℓ	Maximum number of customers at priority level ℓ
b_ℓ	Number of phases for the service time distribution of customer of priority level ℓ
σ_{i_ℓ}	Probability that service starts in phase i for customer of priority level ℓ
μ_{i_ℓ}	Intensity of the phase i for customer of priority level ℓ
q_{ij_ℓ}	Probability that the service proceeds in phase j following the completion of phase i for customer of priority level ℓ
$\hat{q}_{i_\ell}$	Probability that the service ends with the completion of phase i for customer of priority level ℓ
m_ℓ	Number of servers (agents) currently unavailable at level ℓ
i_ℓ	Current phase of the service on the selected service position for level ℓ
$p(n_\ell, m_\ell, i_\ell)$	Steady-state probability at level ℓ
$\nu_\ell(n_\ell, m_\ell, i_\ell)$	Rate of completions at service positions other that the selected one given the current state (n_ℓ, m_ℓ, i_ℓ)
$\alpha_\ell(n_\ell, m_\ell, i_\ell)$	Rate with which servers disappear from level ℓ given the current state (n_ℓ, m_ℓ, i_ℓ)
$\beta_\ell(n_\ell, m_\ell, i_\ell)$	Rate with which servers reappear for level ℓ given the current state (n_ℓ, m_ℓ, i_ℓ)
$p_\ell(n_\ell)$	Probability that there are a total of n_ℓ customers at level ℓ
$u_\ell(n_\ell)$	Overall rate of completions given that the current number of customers is n_ℓ
$\xi_\ell(n_\ell, m_\ell)$	Rate of completions by service positions other than the selected one given there are n_ℓ customers and m_ℓ servers (agents) unavailable
$P_\ell(m_{\ell+1})$	Probability that a total of $m_{\ell+1}$ servers are unavailable for level $\ell+1$

Table 1. Principal notation used in the paper in the case of memoryless arrivals.

4. NUMERICAL RESULTS

With memoryless arrivals, the approximation proposed in this paper contains two possible sources of errors. First, even with exponentially distributed service times, there are possible inaccuracies due to the assumption that the rates of server disappearance and reappearance at a lower priority level depend only on the number of servers occupied at higher priority levels (cf. equations (2) and (3) in Section 3). Second, even at the highest priority level where there are no service interruptions, the reduced state description used to account for non-exponential service time distributions introduces possible errors. As indicated by a study of the reduced state description [BRA14], the errors attributable to this approximation are generally small and tend to decrease as the number of servers grows.

To assess the overall accuracy of our approximate level-by-level solution approach, we studied a fairly large set of numerical examples of preemptive-resume queues, using the results of discrete-event simulation as comparison basis. For the latter we used 7 independent replications [MAC89] of between 700,000 and 70,000,000 completions each. These simulation parameters were chosen in an attempt to minimize "warm-up" effects. The resulting estimated confidence intervals at 95% confidence levels tend to be sufficiently small so that we used only the mid-point value.

We studied two different memoryless arrival patterns. In the first one, arrivals to each priority level come from a separate Poisson source with rate λ_ℓ for level $\ell = 1,...,L$. The number of customers at level ℓ is limited to N_ℓ, resulting in possible lost customers.

In the second arrival pattern, as an example of quasi-Poisson arrivals, customers at each priority level come from a separate finite set of memoryless sources with K_ℓ sources for level ℓ. A customer is either at the source or in the priority queue (waiting or being served) and no customers are lost so that we have $N_\ell = K_\ell$. The rate of customer arrivals to level ℓ with n_ℓ customers already present is $(K_\ell - n_l)\phi_\ell$ where $1/\phi_\ell$ is the mean time a customer spends at the source (on each pass through the system).

We start by the case of Poisson arrivals. The numbers of servers considered were $C =$ 8, 16, 32 and 48. The buffer size for each priority level was set to $N_\ell = 3C$. The number of priority levels was kept at $L = 4$ and we used a set of 4 values for the mean service times at different levels. The mean service time for level 1 (highest priority) was set to 1, for level 2 to 1/2 or 2, for level 3 to 1/4 or 4 and for the last level to 1/8 or 8. We considered 4 values for the coefficient of variation of the service times at different levels: 0.5, 1, 2 and 4. The arrival rates for different priority levels ranged from 0.1 to 1.5 per time unit per server. These arrivals rates λ_ℓ were explored so that our results span cases in which different priority classes dominate the system. The above combinations of parameter values amounted to a total of 960 example points for each of the 4 priority levels.

For each priority level, we used the mean number of customers, the attained throughput and the loss probability as performance indices. Tables 2, 3 and 4 summarize the relative errors versus simulation results obtained for the mean number in system, attained throughput and loss probability, respectively.

These tables include the mean (expected) and median relative errors, as well as the distribution of relative errors. Table 2a shows how the accuracy of our approximation generally improves as the number of servers grows. We note that already for 16 servers the mean error is below 2% and the percentage of example points in which the relative error exceeds 10% is less than 5%. In Table 2b, we have included results for each priority level separately and for all levels combined. We observe in Table 2b that, although (not surprisingly) the accuracy of the approximation degrades for lower priority levels, it remains generally quite good. Even for the lowest priority level in our examples, the expected relative error for the mean number of customers in the system remains below 3%, and the percentage of cases in which the relative error was below 10% is over 90. Overall, in the example points considered in our study, the mean error for the mean number of customers was less than 1.5% while the median error did not exceed 0.03%. Table 3 yields similar observations for the attained throughput. Table 4 summarizes the results obtained for the loss probability. Note that in order to avoid potentially large relative errors we have included in Table 4 only example points in which the loss probability (in the simulation) was above 0.01. Here, the mean relative error remains below 1%. The fact that the relative error seems to decrease for lower priority levels appears to be due to larger values of loss probabilities at lower levels, leading to smaller relative errors.

Number servers	Mean (%)	Median (%)	<1%	1-5%	5-10%	>10%
8	2.28	0.05	81.0	10.0	3.2	5.8
16	1.51	0.02	88.6	6.1	1.7	3.6
32	0.98	<0.005	92.1	4.8	0.7	2.4
48	0.80	<0.005	94.1	2.6	1.5	1.8
All	1.39	0.02	89.0	5.9	1.8	3.4

Table 2a. Distribution of the relative errors for the mean number in system with Poisson arrivals.

Class	Mean (%)	Median (%)	<1%	1-5%	5-10%	>10%
1	0.14	0.03	96.5	3.4	0.1	0.0
2	0.79	0.02	86.2	9.7	2.2	1.9
3	1.78	<0.005	88.4	4.3	2.5	4.5
4	2.97	<0.005	84.4	6.0	2.3	7.2
All	1.39	0.02	89.0	5.9	1.8	3.4

Table 2b. Distribution of the relative errors for the mean number in system with Poisson arrivals.

Class	Mean (%)	Median (%)	<1%	1-5%	5-10%	>10%
1	0.04	0.01	99.2	0.8	0.0	0.0
2	1.43	0.01	88.8	5.4	3.0	3.2
3	1.87	0.01	90.6	3.7	0.8	4.9
4	2.28	0.01	90.9	2.4	2.4	4.2
All	1.24	0.01	92.8	3.1	1.4	2.8

Table 3. Distribution of the relative errors for the attained throughput with Poisson arrivals.

Class	Mean (%)	Median (%)	<1%	1-5%	5-10%	>10%
1	0.88	<0.005	96.3	1.4	0.6	1.7
2	0.16	<0.005	96.0	3.1	0.9	0.0
3	0.11	<0.005	96.3	3.5	0.1	0.0
4	0.09	<0.005	96.4	3.4	0.2	0.0
All	0.35	<0.005	96.2	2.7	0.5	0.5

Table 4. Distribution of the relative errors for the loss probability with Poisson arrivals.

Figure 4 shows an example of the behavior of different priority levels in a preemptive-resume queue with Poisson arrivals, $C = 16$ servers and the maximum number of customers at each priority level limited to $N_\ell = 3C$. The mean service times are 1, ½, ¼ and 1/8 for priority levels 1, 2, 3 and 4, respectively, while the coefficient of variation of the service time is kept at 2 for all customer classes. The rates of customer arrivals are given by $\lambda_1 = \lambda$, $\lambda_2 = \lambda/2$, $\lambda_3 = \lambda/4$ and $\lambda_4 = \lambda/8$ for levels 1,2,3 and 4, respectively, and we vary the factor λ to study the performance of customers at different priority levels as a function of overall offered load. We observe the close agreement between simulation and our approximate results.

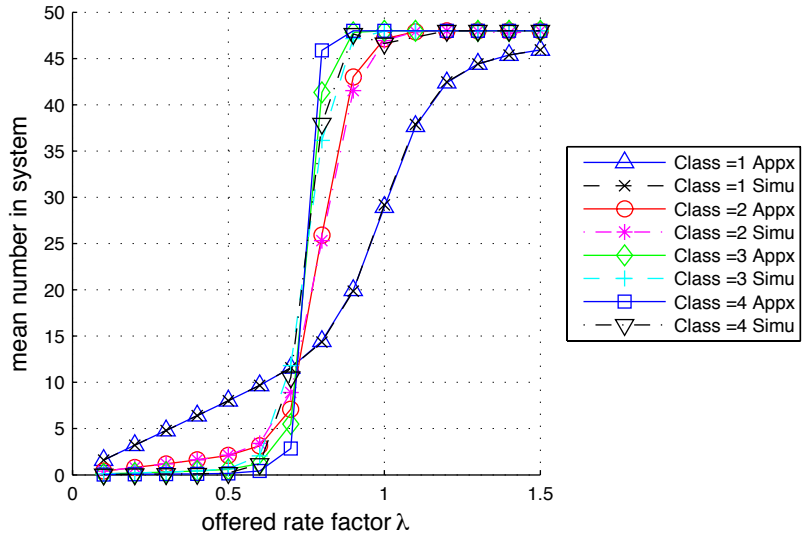

Figure 4a. Mean number in system as a function of the offered rate with Poisson arrivals.

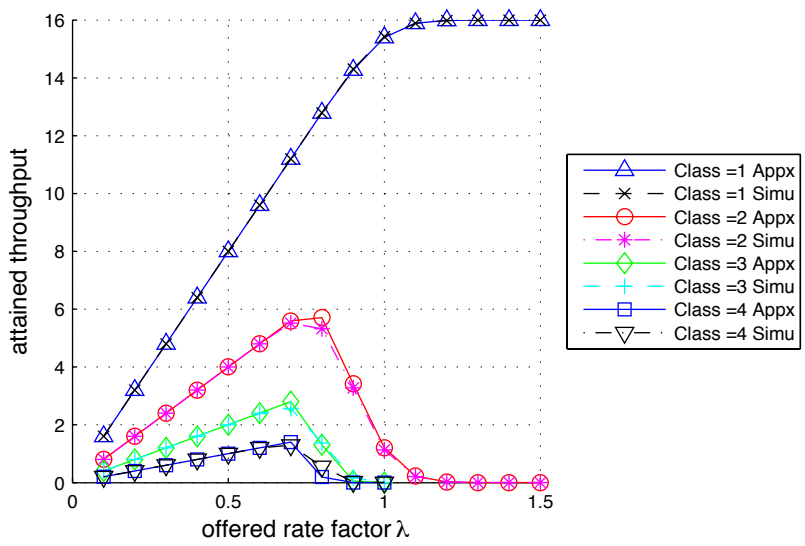

Figure 4b. Attained throughput as a function of the offered rate with Poisson arrivals.

We now consider the second arrival pattern with K_ℓ sources for level ℓ. We used 3 values for the number of sources: $K_\ell =$ 10, 25 and 50. The values of the unitary source rate φ_ℓ ranged from $0.1/K_\ell$ to $0.9/K_\ell$ per time unit. We used the same set of mean service times as in the case of Poisson arrivals. The coefficient of variation of the service times was set to 2. Here, the total number of example points explored was over 10,000. Tables 5 and 6 summarize the relative accuracy of our approximation for the mean number of customers in the system and the customer throughput, respectively. As was the case for Poisson arrivals, we show the mean, median and distribution of relative errors for each priority level, as well as for all levels combined.

We observe that, while the relative errors increase for lower priority levels, the mean error for class 4 remains below 3% in our study, and in over 90% of example points considered the relative errors remain below 10% for this priority class. The median errors are quite small (less than 0.5%). It has been our experience that the infrequent larger relative error tend to occur when the mean service times at higher priority levels are longer than at lower priority levels and when the number of sources at the latter is small.

Class	Mean (%)	Median (%)	<1%	1-5%	5-10%	>10%
1	0.14	0.09	99.5	0.5	0.0	0.0
2	0.59	0.11	90.7	6.2	1.9	1.2
3	1.32	0.15	83.2	10.5	2.7	3.6
4	2.79	0.19	73.9	12.5	5.3	8.3
All	1.04	0.12	88.5	6.6	2.1	2.7

Table 5. Distribution of the relative errors for the mean number in system with discrete sources.

Class	Mean (%)	Median (%)	<1%	1-5%	5-10%	>10%
1	0.10	0.08	100.0	0.0	0.0	0.0
2	0.36	0.10	93.5	5.3	0.8	0.4
3	1.70	0.13	80.7	10.4	3.7	5.2
4	3.28	0.13	76.4	10.6	4.5	8.4
All	1.10	0.10	89.5	5.8	1.9	2.8

Table 6. Distribution of the relative errors for the attained throughput with discrete sources.

Figure 5 shows an example of the behavior of a preemptive-resume priority system with a finite set of memoryless sources in the particular case where the number of customer sources K_ℓ is the same for each of the 4 priority levels considered. In this example, there are $C = 16$ servers, the unitary source rate is set to $\varphi_\ell = 0.5 / K_\ell$, the mean service times are 1, 2, 4 and 8 for priority classes 1, 2, 3 and 4, respectively, while the coefficient of variation of the service time is kept at 2 for all classes.

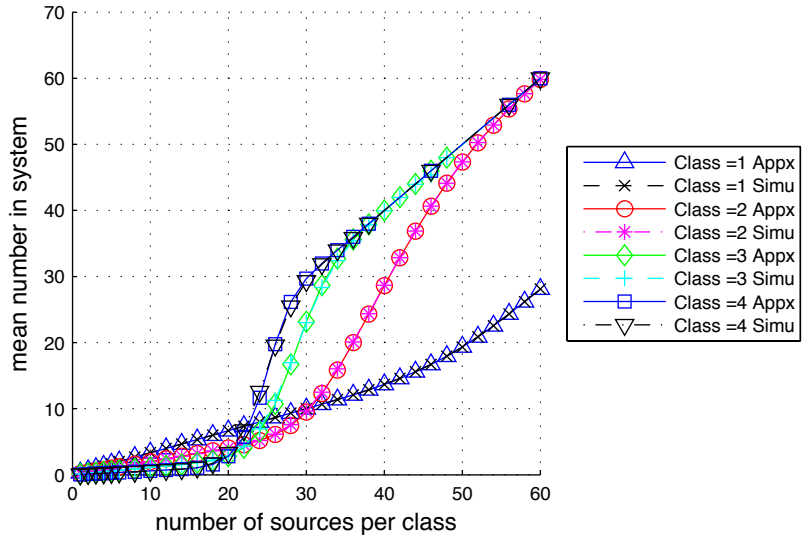

Figure 5a. Mean number in system as a function of the number of sources.

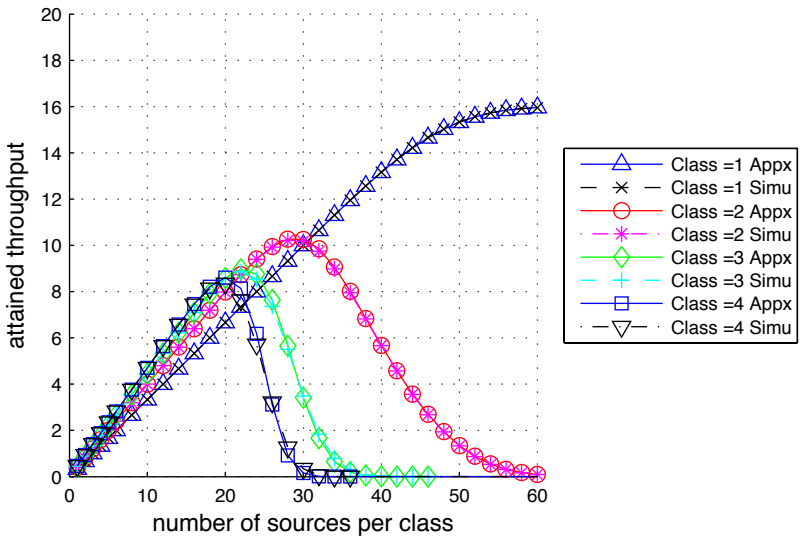

Figure 5b. Attained throughput as a function of the number of sources.

Overall, in the example considered, the results of our approximate solution closely match simulation results. In the next section, we extend our method to include general interarrival time distributions.

5. EXTENSION TO GENERAL ARRIVALS

We now consider a priority system similar to the one described in Section 2 but in which the times between consecutive customer arrivals at each priority level are distributed according to a phase-type distribution, possibly different for each customer class. Specifically, the times between customer arrivals at level ℓ ($\ell = 1,...,L$) are distributed according to a phase-type distribution (see Figure 6) with a total of a_ℓ phases. Referring to level ℓ, we denote by τ_{i_ℓ} the probability that the time between arrivals starts in phase i and by λ_{i_ℓ} the intensity of the corresponding phase. The probability that the service proceeds in phase j following the completion of phase i is given by r_{ij_ℓ} ($i = 1,...,a_\ell$, $j = 1,...,a_\ell$) and the probability that the service ends with the completion of phase i is denoted by $\hat{r}_{i_\ell}$.

To extend our approximate solution of Section 3 to such phase-type distributions of time between arrivals, we note that in our level-by-level approach we are in essence dealing with a multi-server queue in which, with the exception of the highest priority level, servers disappear and reappear with rates dependent on the number of servers currently unavailable to the level considered. Figure 6 shows a single level with phase-type times between arrivals and phase-type service times. Therefore, we propose to extend the approach used recently by the authors for *Ph/Ph/C/N* queues [ATM16]. In this approach such systems are solved by iterating between two simpler models: a model with memoryless state-dependent arrivals and phase-type service (*M/Ph/c/N* queue), and a model with phase-type times between arrivals and memoryless state-dependent service *(Ph/M/c/N queue*). In our case, as shown in

Figure 7, the model with memoryless state-dependent arrivals is in fact the model solved at each level as described in Section 3. Thus, no new solution approach needs to be developed for this part. The *Ph/M/c/N* queue (the other model) can be solved using a simple numerically stable recurrence [BRA14]. The iteration between these two models for each priority level stops when the values for a performance metric such as the mean number of customers obtained from both models become sufficiently close. As discussed in [ATM16], while such a fixed-point iteration between models yields only an approximate solution, the errors it introduces seem generally small. Moreover, typically, the iteration between models tends to converge quite fast.

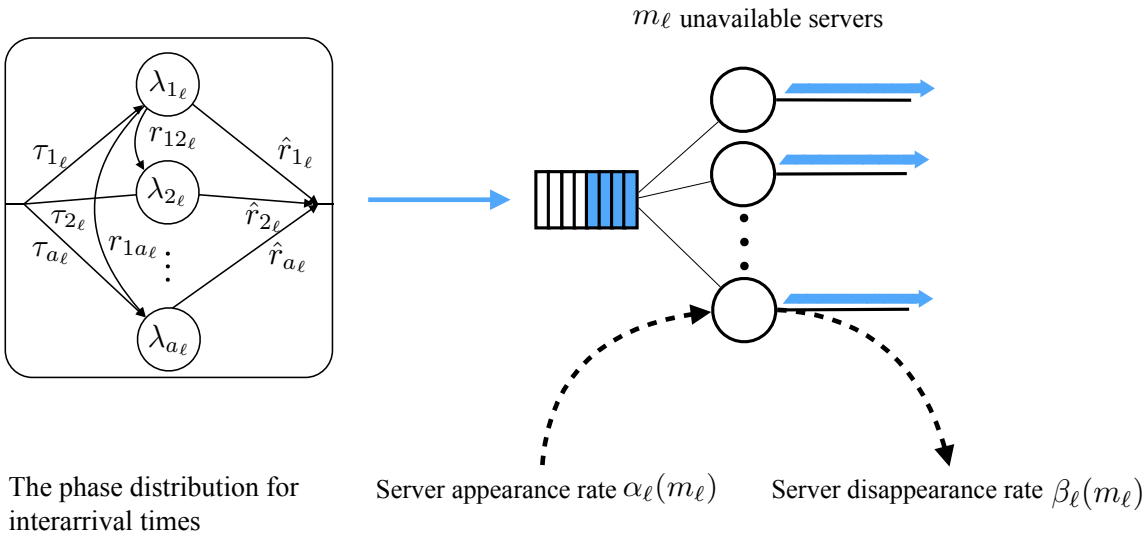

Figure 6. Model of a single level ℓ with phase-type times between arrivals and phase-type service times.

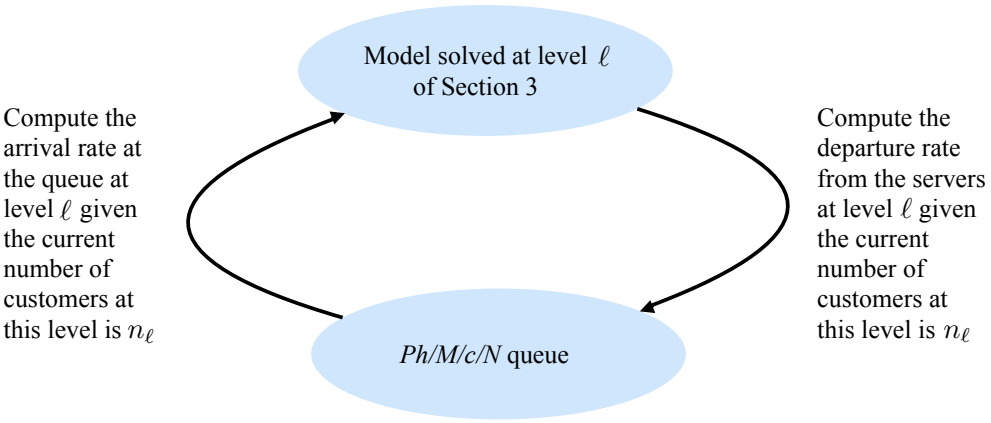

Figure 7. Iteration between a model of a single level with state-dependent arrivals and a *Ph/M/c/N* queue to solve the single level model of Figure 6.

Algorithm 2 summarizes our extended approach.

Algorithm 2. Solving preemptive-resume queues with multiple servers, general (phase-type) service and general interarrival times.

Step 1. Consider level 1

- Use iteration (see Algorithm 3) to obtain approximate values for $p(n_1,i_1),\ n_1=0,...,N_1,\ \ i_1=0,...,b_1$, and $p_1(n_1)$.
- Evaluate performance indices of interest pertaining to level 1.
- Compute $\alpha_2(m_2),\ m_2=0,...,C-1$, and $\beta_2(m_2),\ m_2=1,...,C$ for use in the solution of level 2 (formulas (5) and (6)).

Step 2. Consider levels $\ell=2,...,L$ in the order of decreasing priority. At level ℓ

- Use iteration (see Algorithm 3) to obtain approximate values for $p(n_\ell,m_\ell,i_\ell)$ and $p_\ell(n_\ell)$.
- Evaluate performance indices of interest pertaining to level ℓ.
- If $\ell<L$, compute $\alpha_{\ell+1}(m_{\ell+1})$ and $\beta_{\ell+1}(m_{\ell+1})$ using formulas (9) and (10).

Algorithm 3 summarizes the fixed-point iteration between models at each level, denoting by M the selected performance metric for convergence test.

Algorithm 3. Determining state probabilities at a single level for Algorithm 2.

Step 1. Initialize the arrival rate values $\lambda_\ell(n_\ell)$ to the inverse of the mean time between arrivals.

Step 2. Solve the model with state-dependent memoryless arrivals using the current values of $\lambda_\ell(n_\ell)$.

- Obtain current values for $p(n_\ell,m_\ell,i_\ell)$ and $p_\ell(n_\ell)$, as well as the equivalent service rate $u_\ell(n_\ell)$ as $u_\ell(n_\ell)=\lambda_\ell(n_\ell-1)p_\ell(n_\ell-1)/p_\ell(n_\ell)$.
- Compute current value of M from this model.

Step 3. Solve the *Ph/M/c/N* queue using the current values of $u_\ell(n_\ell)$ from Step 2

- Obtain current values for $p_\ell(n_\ell)$ and $\lambda_\ell(n_\ell)$.
- Compute the current value of M from this model.

Step 4. If the values of M from Step 2 and Step 3 deviate by less than $\varepsilon>0$ then stop the iteration, otherwise go to Step 2.

Step 5. Use the values of $p(n_\ell,m_\ell,i_\ell)$ and $p_\ell(n_\ell)$ from last execution of Step 2 as the solution of level ℓ.

To assess the influence of the variability in the arrival process on the accuracy of the proposed approximate solution, we studied a priority system with four priority levels and three different values of the coefficient of variation of the arrival process, viz. 2, 4 and close to 15. For the latter we used a Pareto-like distribution with 16 phases. In our study we used the same set of values of the mean arrival rates and mean service times as for the case of Poisson arrivals in Section 4 but considered only service times with a coefficient of variation of 2. Table 7 summarizes the relative errors for the mean numbers of customers at each priority level in a system with $C = 16$ servers and the maximum number of customers at each priority level limited to $N_\ell = 3C$. Note that for each value of the coefficient of variation of the time between arrivals we studied 60 example points.

Class	Mean (%)	Median (%)	<1%	1-5%	5-10%	>10%
1	0.21	0.06	95.8	4.2	0.0	0.0
2	0.80	0.08	79.2	16.7	4.2	0.0
3	3.14	0.10	68.8	20.8	2.1	8.3
4	2.97	0.23	73.3	11.1	0.0	15.6
All	1.40	0.05	83.5	10.5	1.3	4.6

Table 7a. Distribution of the relative errors for the mean number in system with a coefficient of variation for interarrival times of 2.

Class	Mean (%)	Median (%)	<1%	1-5%	5-10%	>10%
1	0.29	0.06	89.3	10.7	0.0	0.0
2	1.34	0.12	69.6	23.2	3.6	3.6
3	3.65	0.66	62.5	21.4	7.1	8.9
4	3.19	0.38	64.3	17.9	7.1	10.7
All	1.69	0.07	77.1	14.6	3.6	4.6

Table 7b. Distribution of the relative errors for the mean number in system with a coefficient of variation for interarrival times of 4.

Class	Mean (%)	Median (%)	<1%	1-5%	5-10%	>10%
1	2.99	2.85	3.4	89.8	6.8	0.0
2	3.79	3.06	11.9	74.6	10.2	3.4
3	4.61	2.88	8.5	72.9	8.5	10.2
4	4.72	2.07	18.6	61.0	8.5	11.9
All	3.22	2.75	28.5	59.7	6.8	5.1

Table 7c. Distribution of the relative errors for the mean number in system with a Pareto-like distribution of interarrival times (coefficient of variation of 15).

We observe that the accuracy of the method remains good as the value of the coefficient of variation of the interarrival time doubles from 2 to 4. Even with the large coefficient of variation of the Pareto-like distribution of 16 phases the mean relative errors for the mean numbers of customers in the system remain under 5% for all four priority levels. Interestingly, in the example points considered, the degradation in accuracy for lower priority levels is quite moderate.

As a simple example of the application of the proposed solution, we compare in Figure 8 the mean numbers of customers in a preemptive-resume priority system with two classes of customers with Poisson arrivals versus non-Poisson arrivals. The mean service times are 1 and ½ for priority levels 1 and 2, respectively, with the coefficient of variation of the service times kept at 2 for both customer classes. The mean rates of customers arrivals are λ and $\lambda/2$, for customers at priority levels 1 and 2, respectively. The coefficient of variation of the time between arrivals in the case of non-Poisson arrivals is kept at 4 for both priority classes. The system has $C=8$ servers and the number of customers at each priority level is limited to $N_i=24$ for $i=1,2$.

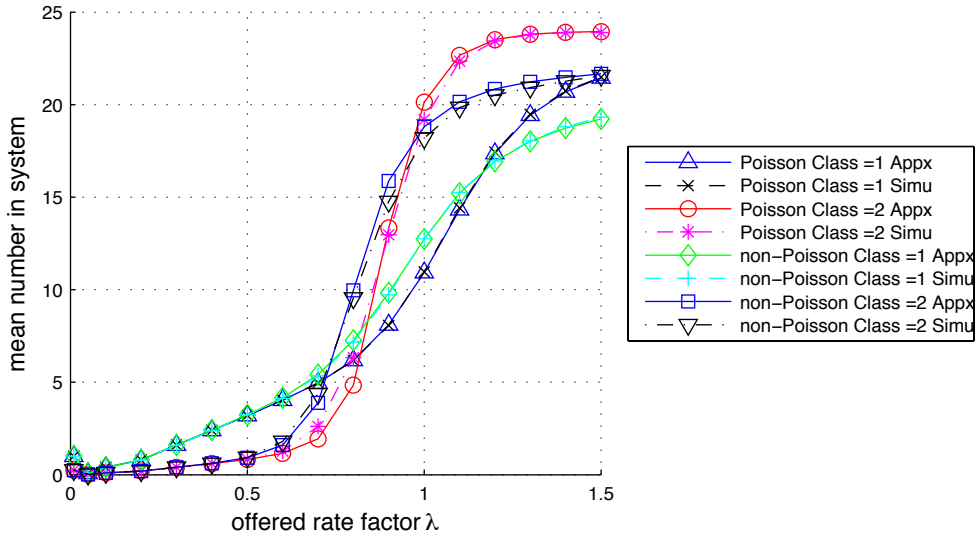

Figure 8. Mean number in system as a function of offered load factor.

We observe that our approximate results with non-Poisson arrivals closely match simulation results. Thus, the proposed approximation provides a tool to study the influence of the arrival process on the performance of such a preemptive priority system.

6. CONCLUSIONS

We have presented a simple approximate solution for preemptive-resume queues with multiple servers, general (phase-type) service and general (phase-type) interarrival time distributions. In our approach, priority levels are solved one at a time in the order of decreasing priorities. We use a reduced state description to deal with general service time distributions each priority level. Thus, the complexity of our approximate solution (in terms of the number of equations solved) grows linearly with the number of priority levels and the number of servers.

We studied many thousands of examples to assess the accuracy of our approximation comparing its results with those of discrete-event simulations. For Poisson and quasi-Poisson arrivals, we included systems with from 8 to 48 servers and a range of values for the mean service times, as well as a large range of values for the offered load at different priority levels. Overall, for these types of arrival processes, in our examples, the mean relative error for the mean number of customers in the system

was below 2% while the corresponding median relative error was below 0.25%. When examined for each priority level separately, as could be expected, relative errors tend to increase for lower priority levels. However, this increase in errors appears relatively slow. As an example, in the case of Poisson arrivals the mean error for the mean number in system grows from about 1.4% for level 2 to about 2.3% for level 4. Therefore, one can reasonably expect errors to remain acceptable even with a larger number of priority levels. The accuracy of our approximation tends to improve as the number of servers grows.

Based on several hundred examples, the good accuracy of our approximation appears to extend to the case of phase-type times between arrivals. As an example, with 16 servers and 4 priority levels, the mean error for the mean number in system grows from 1.4% to about 1.7% and about 3.2% as the coefficient of variation of the times between arrivals grows from 2 to 4 and 15, respectively. Even in the latter case of a Pareto-like distribution with 16 phases, the mean error remains below 5% for each of the 4 priority levels.

For simplicity of exposition, we used classical phase-type distributions for the times between arrivals and the service times. It is a straightforward matter to let the intensity and phase routing parameters of these distributions depend on the number of customers at the corresponding priority level. This extension may be useful in some applications. Additionally, the proposed approach can be readily applied to multi-server queues with preemptive-restart priority levels.

7. ACKNOWLEGEMENTS

This work was supported in part by the LABEX MILYON (ANR-10-LABX-0070) of Université de Lyon, within the program “Investissements d’Avenir” (ANR-11-IDEX-0007) operated by the French National Research Agency (ANR) and by PALLAS International Corporation of San Jose, California.

8. REFERENCES

[ALH15] A. Al Hanbali, E.M. Alvarez, and M.C. van der Heijden. "Approximations for the waiting-time distribution in an *M/PH/c* priority queue." *OR Spectrum* 37.2 (2015): 529-552.

[ALL90] A.O. Allen. *Probability, statistics, and queueing theory*. Second Edition. Elsevier, 1990.

[ATM16] T. Atmaca, T. Begin, A. Brandwajn, and H. Castel-Taleb. "Performance evaluation of cloud computing centers with general arrivals and service." *IEEE Transactions on Parallel and Distributed Systems* 27. 8 (2016): 2341-2348.

[BOB05] A. Bobbio, A. Horváth, and M.Telek. "Matching three moments with minimal acyclic phase type distributions." *Stochastic models* 21.2-3 (2005): 303-326.

[BOL05] G. Bolch, S. Greiner, H. de Meer, and K.S. Trivedi. *Queueing networks and Markov chains: modeling and performance evaluation with computer science applications*. John Wiley & Sons, 2006.

[BRA14] A. Brandwajn and T. Begin. "Reduced complexity in *M/Ph/c/N* queues." *Performance Evaluation* 78 (2014): 42-54.

[DAV66] R.H. Davis. "Waiting-time distribution of a multi-server, priority queuing system." *Operations Research* 14.1 (1966): 133-136.

[DEL13] R. De Lange, I. Samoilovich, and B. van der Rhee. "Virtual queuing at airport security lanes." European Journal of Operational Research 225.1 (2013): 153-165.

[ELL12] W. Ellens, J. Akkerboom, R. Litjens, and H. van den Berg. "Performance of cloud computing centers with multiple priority classes." In *Cloud Computing (CLOUD), 2012 IEEE 5th International Conference on*, pp. 245-252. IEEE, 2012.

[GAN03] N. Gans, G. Koole, and A. Mandelbaum. "Telephone call centers: a tutorial and literature review." *Manufacturing and Service Operations Management* 5.2 (2002): 79-141.

[GUP07] V. Gupta, J. Dai, M. Harchol-Balter, and B. Zwart. "The effect of higher moments of job size distribution on the performance of an *M/G/s* queueing system." *ACM SIGMETRICS Performance Evaluation Review* 35.2 (2007): 12-14.

[HAR05] M. Harchol-Balter, T. Osogami, A. Scheller-Wolf, and A. Wierman. Multi-server queueing systems with multiple priority classes. *Queueing Systems* 51.3-4 (2005): 331-360.

[KEL85] O. Kella and U. Yechiali. "Waiting times in the non-preemptive priority *M/M/c* queue." *Stochastic Models* 1.2 (1985): 257-262.

[LIN14] D. Lin, J. Patrick, and Fabrice Labeau. "Estimating the waiting time of multi-priority emergency patients with downstream blocking." *Health care management science* 17.1 (2014): 88-99.

[MAC89] M.H. MacDougall. *Simulating computer systems: techniques and tools*. MIT press, 1989.

[OSO06] T. Osogami and M. Harchol-Balter. "Closed form solutions for mapping general distributions to quasi-minimal PH distributions." *Performance Evaluation* 63.6 (2006): 524-552.

[STA04] W. Stallings. Operating Systems Internals and Design Principles. Fourth Edition. Prentice Hall, 2004.

[TAK91] H. Takagi and Y. Takahashi. "Priority queues with batch Poisson arrivals." *Operations Research Letters* 10.4 (1991): 225-232.

[WAN15] J. Wang, O. Baron, and A. Scheller-Wolf. "*M/M/c* Queue with Two Priority Classes." *Operations Research* 63.3 (2015): 733-749.

[ZEL04] S. Zeltyn, Z. Feldman, and S. Wasserkrug. "Waiting and sojourn times in a multi-server queue with mixed priorities." *Queueing Systems* 61.4 (2009): 305-328.