Stochastic Models, 21:725–744, 2005

ISSN: 1532-6349 print/1532-4214 online
DOI: 10.1081/STM-200056027

Taylor & Francis
Taylor & Francis Group

APPROXIMATED TRANSIENT QUEUE LENGTH AND WAITING TIME DISTRIBUTIONS VIA STEADY STATE ANALYSIS

B. Van Houdt and C. Blondia □ *University of Antwerp, Antwerpen, Belgium*

□ *We propose a method to approximate the transient performance measures of a discrete time queueing system via a steady state analysis. The main idea is to approximate the system state at time slot t or on the n-th arrival—depending on whether we are studying the transient queue length or waiting time distribution—by the system state after a negative binomially distributed number of slots or arrivals. By increasing the number of phases k of the negative binomial distribution, an accurate approximation of the transient distribution of interest can be obtained.*

In order to efficiently obtain the system state after a negative binomially distributed number of slots or arrivals, we introduce so-called reset Markov chains, by inserting reset events into the evolution of the queueing system under consideration. When computing the steady state vector of such a reset Markov chain, we exploit the block triangular block Toeplitz structure of the transition matrices involved and we directly obtain the approximation from its steady state vector. The concept of the reset Markov chains can be applied to a broad class of queueing systems and is demonstrated in full detail on a discrete-time queue with Markovian arrivals and phase-type services (i.e., the D-MAP/PH/1 queue). We focus on the queue length distribution at time t and the waiting time distribution of the n-th customer. Other distributions, e.g., the amount of work left behind by the n-th customer, that can be acquired in a similar way, are briefly touched upon.

Using various numerical examples, it is shown that the method provides good to excellent approximations at low computational costs—as opposed to a recursive algorithm or a numerical inversion of the Laplace transform or generating function involved—offering new perspectives to the transient analysis of practical queueing systems.

Keywords Markov chains; Matrix analytic method; Queueing systems; Queue length; Reset events; Steady state; Transient distribution; Waiting time.

Mathematics Subject Classification Primary 60K25; Secondary 60M20, 90B22.

Received September 2004; Accepted February 2005

B. Van Houdt is a postdoctoral fellow of the FWO Flanders.

Address correspondence to B. Van Houdt, University of Antwerp, Middleheimlaan 1, Antwerpen B-2020, Belgium; E-mail: benny.vanhoudt@ua.ac.be

1. INTRODUCTION

Transient performance measures have long been recognized as being complementary to the steady state characteristics of a queueing system, especially when the inter-arrival and service times are not exponential, either because there often exists a need to understand the initial behavior of a system, or simply because the system has no steady state. Obtaining transient information is generally considered more complicated in comparison to a steady state analysis. Roughly speaking, two main approaches have been developed to obtain transient distributions: the first relies on numerically inverting the Laplace transform or generating function involved (Choudhury[3]; Hofkens[4]; Lucantoni[9]), whereas the second is based on recursive computations. Others, such as Ny[15], combine uniformization techniques to reduce the problem to discrete time and afterward apply a recursive algorithm. Although these methods are effective in obtaining transient distributions related to the system behavior at the very beginning, their computational costs grow rapidly when considering events further along the time axis. As systems often take a considerable amount of time to reach their steady state, these methods can often no longer provide results within acceptable time frames. We propose a method that can achieve accurate approximations to the transient problem by making use of powerful steady state algorithms.

The key property of the proposed method is that we can approximate the interval $[0, t]$ or $[1, n]$ by a negative binomial distribution—which is the discrete time counterpart of the Erlang distribution—with k phases, for k sufficiently large. Meaning, the system state at time t (or on the n-th arrival) can be approximated by observing the system after a negative binomially distributed number of slots (or arrivals). The idea to approximate time t by an Erlang distribution is not new and was explored some time ago to obtain transient probabilities of finite state continues time Markov chains (Carmo[2]; Ross[16]). The choice of the negative binomial distribution to approximate time t (or the nth arrival) is in some sense optimal. Telek[17] has proven that the discrete time PH distribution with k phases, a mean $m_u > k$ and a minimal coefficient of variation is the negative binomial distribution with parameters $(k/m_u, k)$. Thus, the closest we can get to a deterministic distribution with a mean m_u, if we make use of a *discrete* PH distribution with at most k phases, is the negative binomial distribution with parameters $(p = k/m_u, k)$.

The method proposed in this paper computes the approximated transient performance measures via a single steady state analysis of a so-called *reset* Markov chain, which we obtain by introducing reset events in the evolution of the queueing system under consideration. The steady state vector of this reset Markov chain, which we compute by exploiting the structural properties of the transition block matrices, directly leads to

the approximation of interest. The concept of the reset Markov chains can be applied to a broad class of queueing systems, the main requirement being that the queue is governed by a homogeneous Markov chain whose transition matrix is sufficiently structured, e.g., a Quasi-Birth-Death (QBD), M/G/1 or GI/M/1 type (Neuts[13,14]). Instead of giving a general rather superficial discussion, we have chosen to present a detailed exposition of the method when applied on a discrete time queue with Markovian arrivals and phase-type services (the D-MAP/PH/1 queue). This choice was motivated by Hofkens[4], where a VBR playout buffer is dimensioned using the transient waiting time distribution of the nth customer in a D-MAP/PH/1 queue. It should however be clear that reset Markov chains can also be used for queueing systems containing batch arrivals, service vacations or multiple servers.

The paper is structured as follows: Section 2 describes the main characteristics of the D-MAP/PH/1 queue. Approximations to the queue length distribution at time t and the waiting time distribution of the nth customer, under the assumption that the queue is empty at time 0, are developed in sections 3 and 4. Within section 5 we indicate how to compute the steady state vector of the reset Markov chains discussed in sections 3 and 4. More general initial conditions, where the queue holds r customers at time 0, are dealt with in section 6. Other transient distributions such as the work left behind by the nth customer are briefly touched upon in section 7. Finally, a considerable amount of numerical examples demonstrating the accuracy and efficiency of the new method are presented in section 8.

2. THE D-MAP/PH/1 QUEUE

The D-MAP arrival process considered is the discrete time version of the Markovian arrival process (MAP) Lucantoni[8,10] and was first introduced in Blondia[1]. D-(B)MAPs form a class of tractable Markovian arrival processes, which, in general, are non-renewal and which include the discrete time variants of the Markov modulated Poisson process, the PH-renewal process and superpositions of such processes as particular cases. Formally, a D-MAP is defined by a set of two positive $l \times l$ matrices D_0 and D_1, with the property that

$$D = D_0 + D_1 \tag{1}$$

is a transition matrix. By definition, the Markov chain J_t associated with D and having $\{j | 1 \leq j \leq l\}$ as its state space, is controlling the actual arrival process as follows. Suppose J_t is in state j_1 at time t. By going to the next time instance $t+1$, there occurs a transition to another or possibly the same state, and an arrival may or may not occur. The entries $(D_1)_{j_1,j_2}$

represent the probability of having a transition from state j_1 to j_2 and a customer arrival. A transition from state j_1 to j_2 without an arrival will occur with probability $(D_0)_{j_1,j_2}$. For D primitive, the Markov chain J_t has a unique stationary distribution. Let θ be the stationary probability vector of the Markov chain J_t, i.e., $\theta D = \theta$ and $\theta e_l = 1$, with e_i an $i \times 1$ column vector of ones. The mean arrival rate λ of the D-MAP is given by $\lambda = \theta D_1 e_l$.

The service times are assumed to follow a discrete time phase-type (PH) distribution with matrix representation (m, β, T), where m is a scalar, β a $1 \times m$ stochastic vector and T a $m \times m$ substochastic matrix (Neuts[14]). The sth component of the vector β represents the probability that a customer starts his service in phase s. Let $T^* = e_m - Te_m$, then the sth entry of T^* denotes the probability that a customer completes his service provided that he is in phase s at the current time epoch. Finally, the (s_1, s_2)th entry of T equals the probability that a customer continues his service in phase s_2 at the next time epoch provided that he is in phase s_1 at the current time epoch. The mean service time is given by $E[S] = \beta(I_m - T)^{-1}e_m$, where I_i is a unity matrix of dimension i. The set of discrete time PH distributions is known to be very useful in approximating service time distributions encountered in communications networks (Lang[5]).

A single work conserving server is considered. All events, such as new arrivals and service completions, are assumed to occur just prior to the discrete time epochs. It is well known that the D-MAP/PH/1 queue forms a QBD Markov chain with transition matrix $\overline{P}$ (Blondia[1]):

$$\overline{P} = \begin{bmatrix} \overline{B}_1 & \overline{B}_0 & 0 & 0 & \cdots \\ \overline{B}_2 & \overline{A}_1 & \overline{A}_0 & 0 & \ddots \\ 0 & \overline{A}_2 & \overline{A}_1 & \overline{A}_0 & \ddots \\ \vdots & \ddots & \ddots & \ddots & \ddots \end{bmatrix}, \tag{2}$$

where $\overline{B}_0 = D_1 \otimes \beta$, $\overline{B}_1 = D_0$, $\overline{B}_2 = D_0 \otimes T^*$, $\overline{A}_0 = D_1 \otimes T$, $\overline{A}_1 = (D_0 \otimes T) + (D_1 \otimes T^*\beta)$ and $\overline{A}_2 = D_0 \otimes T^*\beta$, with $\otimes$ the matrix Kronecker product. Notice, depending on whether $\rho = \lambda E[S] < 1$, the Markov chain $\overline{P}$ is stationary or not.

3. THE QUEUE LENGTH DISTRIBUTION OF A D-MAP/PH/1 QUEUE AT TIME t

In this section, we focus on approximating the state of the Markov chain $\overline{P}$ at time t, given that the queue is empty at time zero and the initial phase of the D-MAP is distributed according to some stochastic vector α (meaning that its ith entry α_i equals the probability that state i is the initial

D-MAP state). Denote $X(t) = (\alpha, 0, 0, \dots)(\overline{P})^t$ as the probability vector of the system at time t. Clearly, unless t is small, computing $X(t)$ recursively by means of $X(t) = X(t-1)\overline{P}$ is a time consuming process, even when exploiting the structure of $\overline{P}$.

We propose a method that allows us to approximate $X(t)$ directly, by considering the system state $X_k(t)$ after a negative binomially distributed number of slots $Z_{k,t}$, which we compute by means of a steady state analysis of a reset Markov chain. The random variable $Z_{k,t}$ is chosen as having a negative binomial distribution with parameters $(p = k/(t+1), k)$, meaning $Z_{k,t}$ is the sum of k independent geometric[a] random variables with parameter $p = k/(t+1)$. Using some of the basic properties of the negative binomial distribution we have: $E[Z_{k,t}] = t + 1$ and $Var[Z_{k,t}] = k(1-p)/p^2 = (t+1)^2/k - (t+1)$. Thus, as k, for $1 \le k \le t+1$, increases, the variance of $Z_{k,t}$ decreases to zero and $Z_{k,t}$ becomes deterministic. Assuming that $Z_{k,t}$ can be regarded as close to deterministic, the system state at time $Z_{k,t}$ should provide us with a good approximation to the system state at time t.

Let us now explain how to compute the system state at time $Z_{k,t}$ via a steady state analysis. Consider the stochastic process that evolves according to the transition matrix $\overline{P}$, but that is repeatedly reset after a time $Z_{k,t}$. Meaning, if we perform a Bernoulli trial at each time epoch, with parameter $p = k/(t+1)$, the system will be reset whenever k successes have occurred. The reset counter is then defined as the number of pending successes before the next reset event. Clearly the reset counter takes values in the range $\{1, 2, \dots, k\}$. Although adding the reset counter variable as an additional auxiliary variable to the Markov chain $\overline{P}$ increases the size of its transition blocks by a factor k, we will demonstrate that by exploiting their structural properties, we can drastically reduce the time and memory complexity needed to compute its steady state vector (see section 5). After adding the reset counter as an additional auxiliary variable to the Markov chain $\overline{P}$, the reset process becomes a QBD characterized by the transition matrix $P_{k,t}$:

$$P_{k,t} = \begin{bmatrix} B_1^{k,t} + C_0^{k,t} & B_0^{k,t} & 0 & 0 & \dots \\ B_2^{k,t} + C_1^{k,t} & A_1^{k,t} & A_0^{k,t} & 0 & \ddots \\ C_1^{k,t} & A_2^{k,t} & A_1^{k,t} & A_0^{k,t} & \ddots \\ C_1^{k,t} & 0 & A_2^{k,t} & A_1^{k,t} & \ddots \\ \vdots & \vdots & \ddots & \ddots & \ddots \end{bmatrix}, \tag{3}$$

[a]Notice, we use the following definition for a geometric random variable: X is geometric if $P[X = i] = (1-p)^{i-1}p$, for $i > 0$.

where

$$A_i^{k,t} = H_0^k(p) \otimes \overline{A}_i, \tag{4}$$

$$B_i^{k,t} = H_0^k(p) \otimes \overline{B}_i, \tag{5}$$

$$C_0^{k,t} = H_1^k(p) \otimes (e_l \alpha), \tag{6}$$

$$C_1^{k,t} = H_1^k(p) \otimes (e_{ml} \alpha) = C_0^{k,t} \otimes e_m, \tag{7}$$

for $i = 0, 1$ or 2, and $p = k/(t+1)$. The $k \times k$ matrices $H_i^k(p)$ given below, describe the evolution of the reset counter:

$$H_0^k(p) = \begin{bmatrix} (1-p) & 0 & \dots & 0 & 0 \\ p & (1-p) & \ddots & 0 & 0 \\ 0 & p & \ddots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & p & (1-p) \end{bmatrix},$$

$$H_1^k(p) = \begin{bmatrix} 0 & 0 & \dots & 0 & p \\ 0 & 0 & \ddots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & 0 \end{bmatrix}. \tag{8}$$

Notice, the matrix $H_0^k(p)$ covers all the transitions that do not bring about a reset event, either because the reset counter was larger than one (in which case it decreases by one with probability p) or because the reset counter did equal one, but the Bernoulli trial failed. $H_1^k(p)$ covers the transitions associated with a reset event; such an event can only occur if the reset counter was equal to one and the trial was successful.

Even though the D-MAP/PH/1 queue of interest might be unstable, $P_{k,t}$ is always stationary (for $t+1 > k > 0$) due to the reset feature (see Appendix A). An efficient algorithm that computes the steady state vector $\pi_{k,t}$, defined as $\pi_{k,t} P_{k,t} = \pi_{k,t}$ and $\pi_{k,t} e = 1$ (where e is an infinite column vector with all its entries equal to one), for a matrix of the form $P_{k,t}$ is presented in section 5. This algorithm takes advantage of the block triangular block Toeplitz form of the matrices $A_0^{k,t}$, $A_1^{k,t}$ and $A_2^{k,t}$. In order to obtain the system state $X_k(t)$ of the original Markov chain $\overline{P}$ at time $Z_{k,t}$, it suffices to consider the system state of the Markov chain $P_{k,t}$ when a reset event occurs. Denote $\pi_{k,t} = (\pi_0^{k,t}, \pi_1^{k,t}, \dots,)$, where $\pi_0^{k,t}$ is a $1 \times kl$ vector and $\pi_i^{k,t}$, for $i > 0$, a $1 \times kml$ vector. Moreover, let $\pi_i^{k,t} = (\pi_{i,1}^{k,t}, \dots, \pi_{i,k}^{k,t})$, with $\pi_{i,j}^{k,t}$, for $j = 1, \dots, k$, a $1 \times ml$ $(1 \times l)$ vector for $i > 0$ $(i = 0)$. The probability of

being in some state s when a reset event takes place equals the expected number of reset events that occur at an arbitrary time slot from state s, divided by the rate $1/(t+1)$ at which reset events occur. This yields

$$X_k(t) = p(t+1)(\pi_{0,1}^{k,t}, \pi_{1,1}^{k,t}, \pi_{2,1}^{k,t}, \dots) = k(\pi_{0,1}^{k,t}, \pi_{1,1}^{k,t}, \pi_{2,1}^{k,t}, \dots). \tag{9}$$

Indeed, a reset event can only occur (with probability p) when the reset counter equals 1. Having found an approximation $X_k(t)$ for $X(t)$, we easily find an approximation $Q_k(t)$ for $Q(t)$, the queue length distribution at time t, by summing the appropriate probabilities.

Remark. Instead of approximating the system state at time t by the state at time $Z_{k,t}$, we can also develop a slightly different approximation as follows. We start by observing that the steady state vector $\pi_{k,t}$ of $P_{k,t}$ obeys the following equation:

$$\Xi_k(\pi_{k,t}) = \frac{1}{t+1} \sum_{i=0}^{\infty} P[Z_{k,t} > i] X(i), \tag{10}$$

where $\Xi_k(x) = \sum_{i=1}^{k} \left(\pi_{0,i}^{k,t}, \pi_{1,i}^{k,t}, \pi_{2,i}^{k,t}, \dots\right)$. Indeed, if we observe the Markov chain characterized by $P_{k,t}$ at an arbitrary time instant, it is easy to show that $jP[Z_{k,t} = j]/(t+1)$ equals the probability that the length of the reset interval in which our observed slot lies, equals j. Thus, $P[Z_{k,t} = j]/(t+1)$ equals the probability that the observed slot is the $(i+1)$th of those j slots, for $i+1 \le j$. Hence, $\sum_{j>i} P[Z_{k,t} = j]/(t+1) = P[Z_{k,t} > i]/(t+1)$ equals the probability that we observe the system i time units after a reset event. The probability vector corresponding to such a slot is $X(i)$. As k increases to $t+1$, these probabilities approach 1 for $i \le t$ and 0 for $i > t$. Meaning that as k increases, $\Xi_k(\pi_{k,t})$ approaches $1/(t+1)\sum_{i=0}^{t} X(i)$. Defining $X'_k(t) = (t+1)\Xi_k(\pi_{k,t}) - t\Xi_k(\pi_{k,t-1})$ thus provides us with an alternative approximation for $X(t)$. When comparing the approximations $Q_k(t)$ and $Q'_k(t)$, obtained from $X_k(t)$ and $X'_k(t)$, we found that the tail probabilities of $X_k(t)$ where somewhat more accurate (especially for k small), whereas $X'_k(t)$ provides us with a closer match for the initial probabilities of the queue length distribution.

4. THE WAITING TIME DISTRIBUTION OF THE *n*th CUSTOMER IN A D-MAP/PH/1 QUEUE

We can apply a similar idea as in the previous section to obtain an approximation $W_k(n)$ for the waiting time distribution $W(n)$ of the nth customer. The key is to reset the Markov chain $\overline{P}$ not at *time* $Z_{k,t}$, but at the $Z_{k,n}$*th arrival*, where $Z_{k,n}$ is a random variable having a negative binomial

distribution with parameters $(p = k/n, k)$. Let us have a closer look at the reset Markov chain used to obtain the approximation $W_k(t)$. Resetting the system at the $Z_{k,n}$th arrival means that we perform a Bernoulli trial at each *arrival* epoch and when k successes have occurred, we reset the system. Thus, the reset counter will remain the same unless there is an arrival with an associated Bernoulli trial that is successful, in which case the counter decreases by one (unless it equaled one and is subsequently set to k). Denoting the transition matrix of this Markov chain as $\widetilde{P}_{k,n}$, we have

$$\widetilde{P}_{k,n} = \begin{bmatrix} \widetilde{B}_1^{k,n} + \widetilde{C}_0^{k,n} & \widetilde{B}_0^{k,n} & 0 & 0 & \dots \\ \widetilde{B}_2^{k,n} + \widetilde{C}_1^{k,n} & \widetilde{A}_1^{k,n} & \widetilde{A}_0^{k,n} & 0 & \ddots \\ \widetilde{C}_1^{k,n} & \widetilde{A}_2^{k,n} & \widetilde{A}_1^{k,n} & \widetilde{A}_0^{k,n} & \ddots \\ \widetilde{C}_1^{k,n} & 0 & \widetilde{A}_2^{k,n} & \widetilde{A}_1^{k,n} & \ddots \\ \vdots & \vdots & \ddots & \ddots & \ddots \end{bmatrix}, \tag{11}$$

where

$$\widetilde{A}_0^{k,n} = H_0^k(p) \otimes \overline{A}_0, \tag{12}$$

$$\widetilde{A}_1^{k,n} = (H_0^k(p) \otimes (D_1 \otimes T^*\beta)) + (I_k \otimes (D_0 \otimes T)), \tag{13}$$

$$\widetilde{A}_2^{k,n} = I_k \otimes \overline{A}_2, \tag{14}$$

$$\widetilde{B}_0^{k,n} = H_0^k(p) \otimes \overline{B}_0, \tag{15}$$

$$\widetilde{B}_i^{k,n} = I_k \otimes \overline{B}_i, \tag{16}$$

$$\widetilde{C}_0^{k,n} = H_1^k(p) \otimes (D_1 e_l \alpha) \tag{17}$$

$$\widetilde{C}_1^{k,n} = H_1^k(p) \otimes (D_1 e_l \alpha \otimes e_m) = \widetilde{C}_0^{k,n} \otimes e_m, \tag{18}$$

with $i = 1, 2$ and $p = k/n$. Notice, the matrices $\overline{A}_0$ and $\overline{B}_0$ correspond to an arrival, whereas the matrices $\overline{A}_2, \overline{B}_1$ and $\overline{B}_2$ do not.

Denote $Y_{k,n}$ as the time at which the $Z_{k,n}$th arrival occurs in the original Markov chain $\overline{P}$ and let the vector $X(Y_{k,n})$ reflect the system state of the Markov chain $\overline{P}$ at time $Y_{k,n}$. In order to obtain $X(Y_{k,n})$ it suffices to consider the system state of the Markov chain $\widetilde{P}_{k,n}$ when a reset event occurs. Denote $\tilde{\pi}_{k,n} = (\tilde{\pi}_0^{k,n}, \tilde{\pi}_1^{k,n}, \dots,)$, where $\tilde{\pi}_0^{k,n}$ is a $1 \times kl$ vector and $\tilde{\pi}_i^{k,n}$, for $i > 0$, a $1 \times kml$ vector. Moreover, let $\tilde{\pi}_i^{k,n} = (\tilde{\pi}_{i,1}^{k,n}, \dots, \tilde{\pi}_{i,k}^{k,n})$, with $\tilde{\pi}_{i,j}^{k,n}$, for $j = 1, \dots, k$, a $1 \times ml$ $(1 \times l)$ vector for $i > 0$ $(i = 0)$. Then, in view of the argument presented to obtain Eqn. (9), it follows

$$X(Y_{k,n}) = E[\text{reset}]p(\tilde{\pi}_{0,1}^{k,n} \cdot (D_1 e_l)^T, \tilde{\pi}_{1,1}^{k,n} \cdot (D_1 e_l \otimes e_m)^T, \tilde{\pi}_{2,1}^{k,n} \cdot (D_1 e_l \otimes e_m)^T, \dots), \tag{19}$$

where E[reset] is the expected reset time of the Markov chain $\widetilde{P}_{k,n}$ (which might differ from the expected arrival time of the nth customer) and the vector product appearing in Eqn. (19) is the point-wise product. An algorithm to compute E[reset] is presented in Appendix B. Indeed, a reset event can only occur (with probability p) when the reset counter equals 1 and when an arrival occurs. Having found an approximation $X(Y_{k,n})$ for the system state at the nth arrival, we can define the following approximation $W_k(n)$ to $W(n)$, the waiting time distribution of the nth customer[b]:

$$P[W_k(n) = 0] = \sum_j X(Y_{k,n})_{\langle 0,j\rangle} + \sum_{s,j} X(Y_{k,n})_{\langle 1,j,s\rangle}(T^*)_s, \tag{20}$$

$$P[W_k(n) = w] = \sum_{q\geq 1}\sum_{s,j} X(Y_{k,n})_{\langle q,j,s\rangle} P[S^{(q-1)*} + R(s) = w], \tag{21}$$

where $S^{(q-1)*}$ denotes the $(q-1)$-fold convolution of the service time distribution S, $X(Y_{k,n})_{st}$ the entry of the vector $X(Y_{k,n})$ that corresponds to state st and $R(s)$ denotes the residual service time provided that the current phase of service is s, i.e., $P[R(s) = r] = (T^r T^*)_s$.

5. COMPUTING THE STATIONARY VECTORS $\pi_{k,t}$ AND $\tilde{\pi}_{k,n}$

Both the transition matrices $P_{k,t}$ and $\widetilde{P}_{k,n}$ have the following form:

$$P = \begin{bmatrix} B_1 + C_0 & B_0 & 0 & 0 & \cdots \\ B_2 + C_1 & A_1 & A_0 & 0 & \ddots \\ C_1 & A_2 & A_1 & A_0 & \ddots \\ C_1 & 0 & A_2 & A_1 & \ddots \\ \vdots & \vdots & \ddots & \ddots & \ddots \end{bmatrix}. \tag{22}$$

A Markov chain of this form is called a Quasi-Birth-Death (QBD) Markov chain with a generalized initial condition Neuts[13]. The key in finding the steady state probability vector $\pi = (\pi_0, \pi_1, \dots)$ of P, where π_0 and π_i, for $i > 0$, have the same dimension as B_1 and A_1, respectively, is to solve the following equation:

$$G = A_2 + A_1 G + A_0 G^2. \tag{23}$$

[b]The notations $\langle 0, j\rangle$ and $\langle q, s, j\rangle$ are used to reflect the states associated with an empty queue and a queue holding q customers (the first being in phase s in the service facility), respectively, while the current state of the arrival process is j.

The matrices A_0, A_1 and A_2 of both the $P_{k,t}$ and the $\widetilde{P}_{k,n}$ Markov chains are block triangular block Toeplitz (btbT) matrices. A btbT matrix X is characterized by its first block column as follows:

$$X = \begin{bmatrix} X_1 & 0 & \dots & 0 & 0 \\ X_2 & X_1 & \ddots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ X_{k-1} & X_{k-2} & \ddots & X_1 & 0 \\ X_k & X_{k-1} & \dots & X_2 & X_1 \end{bmatrix}. \tag{24}$$

One of the advantages of working with btbT matrices is that it suffices to store the first block column. Moreover, it is easily seen that the product between two btbT matrices is again btbT. Therefore, the G matrices $G_{k,t}$ and $\widetilde{G}_{k,n}$, corresponding to $P_{k,t}$ and $\widetilde{P}_{k,n}$, are also btbT matrices.

We propose to use the Cyclic Reduction (CR) algorithm to compute G (Meini[12]). This algorithm is very easy to implement, requires a low amount of memory, converges quadratically and is numerically stable. Although $A = A_0 + A_1 + A_2$ is not stochastic in our case (as $C_1 > 0$), the convergence of the CR algorithm is still guaranteed by the stationarity of the Markov chains concerned. Additionally, each of the intermediate matrices used by the CR algorithm is a btbT matrix (because the inverse of a btbT is also btbT). Hence, we can easily reduce the time and memory complexity of a single iteration from $O(k^3m^3l^3)$ and $O(k^2m^2l^2)$ to $O(k^2m^3l^3)$ and $O(km^2l^2)$, respectively. The time complexity can even be further reduced to $O(km^3l^3 + k\log(k)m^2l^2)$ by making use of fast Fourier transforms (see Meini[11], Chapter 2 for details).

Having found G, one computes the btbT matrices R and R_1 as $A_0(I - A_1 - A_0G)^{-1}$ and $B_0(I - A_1 - A_0G)^{-1}$ (notice, the blocks of the btbT matrix R_1 are not square) (Latouche[7]). The steady state probability vectors π_i are then found as:

$$\pi_0 = \pi_0(B_1 + C_0 + R_1(B_2 + (I - R)^{-1}C_1)), \tag{25}$$

$$\pi_1 = \pi_0 R_1, \tag{26}$$

$$\pi_i = \pi_{i-1}R, \tag{27}$$

for $i > 1$, while π_0 and π_1 are normalized as $\pi_0 e + \pi_1(I - R)^{-1}e = 1$.

6. GENERAL INITIAL CONDITIONS

In the previous sections we considered the transient behavior of the D-MAP/PH/1 queue where the system was empty at time 0 and the initial D-MAP state was determined by a vector α. In this section we indicate how

to deal with a more general initial system state. We assume that r customers are present in the queue at time 0 and let β_1 determine the phase of the customer in service at time 0, that is, $(\beta_1)_i$ represents the probability that its service is in phase i at time 0. Clearly, we can apply the same principles as in the previous two sections to obtain an approximation for the queue length distribution at time t, as well as the waiting time distribution of the nth customer. The only difficulty lies in the fact that the transition matrices involved are no longer of the form given in section 5, because the reset events no longer result in a transition to an empty queue state.

Given the more general initial condition mentioned above, it can be readily seen that both transition matrices $P_{k,t}$ and $\widetilde{P}_{k,n}$ involved have the following form:

$$P = \begin{bmatrix} B_1 & B_0 & 0 & \dots & 0 & C_0 & 0 & \dots \\ B_2 & A_1 & A_0 & \dots & 0 & C_1 & 0 & \dots \\ 0 & A_2 & A_1 & \ddots & 0 & C_1 & 0 & \dots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \ddots \\ 0 & 0 & 0 & \ddots & A_1 & A_0 + C_1 & 0 & \dots \\ 0 & 0 & 0 & \ddots & A_2 & A_1 + C_1 & A_0 & \ddots \\ 0 & 0 & 0 & \ddots & 0 & A_2 + C_1 & A_1 & \ddots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots & \vdots & \ddots \end{bmatrix}, \tag{28}$$

where the matrices C_0 and C_1 appear on the $(r+1)$th block column of the transition matrix P. As in the previous sections, these matrices correspond to a reset event.

The C_0 and C_1 matrices appearing in both Markov chains of interest have the following form: $C_0 = \gamma_0\gamma$ and $C_1 = \gamma_1\gamma$ for some column vectors γ_i, for $i = 0, 1$, and for some row vector γ. We shall use the superscripts k, t and k, n to identify the system of interest. The γ_0 and γ_1 vectors hold the probabilities that a reset event takes place given that the server is idle and busy, respectively. The current phase of the customer in service does not influence these probabilities; hence, $\gamma_1 = \gamma_0 \otimes e_m$. Given that we are in some state i of the Markov chain, the reset probabilities are not affected by the initial state. Therefore, looking at the two previous sections, we find $\gamma_0^{k,t} = (H_1^k(k/(t+1))e_k) \otimes e_l$ and $\gamma_0^{k,n} = (H_1^k(k/n)e_k) \otimes (D_1 e_l)$. The vector γ determines the new initial state after resetting the system: $\gamma = (0, (\alpha \otimes \beta_1))$, where 0 denotes a zero vector of the appropriate dimension (that is, of dimension $ml(k-1)$).

Although we can regard a transition matrix of the form given in Eqn. (28) as a QBD with a generalized boundary condition, a more efficient algorithm is presented in Appendix C. This algorithm is based on

a QBD reduction technique to compute the steady state vector and exploits the structure of the matrices C_0 and C_1. We shall refer to a Markov chain of this particular form as an r-reset QBD, as P is a QBD that also allows an immediate reset event to level r.

Remark. The more general case where the initial number of customers follows a bounded distribution N_{ini}, i.e., there exists some n_{max} such that $P[N_{ini} \leq n_{max}] = 1$, can be treated in a similar way. However, in this case one needs to add two artificial states to each level ($i < n_{max}$) to construct the QBD (one for the upward and one for the downward direction). Two artificial states suffice as drawing a random number r_{ini} from the distribution N_{ini}, can be split in several steps: (i) we first draw a random number to decide whether r_{ini} is more, less or equal to i (where i is the level in which the reset event occurs), (ii) given that r_{ini} is more (less) than some j we can draw a random number to decide whether it is equal to $j+1$ or more ($j-1$ or less) and repeat this step until r_{ini} is determined.

7. OTHER TRANSIENT PERFORMANCE DISTRIBUTIONS

The introduction of reset events is not only effective in acquiring the distribution of the queue length at time t or the waiting time of the nth customer, but can also facilitate the computation of other transient performance distributions.

For instance, assume we want to calculate the distribution of the amount of work left behind by the nth customer. In order to find this distribution, it suffices to obtain the queue length distribution at the nth service completion. We can approximate this distribution by means of a technique analogue to section 4, but instead of resetting the Markov chain after $Z_{k,n}$ arrivals, the reset event takes place at the $Z_{k,n}$th service completion. We can incorporate such events in the Markov chain $\overline{P}$, by performing a Bernoulli trail (with success probability $p = k/n$) whenever a customer leaves the service facility and by resetting the chain at the kth success.

Other examples include the queue length distribution when a specific state j of the arrival process (or a certain phase s of the service process) is entered/visited for the nth time. When considering a D-MAP/PH/1 queue with service vacations, reset events can be used to approximate the queue length distribution at the beginning/end of the nth service vacation and so on.

8. NUMERICAL EXAMPLES

A fairly arbitrary D-MAP/PH/1 queue was chosen to perform these experiments. Many other cases not presented in this section provided

similar results. We consider a 2-state D-MAP that generates an arrival with probability $d_1 = 0.1$ when in state 1 and with probability $d_2 = 0.25$ or 0.5 while in state 2. The average sojourn time in both states is 500 and 1000 slots, respectively. Hence,

$$D_0 = \begin{bmatrix} 0.998(1-d_1) & 0.002(1-d_1) \\ 0.001(1-d_2) & 0.999(1-d_2) \end{bmatrix}, \quad D_1 = \begin{bmatrix} 0.998d_1 & 0.002d_1 \\ 0.001d_2 & 0.999d_2 \end{bmatrix}. \tag{29}$$

For $d_2 = 0.25$, the arrival rate λ of the D-MAP equals 0.2, whereas for $d_2 = 0.5$ we have $\lambda = 0.3666$. The service times follow a 3-phase PH distribution characterized by $(3, \beta, T)$:

$$T = \begin{bmatrix} 4/5 & 0 & 0 \\ 0 & 1/2 & 0 \\ 0 & 0 & 1/4 \end{bmatrix}, \quad \beta = (3/4, 1/8, 1/8). \tag{30}$$

The mean service time $E[S] = \beta(I - T)^{-1}e_3 = 4.1666$, resulting in a system load $\rho = 0.8333$ and 1.5277 for $d_2 = 0.25$ and 0.5, respectively. Notice, for $d_2 = 0.5$ the system is severely overloaded causing a continuous growth of the mean queue length.

8.1. Queue Length Distribution *Q(t)*

Figure 1 presents the exact queue length distribution $Q(t)$, for $t =$ 50, 500 and 5000, as well as the approximations $Q_k(t)$ for various values of k. The exact results were obtained by a brute-force method whose computation time grows as a function of t, whereas the computational resources needed for the approximation method are almost insensitive to t (for a fixed value of k). The computation time for $k = 150$ takes less than 10 seconds[c] on a PC with a 2Ghz Intel Pentium CPU and 512 MB RAM without relying on fast Fourier transforms. The initial state of the D-MAP was state 2 (i.e., $\alpha = (0, 1)$). Results not presented here have shown that the accuracy of the results was not influenced by the choice of the initial D-MAP state. A number of conclusions can be drawn from Figure 1. In general, it is fair to say that the approximation method provides good-to-excellent results for fairly limited values of the system parameter k. Secondly, as t grows, the accuracy of $Q_k(t)$ tends to decrease (while keeping k fixed). This is logical because $Var[Z_{k,t}] = (t+1)^2/k - (t+1)$ grows as a function of t for k fixed. The exception to this general rule is Figure 1(e), were we obtain better results in comparison with $t = 500$. This is

[c]Except for Figure 1(f), due to the long queue lengths. Computing R and solving the boundary problem takes less than 10 seconds even for $k = 200$.

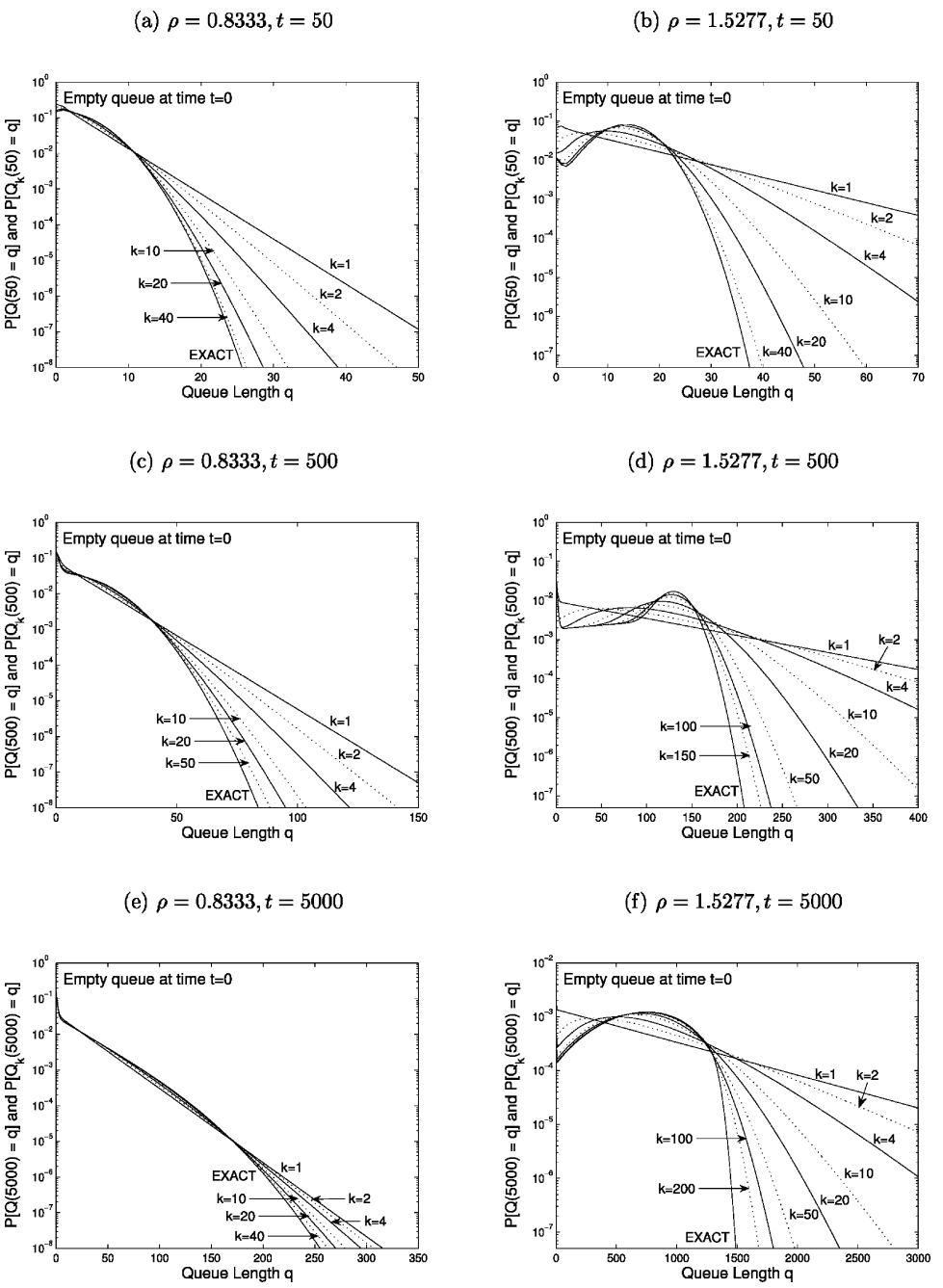

FIGURE 1 Queue length distribution $Q(t)$ for $\rho = 0.8333$: a) $t = 50$, c) $t = 500$ and e) $t = 5000$; and $\rho = 1.5277$: b) $t = 50$, d) $t = 500$, f) $t = 5000$.

due to the fact that the queueing system is stable, i.e., $\rho < 1$, therefore, it has a steady state distribution Q. Given the mean sojourn times of 500 and 1000 slots for the D-MAP arrival process, it can be expected that $Q(t)$ starts to approach the steady state distribution Q for t larger than a few thousand. Furthermore, it should be intuitively clear that the error made by $Q_k(t)$ depends to a great extent on the magnitude of the changes that $Q(t)$ undergoes while increasing (or decreasing) t. If t is large and the system has a steady state, we may expect these changes to be minor and therefore, $Q_k(t)$ more easily approximates $Q(t)$ for k small. This intuition is also confirmed by Figures 1(b,d,f) were the changes are significant for all t as the mean queue length grows continuously as a function of t. Indeed, even for $k = 50$ the error on the 10^{-7}-quantile is considerable.

8.2. Waiting Time Distribution W(n)

In this section we present some results for the waiting time distribution of the nth customer. We consider the same D-MAP/PH/1 queue as in the previous section and set $d_1 = 0.25$. As before, the initial D-MAP state vector is chosen as $\alpha = (0, 1)$. Figure 2 presents a comparison between the approximations $W_k(n)$ and the exact distribution $W(n)$ for $n =$ 150 and 400. Exact results were obtained by numerically inverting the two-dimensional generating function of the queueing delay of the nth arrival (Hofkens[4], Theorem 1) by means of a MATLAB implementation of the Fourier-series method presented in Choudhury[3]. Unfortunately computing exact results in this manner is very time consuming, especially for n large. Therefore, we had to limit ourselves to $n \leq 400$ (which already took many hours to compute). The computation time of the approximation method is nearly insensitive to the value of n and

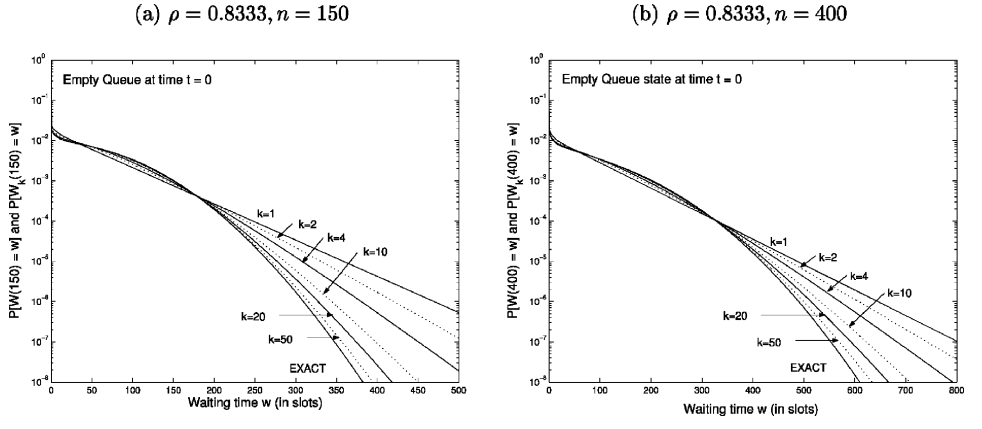

FIGURE 2 Waiting time distribution $W(n)$ for $\rho = 0.8333$: a) $n = 150$, b) $n = 400$.

computing $X(Y_{k,n})$ requires about 1 second for $k = 50$, obtaining the waiting time distribution from $X(Y_{k,n})$ takes a few seconds. In conclusion, using the approximation method we can obtain (fairly) accurate results at substantially lower computational costs, bringing the computation of transient waiting time distributions more within practical reach.

8.3. General Initial Conditions

Finally, some results on the same D-MAP/PH/1 queue (with $d_1 = 0.25$) and an initial number of $r = 50$ customers in the queue at time 0, are depicted in Figure 3. The first of these 50 customers will start his service at time 0, meaning $\beta_1 = \beta$ in this particular case. As before, we find a good agreement between the exact results, obtained by a brute-force computation and our approximation method. Both queue length distributions (at time $t = 50$ and 500) are strongly affected by the initial 50 customers present in the system. As t is further increased the local maximum around $q = 50$ will eventually disappear (as the system has a steady state for $\rho = 0.8333$).

APPENDIX A: STATIONARITY OF $P_{k,t}$ AND $\widetilde{P}_{k,n}$

The Markov chains characterized by the transition matrices $P_{k,t}$ and $\widetilde{P}_{k,n}$ are stationary, for $1 \leq k \leq t$ and $1 \leq k < n$.

Proof. We start with the Markov chain $P_{k,t}$. Denote I as the set of empty queue states $\langle 0, k, i \rangle$ of $P_{k,t}$ for which $\alpha_i > 0$, where $\langle 0, c, i \rangle$ reflects an empty queue with the D-MAP state being i and the reset counter equal to c. The states in I are obviously recurrent as their expected return time is

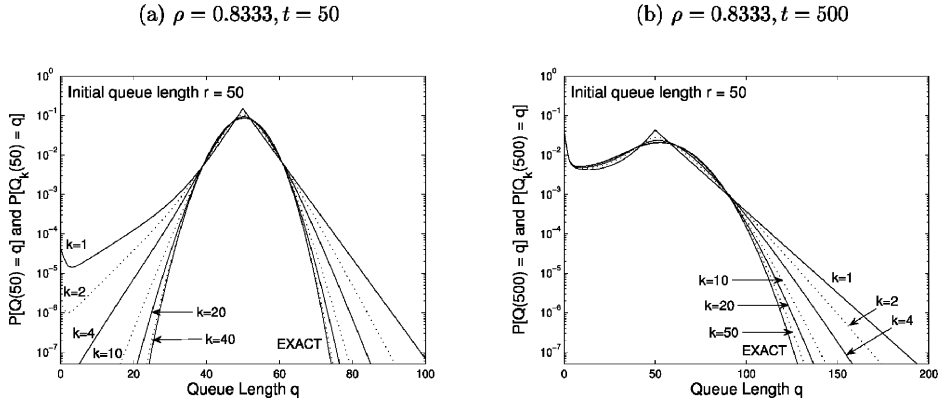

FIGURE 3 Queue length distribution $Q(t)$ with $r = 50$ initial customers for $\rho = 0.8333$: a) $t = 50$, b) $t = 500$.

upper bounded by $(t+1)/\alpha_i$. All other states can be subdivided into two subsets: (i) those states j that are unreachable from any of the states in I and (ii) the states j that can be reached with some probability $p_{ij} > 0$ from one of the states i in I (before a reset event occurs). The first set of states is clearly transient as the probability that $Z_{k,t}$ is finite equals one. Notice, depending on the initial state of the Markov chain, these states are either visited a finite number of times or not at all. The second set of states forms an irreducible class of states that are all recurrent, because the expected return time of any state j belonging to the second subset is upper bounded by $(t+1)/(\alpha_i p_{ij})$. The arguments for $\widetilde{P}_{k,n}$ are analogous and make use of the fact that $E[\text{reset}]$ is finite for all n and $k \leq n$ whenever the arrival rate of the D-MAP $\lambda > 0$.

The presence of possible transient states does not cause any problems as their corresponding entries in the stationary probability vector equal zero.

APPENDIX B: COMPUTING THE EXPECTED VALUE *E*[*reset*]

The expected value of the nth arrival, denoted as $E[Y_n]$, can obviously be read as

$$E[Y_n] = E[Y_n - Y_{n-1}] + E[Y_{n-1} - Y_{n-2}] + \cdots + E[Y_2 - Y_1] + E[Y_1 - Y_0], \tag{31}$$

where $Y_0 = 0$. The probability vector of the D-MAP state after i arrivals equals $\alpha((I - D_0)^{-1}D_1)^i$. Meaning that $E[Y_i - Y_{i-1}]$, for $i = 1, \ldots n$, matches

$$E[Y_i - Y_{i-1}] = \alpha((I - D_0)^{-1}D_1)^{i-1} \sum_{j=1}^{\infty} j(D_0)^{j-1} D_1 e_l, \tag{32}$$

$$= \alpha((I - D_0)^{-1}D_1)^{i-1}((I - D_0)^{-2}D_1)e_l. \tag{33}$$

Define $\psi_1 = ((I - D_0)^{-2}D_1)e_l$ and $\psi_i = (I - D_0)^{-1}D_1\psi_{i-1}$, for $i > 1$, then $E[Y_i - Y_{i-1}] = \alpha\psi_i$. Hence, the expected reset time of the Markov chain $\widetilde{P}_{k,n}$ can be written as

$$E[\text{reset}] = \sum_{i \geq 1} P[Z_{k,n} \geq i]\alpha\psi_i, \tag{34}$$

where the probabilities $P[Z_{k,n} \geq i]$ decrease to zero and can be computed by means of a k-fold convolution or explicitly as

$$P[Z_{k,t} \geq i] = 1 - P[Z_{k,t} < i] = 1 - \sum_{j=k}^{i-1} \binom{j-1}{k-1} \left(\frac{k}{t+1}\right)^k \left(1 - \frac{k}{t+1}\right)^{j-k}. \tag{35}$$

APPENDIX C: COMPUTING THE STATIONARY VECTOR OF AN r-RESET QBD

To compute the steady state vector π of P (see Eqn. (28)), we shall construct a level dependent QBD Markov chain (Latouche[6]) with transition matrix P_{QBD}, by exploiting the structure of the matrices C_0 and C_1. The matrix P_{QBD} is set up such that, when censored on the states of P only, P_{QBD} coincides with P. The set of states of P that correspond to having i, for $i \geq 0$, customers in the queue is referred to as level i of the Markov chain P. To construct P_{QBD} we add a single state to each level of the Markov chain P and we call this state the artificial state of level i. The idea behind this construction is the following: whenever a reset event occurs, P_{QBD} enters the artificial state of the current level i. Next, if $r > i$, $r - i$ transitions between artificial states will follow, each one increasing the level by one. Similarly, if $r < i$, $i - r$ transitions will follow, each one decreasing the level by one. Finally, when level r is reached, we make a transition from the artificial state of level r to one of the other states of level r, using the vector γ. Hence,

$$P_{QBD} = \begin{bmatrix} B_1^r & B_0^r & 0 & \dots & 0 & 0 & 0 & 0 & \dots \\ B_2^r & A_1^{<r} & A_0^{<r} & \dots & 0 & 0 & 0 & 0 & \dots \\ 0 & A_2^{<r} & A_1^{<r} & \ddots & 0 & 0 & 0 & 0 & \dots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots & \ddots \\ 0 & 0 & 0 & \ddots & A_1^{<r} & A_0^{<r} & 0 & 0 & \dots \\ 0 & 0 & 0 & \ddots & A_2^r & A_1^r & A_0^r & 0 & \ddots \\ 0 & 0 & 0 & \ddots & 0 & A_2^{>r} & A_1^{>r} & A_0^{>r} & \ddots \\ 0 & 0 & 0 & \ddots & 0 & 0 & A_2^{>r} & A_1^{>r} & \ddots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots & \vdots & \ddots & \ddots \end{bmatrix}, \tag{36}$$

where

$$\begin{aligned} A_0^r = A_0^{>r} &= \begin{bmatrix} A_0 & 0 \\ 0 & 0 \end{bmatrix}, \quad A_0^{<r} = \begin{bmatrix} A_0 & 0 \\ 0 & 1 \end{bmatrix}, \quad B_0^r = \begin{bmatrix} B_0 & 0 \\ 0 & 1 \end{bmatrix}, \\ A_1^{<r} = A_1^{>r} &= \begin{bmatrix} A_1 & \gamma_1 \\ 0 & 0 \end{bmatrix}, \quad A_1^r = \begin{bmatrix} A_1 & \gamma_1 \\ \gamma & 0 \end{bmatrix}, \quad B_1^r = \begin{bmatrix} B_1 & \gamma_0 \\ 0 & 0 \end{bmatrix}, \\ A_2^r = A_2^{<r} &= \begin{bmatrix} A_2 & 0 \\ 0 & 0 \end{bmatrix}, \quad A_2^{>r} = \begin{bmatrix} A_2 & 0 \\ 0 & 1 \end{bmatrix}, \quad B_2^r = \begin{bmatrix} B_2 & 0 \\ 0 & 0 \end{bmatrix}. \end{aligned} \tag{37}$$

Making use of the results presented in Latouche[6], we can develop the following algorithm to compute $\hat{\pi}$, the invariant vector of P_{QBD}. Denote

$\hat{\pi} = (\hat{\pi}_0, \hat{\pi}_1, \dots)$, where $\hat{\pi}_0$ and $\hat{\pi}_i$, for $i > 0$, have the same dimension as B_1^r and A_1^r. We start by defining a set of r matrices denoted as $G_2, \dots, G_r$ and $G_{>r}$. The matrix $G_{>r}$ is computed by solving the following equation:

$$G_{>r} = A_2^{>r} + A_1^{>r} G_{>r} + A_0^{>r} (G_{>r})^2. \tag{38}$$

This equation is of the same form as Eqn. (23) and is solved using the CR algorithm (Meini[12]). Next, the remaining $r-1$ matrices G_i are found in a backward order:

$$G_r = (I - A_1^r - A_0^r G_{>r})^{-1} A_2^r, \tag{39}$$

$$G_i = (I - A_1^{<r} - A_0^{<r} G_{i+1})^{-1} A_2^{<r}, \tag{40}$$

for $i = r-1, r-2, \dots, 2$. Using these r matrices, we proceed by introducing the matrices $R_0, \dots, R_r$ and $R_{>r}$:

$$R_0 = B_0^r (I - A_1^{<r} - A_0^{<r} G_2)^{-1}, \tag{41}$$

$$R_i = A_0^{<r} (I - A_1^{<r} - A_0^{<r} G_{i+2})^{-1}, \tag{42}$$

$$R_{r-1} = A_0^{<r} (I - A_1^r - A_0^r G_{>r})^{-1}, \tag{43}$$

$$R_r = A_0^r (I - A_1^{>r} - A_0^{>r} G_{>r})^{-1}, \tag{44}$$

$$R_{>r} = A_0^{>r} (I - A_1^{>r} - A_0^{>r} G_{>r})^{-1}, \tag{45}$$

for $i = 1, \dots, r-2$. Applying Theorem 3.2 from Laoutche[7], the steady state vector $\hat{\pi}$ can be expressed as:

$$\hat{\pi}_0 = \hat{\pi}_0 \left(B_1^r + R_0 B_2^r \right), \tag{46}$$

$$\hat{\pi}_i = \hat{\pi}_{i-1} R_{i-1}, \tag{47}$$

$$\hat{\pi}_{r+j} = \hat{\pi}_{r+1} (R_{>r})^{j-1}, \tag{48}$$

for $i = 1, \dots, r+1$ and $j \geq 2$. The vector $\hat{\pi}_0$ is normalized as

$$\hat{\pi}_0 e + \hat{\pi}_0 \sum_{k=0}^{r-1} \left(\prod_{j=0}^{k} R_j \right) e + \hat{\pi}_0 \left(\prod_{j=0}^{r} R_j \right) (I - R_{>r})^{-1} e = 1. \tag{49}$$

Having obtained the vector $\hat{\pi}$, we can derive π, the invariant vector of P, as follows. Write $\hat{\pi}_0$ and $\hat{\pi}_i$, for $i > 0$, as $(\hat{\pi}_0^f, \hat{\pi}_0^{art})$ and $(\hat{\pi}_i^f, \hat{\pi}_i^{art})$, respectively, where both $\hat{\pi}_0^{art}$ and $\hat{\pi}_i^{art}$ are scalars. When censored on all the non-artificial states, P_{QBD} coincides with P, therefore,

$$\pi_i = \hat{\pi}_i^f / (1 - c), \tag{50}$$

where $i \geq 0$ and $c = \sum_{i \geq 0} \hat{\pi}_i^{art}$.

ACKNOWLEDGEMENT

The authors would like to thank T. Hofkens for providing us with a MATLAB program to compute the exact waiting time distribution of the nth customer via numerical inversion of the generating function involved. We also thank the reviewers for their valuable comments that improved the presentation of the paper.

REFERENCES

1. Blondia, C. A discrete-time batch markovian arrival process as B-ISDN traffic model. Belgian Journal of Operations Research, Statistics and Computer Science **1993**, *32* (3,4), 3–23.
2. Carmo, R.M.L.R.; de Souza e Silva, E.; Marie, R. Efficient solutions for an approximation technique for the transient analysis of markovian models. Technical report, IRISA Publication Interne. **1996**, N 1067.
3. Choudhury, G.L.; Lucantoni, D.M.; Whitt, W. Multidimensional transform inversion with applications to the transient M/G/1 queue. Annals of Applied Probability **1994**, *4*, 719–740.
4. Hofkens, T.; Spaey, K.; Blondia, C. Transient analysis of the D-BMAP/G/1 queue with an application to the dimensioning of a playout buffer for VBR traffic. In *Proc. of Networking 2004*; Athens, Greece, 2004.
5. Lang, A.; Arthur. J. L. Parameter approximation for phase-type distributions. In *Matrix-Analytic Methods in Stochastic Models*; Chakravarthy, S.R., Alfa, A.S., Eds.; Marcel-Dekker, Inc: New York, 1996; 151–206.
6. Latouche, G.; Pearce, C.E.M.; Taylor, P.G. Invariant measure for quasi-birth-and-death processes. Stochastic Models **1998**, *14* (1&2), 443–460.
7. Latouche, G.; Ramaswami, V. *Introduction to Matrix Analytic Methods and Stochastic Modeling*. SIAM: Philadelphia, 1999.
8. Lucantoni, D.M. New results on the single server queue with a batch markovian arrival process. Stochastic Models **1991**, *7* (1), 1–46.
9. Lucantoni, D.M.; Choudhury, G.L.; Whitt, W. The transient BMAP/PH/1 queue. Stochastic Models **1994**, *10*, 461–478.
10. Lucantoni, D.M.; Meier-Hellstern, K.S.; Neuts, M.F. A single server queue with server vacations and a class of non-renewal arrival processes. Advances in Applied Probability **1990**, *22*, 676–705.
11. Meini, B. *Fast Algorithms for the Numerical Solution of Structured Markov Chains*. PhD thesis, University of Pisa, 1998.
12. Meini, B. Solving QBD problems: the cyclic reduction algorithm versus the invariant subspace method. Advances in Performance Analysis **1998**, *1*, 215–225.
13. Neuts, M.F. *Matrix-Geometric Solutions in Stochastic Models, An Algorithmic Approach*; John Hopkins University Press, 1981.
14. Neuts, M.F. *Structured Stochastic Matrices of M/G/1 Type and Their Applications*; Marcel Dekker, Inc.: New York and Basel, 1989.
15. Ny, L.M. Le; Sericola, B. Transient analysis of the BMAP/PH/1 queue. I.J. of Simulation **2003**, *3* (3-4), 4–14.
16. Ross, S.M. Approximating transient probabilities and mean occupation times in continues-time markov chains. Probability in the Engineering and Informational Sciences **1987**, *1*, 251–264.
17. Telek, M. Minimal coefficient of variation of discrete phase type distributions. In *3rd International Conference on Matrix Analytic Methods in Stochastic Models*; Notable Publications Inc.: Leuven, Belgium, 2000; 391–400.