Applied Probability Trust (7 January 2002)

THE DELAY DISTRIBUTION OF A TYPE K CUSTOMER IN A FCFS MMAP[K]/PH[K]/1 QUEUE.

B. VAN HOUDT AND C. BLONDIA,[*] *University of Antwerp*

Abstract

This paper presents an algorithmic procedure to calculate the delay distribution of a type k customer in a first-come-first-serve (FCFS) discrete-time queueing system with multiple types of customers, where each type has different service requirements (the MMAP[K]/PH[K]/1 queue). First, we develop a procedure, using matrix analytical methods, to handle arrival processes that do not allow batch arrivals to occur. Next, we show that this technique can be generalized to arrival processes that do allow batch arrivals to occur. We end the paper by presenting some numerical examples.

Keywords: Queueing Theory, Delay Distribution, first-come-first-serve, Multiple Customer Types, Matrix Analytic Methods, Phase Type Distribution, Markovian Arrival Process.

AMS 2000 Subject Classification: Primary 60K25
Secondary 60M20, 90B22

1. Introduction

In this paper we study a class of queues with multiple types of customers, where each type has different service requirements, known as the discrete time MMAP[K]/PH[K]/1 queue. The MMAP[K] arrival process, introduced in [7], is a Markovian arrival process that generates customers of K different types and is a generalization of the batch Markovian arrival process (BMAP). Its potential applications to telecommunications, manufacturing and service industries have been demonstrated extensively in [7, 4].

Queues with MMAP[K] input, i.e., MMAP[K]/G[K]/1 queues, with a first-come-first-served (FCFS) service discipline have been studied in [4, 2, 13, 14]. Within these

[*] Postal address: Department of Mathematics and Computer Science, Performance Analysis of Telecommunication Systems Research Group, Universiteitsplein, 1, B-2610 Antwerp - Belgium, {vanhoudt,blondia}@uia.ua.ac.be

papers, explicit formulas for the Laplace Stieltjes Transform (LST) of the actual waiting times of a customer of type k were obtained. A lot of research effort has been done for the analysis of the MMAP[K]/PH[K]/1 queue with a last-come-first-serve (LCFS) service discipline, e.g., [6, 5]. In [6], the authors present an algorithmic procedure to calculate the steady state probabilities of the MMAP[K]/PH[K]/1/LCFS-GPR queue using tree structured Quasi-Birth-Death (QBD) Markov chains [15] (where GPR stands for generalized preemptive resume).

In this paper, we develop a simple algorithmic procedure to calculate the delay distribution of a type k customer for a discrete-time MMAP[K]/PH[K]/1 queue using Markov chains of the $GI/M/1$ type [10, 9]. Markov chains of the $M/G/1$ and $GI/M/1$ type have been used in the past to study some of the more classical queues [11, 8]. The method introduced in this paper can also be used as an alternative way to obtain the delay distribution related to some of the more classical queues. In Section 2 we start by restricting ourselves to MMAP[K] arrival processes that do not allow batch arrivals to occur ([6, 2] do not allow batches either). Afterwards, we extend our method to more general MMAP[K] processes. We end this paper by presenting some numerical examples that further demonstrate the usefulness of these queueing systems.

2. The discrete-time MMAP[K]/PH[K]/1 queue

The arrival process of the queueing system of interest is a discrete time Markov arrival process with marked transitions (MMAP[K]) that does not allow batch arrivals. Customers are distinguished into K different types. The MMAP[K] is characterized by a set of $m \times m$ matrices $\{D_k \mid 0 \leq k \leq K\}$, with m a positive integer. The $(j_1, j_2)^{th}$ entry of the matrix D_k, for $k > 0$, represents the probability that a customer of type k arrives and the underlying Markov chain makes a transition from state j_1 to state j_2. The matrix D_0 covers the case when there are no arrivals. The matrix D, defined as

$$D = \sum_{k=0}^{K} D_k,$$

represents the stochastic $m \times m$ transition matrix of the underlying Markov chain of the arrival process. Let θ be the stationary probability vector of D, that is, $\theta D = \theta$ and $\theta e = 1$, where e is a column vector with all entries equal to one. The stationary arrival rate of type k customers is given by $\lambda_k = \theta D_k e$.

The service times of type k customers have a common phase-type distribution function with a matrix representation (m_k, α_k, T_k), where m_k is a positive integer, α_k is an $1 \times m_k$ nonnegative stochastic vector and T_k is an $m_k \times m_k$ substochastic matrix. Let $T_k^0 = e - T_k e$, then the mean service time of a type k customer equals $1/\mu_k = \alpha_k (I - T_k)^{-1} e$. Define $m_{ser} = \sum_{k=1}^{K} m_k$, the $m_{ser} \times m_{ser}$ matrix T_{ser} and the $m_{ser} \times 1$ vector T_{ser}^0 as

$$T_{ser} = \begin{bmatrix} T_1 & 0 & \dots & 0 \\ 0 & T_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & T_K \end{bmatrix}, \qquad T_{ser}^0 = \begin{bmatrix} T_1^0 \\ T_2^0 \\ \vdots \\ T_K^0 \end{bmatrix}.$$

Let $m_{tot} = m_{ser} m$. The customers are served, by a single server, according to a first-come-first-serve (FCFS) service discipline.

2.1. Constructing a $GI/M/1$ Type Markov chain (MC)

In this section, we indicate how to calculate the delay distribution of a type k customer by creating a $GI/M/1$ type Markov chain with a generalized initial condition. As opposed to the general approach in many queueing systems, we calculate the delay distribution without obtaining the steady state probabilities of the queue length. The trick used in this section is to keep track of the "age" of the customer in service, while keeping the MMAP[K] state constant until the service is completed.

Consider a Markov chain (MC) with an infinite number of states labeled $1, 2, \ldots$. The set of states $\{1, \ldots, m\}$ is referred to as level zero of the MC, whereas the set of states $\{(i-1)m_{tot} + m + 1, \ldots, i m_{tot} + m\}$ is referred to as level i of the MC. The states of level $i > 0$ are labeled as (k, s, j), where $1 \leq k \leq K$, $1 \leq s \leq m_k$ and $1 \leq j \leq m$. Let state j of level zero of the MC correspond to the situation in which the queue and the server are empty, while the current state of the MMAP[K] is j. Let state (k, s, j) of level i of the MC correspond to the situation in which there is a customer of type k in service, that arrived i time instances ago, while the service process is currently in phase s and the MMAP[K] arrival process was in state j at time $n - i + 1$, where n is the current time instance.

The level of the Markov chain can never increase by more than one during a transition between time instance n and $n+1$. Moreover, the probability of making a transition between state (k_1, s_1, j_1) of level $i_1 > 0$ and state (k_2, s_2, j_2) of level $i_2 > 0$ does not depend upon i_1 and i_2, but only upon the difference between i_1 and i_2. Therefore, the system can be described by a transition matrix P with the following structure:

$$P = \begin{bmatrix} B_1 & B_0 & 0 & 0 & 0 & \dots \\ B_2 & A_1 & A_0 & 0 & 0 & \dots \\ B_3 & A_2 & A_1 & A_0 & 0 & \dots \\ B_4 & A_3 & A_2 & A_1 & A_0 & \dots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots \end{bmatrix}, \qquad (1)$$

where A_l are $m_{tot} \times m_{tot}$ matrices, $B_l, l > 1$, are $m_{tot} \times m$ matrices, B_1 is an $m \times m$ matrix and B_0 is an $m \times m_{tot}$ matrix. The matrices B_0, resp. B_1, represent the probabilities of making a transition from level zero to level zero, resp. level one. The matrices A_l represent the transition probabilities between level $i \geq l$ and level $i - l + 1$, whereas B_l holds the probabilities of making a transition from level $l-1$ to level zero of the MC.

In order to express the matrices A_l and B_l, for $l \geq 0$, we define the following $m \times m_{tot}$ matrix L:

$$L = [(\alpha_1 \otimes D_1)\ (\alpha_2 \otimes D_2)\ \dots\ (\alpha_K \otimes D_K)].$$

The entries of the matrix L hold the probabilities that the MMAP[K] arrival process makes a transition from state j_1 to state j_2, with $1 \leq j_1, j_2 \leq m$, while a type k, $1 \leq k \leq K$, customer arrives and the customer will start its service in phase s, with $1 \leq s \leq m_k$. Let $L_l = (D_0)^{l-1} L$ for $l \geq 1$. Based on the probabilistic interpretation of the matrices A_l and B_l we find:

$$\begin{aligned} A_0 &= T_{ser} \otimes I_m, \\ A_l &= T_{ser}^0 \otimes L_l, \\ B_0 &= L, \\ B_1 &= D_0, \\ B_l &= T_{ser}^0 \otimes (D_0)^{l-1}, \end{aligned}$$

where $\otimes$ denotes the Kronecker product between matrices and I_m the $m \times m$ unity matrix. Notice that the matrices A_l and B_l decrease to zero according to $(D_0)^l$. Looking at the probabilistic interpretation of D_0, it should be clear that, in general, the smaller the arrival rate $\lambda = \sum_{k=1}^{K} \lambda_k$ the slower A_l and B_l decrease to zero.

The $GI/M/1$ type MC defined above observes the system at each time instance, even during the time instances when the server is empty, in which case the Markov chain is at level zero. It is also possible to create a $GI/M/1$ type MC that observes the system only at time instances when the server is busy. Moreover, the matrices A_l, for $l \geq 0$, would be identical to those defined above. However, the matrices B_l, for $l \geq 0$, would have a different dimension and different equations for B_l would apply. Both approaches lead to the same results and have a similar time and space complexity.

2.2. Calculating the Steady-State Probabilities

For some MMAP[K] arrival processes, the MC defined in the previous section contains some obvious transient states. Indeed, the states (k, s, j), for $1 \leq s \leq m_k$ at level $i > 0$ are all transient, whenever the j-th component of the vector θD_k is zero (which indicates that the MMAP[K] cannot be in state j after generating a type k customer). We could easily eliminate the rows and columns of A_l and B_l that correspond to these states, however, this is not necessary because the algorithm that computes the steady state probabilities will automatically produce a zero for these transient states. However, if a high percentage of the states is transient, it is worth to eliminate them as this will reduce the computation time.

Whenever we state that P is ergodic, we mean to say that P is ergodic after removing the obvious transient states mentioned above. A proof that the MC defined by equation (1) is ergodic if and only if $\rho < 1$, where $\rho = \sum_{k=1}^{K} \lambda_k / \mu_k$, is provided in the appendix. In [3], it was also shown that the MMAP[K]/PH[K]/1 queue is stable if and only if $\rho < 1$; therefore, the waiting times have a limiting distribution.

Define $\pi_i^n(k, s, j), i > 0$, resp. $\pi_0^n(j)$, as the probability that the system is in state (k, s, j) of level i, resp. state j of level zero, at time instance n. Let

$$\begin{aligned} \pi_0(j) &= \lim_{n \to \infty} \pi_0^n(j), \\ \pi_i(k, s, j) &= \lim_{n \to \infty} \pi_i^n(k, s, j). \end{aligned}$$

Define the $1\times m$ vector $\pi_0 = (\pi_0(1),\dots,\pi_0(m))$ and the $1\times m_{tot}$ vectors $\pi_i = (\pi_i(1,1,1),$ $\pi(1,1,2),\ \dots,\pi_i(1,1,m),\ \pi_i(1,2,1),\dots,\pi_i(1,2,m),\ \pi_i(1,3,1),\dots,\pi_i(K,m_K,m)),$ for $i>0$. From the transition matrix P, defined in equation (1), we see that the Markov chain is a generalized Markov chain of the $GI/M/1$ Type [9]. For such a positive recurrent Markov chain, we have $\pi_i = \pi_{i-1}R, i>1$, where R is an $m_{tot}\times m_{tot}$ matrix that is the smallest nonnegative solution to the following equation:

$$R = \sum_{l\geq 0} R^l A_l.$$

This equation is solved by means of an iterative scheme [9, 12]. In order to obtain π_0 and π_1 we solve the following equation

$$(\pi_0,\pi_1) = (\pi_0,\pi_1)\begin{bmatrix} B_1 & B_0 \\ \sum_{l\geq 2} R^{l-2}B_l & \sum_{l\geq 1} R^{l-1}A_l \end{bmatrix}.$$

The vector (π_0,π_1) is normalized as $\pi_0 e_m + \pi_1(I-R)^{-1}e_{m_{tot}} = 1$, where I is the unity matrix of size m_{tot} and e_l is an $l\times 1$ vector whose elements equal one.

2.3. Calculating the Delay Density Function

Let d_k be the random variable that denotes the delay suffered by a type k customer. Notice, the delay is defined as the sum of the time that the customer spends in the queue and the time spent in the server. The probability that a type k customer has a delay of i time units, can be calculated as the expected number of type k customers with an "age" of i time units that complete their service at an arbitrary time instance, divided by the expected number of type k customers that complete their service during an arbitrary time instance (that is, λ_k for a stable queue). Using the steady state probabilities we easily find, by noticing that the MC defined in Section 2.1 observes the system at each time instance,

$$P[d_k = i] = \sum_{s=1}^{m_k} \frac{(T_k^0)_s}{\lambda_k} \sum_{j=1}^{m} \pi_i(k,s,j),$$

for $i\geq 1$, with λ_k the arrival rate of the type k customers. $(T_k^0)_s$ represents the s-th component of the column vector T_k^0. Notice, $P[d_k=0]=0$, because a customer spends at least one time unit in the server. Thus, using this procedure, we are able to calculate the delay distribution without any knowledge of the queue length.

3. A MMAP[2]/PH[2]/1 queue with Batch Arrivals

We start by considering a simple example of a MMAP[K] queue with batch arrivals and $K = 2$ customer types, and develop a procedure to calculate the delay distribution for a type $k = 1, 2$ customer. In the next section, we generalize this idea to an arbitrary MMAP[K] arrival process with batch arrivals.

Consider a single server queue with two correlated input sources A and B. Both sources generate zero or one customer during a time instance. Therefore, we can model the input traffic as a MMAP[2] arrival process characterized by the $m \times m$ matrices D_0, D_1, D_2, D_{12} and D_{21}, where the $(j_1, j_2)^{th}$ element of the matrix D_C, with $C = c_1, \ldots, c_b$ a string of $b = 1, 2$ integers between 1 and $K = 2$, represents the probability of having a batch of $b = 1, 2$ arrivals, while the underlying Markov chain makes a transition from state j_1 to j_2. The first customer of the batch is of type c_1, the possible second of type c_2. We assume that the service time of a type $k = 1, 2$ customer follows a phase-type distribution function with matrix characterization (m_k, α_k, T_k).

In order to study this MMAP[2]/PH[2]/1 queue, we define a new MMAP[2] arrival process characterized by the following $3m \times 3m$ matrices:

$$\tilde{D_0} = \begin{bmatrix} D_0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \tilde{D_1} = \begin{bmatrix} D_1 & 0 & D_{12} \\ I_m & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \tilde{D_2} = \begin{bmatrix} D_2 & D_{21} & 0 \\ 0 & 0 & 0 \\ I_m & 0 & 0 \end{bmatrix}.$$

This MMAP[2] process is identical to the first, except that time instances during which a batch of size two occurs, are replaced by two time instances each holding a single customer. In order to obtain the delay distribution of the original system, we cannot simply calculate the delay distribution related to the new MMAP[2], because the splitting of the size two batches would result in an error of one time unit to the age of the second customer of the batch. Moreover, looking at the MC of Section 2.1, we know that the age of a customer is determined using the age of the previous (unless it arrives when the queue is empty); therefore, the error would propagate and result in an incorrect age for all future customers.

Nevertheless, the method of the previous section still works if we somehow manage to correct the age of each customer that is generated while the new MMAP[2] process is in a state $m < j \leq 3m$, by one time unit when it enters the server. Indeed, by definition, such a customer enters the server as soon as the previous customer leaves the server; therefore, its age is based upon the age of the previous customer. This correction of one time unit can be realized by changing the transition matrices A_1 and A_0 accordingly. Let us explain this procedure in more detail. If a customer, with age i, completes its service at time instance n and the next customer was generated one time unit after the one that completed its service, the MC would, according to the previous section, still be at level i at time instance $n+1$. If we add one time unit to the age of such a customer, provided that it was generated while the state of the new MMAP[2] was $j > m$, in order to get the correct age, the MC would have to be at level $i+1$ at time instance $n+1$. Finally, a small modification to level zero of the MC is also made because the MMAP[2] process can never be in a state $j > m$ if the server is empty.

The remainder of this section applies to all MMAP[K] arrival processes and not merely the MMAP[2] arrival process described above. Suppose that we wish to calculate the delay distribution of a type k customer in a MMAP[K]/PH[K]/1 queue, where the MMAP[K] allows for batch arrivals to occur. Moreover, suppose that the MMAP[K] is characterized by a set of $m \times m$ matrices D_C, where C is a string of integers between 1 and K. Then, we start by constructing a new MMAP[K] characterized by the $am \times am$ matrices $\tilde{D}_0, \tilde{D}_1, \ldots, \tilde{D}_K$, with $a \geq 2$ an integer. A general procedure to construct the new MMAP[K] is presented in the next section. Afterwards, we follow the procedure outlined below.

Construct a $GI/M/1$ type MC characterized by the following transition matrix:

$$\tilde{P} = \begin{bmatrix} \tilde{B}_1 & \tilde{B}_0 & 0 & 0 & 0 & \ldots \\ \tilde{B}_2 & \tilde{A}_1 & \tilde{A}_0 & 0 & 0 & \ldots \\ \tilde{B}_3 & \tilde{A}_2 & \tilde{A}_1 & \tilde{A}_0 & 0 & \ldots \\ \tilde{B}_4 & \tilde{A}_3 & \tilde{A}_2 & \tilde{A}_1 & \tilde{A}_0 & \ldots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots \end{bmatrix}, \tag{2}$$

where $\tilde{A}_l$ are $am_{tot} \times am_{tot}$ matrices, $\tilde{B}_l, l > 1$, are $am_{tot} \times m$ matrices, $\tilde{B}_1$ is an $m \times m$ matrix and $\tilde{B}_0$ is an $m \times am_{tot}$ matrix. Let $\tilde{D}_k^f$ be the first m rows of the matrix $\tilde{D}_k$ and $\tilde{D}_k^r$ the remaining $am - m$ rows, for $1 \leq k \leq K$. Next, define the $am \times am_{tot}$ matrix $\tilde{L}$ as

$$\tilde{L} = \left[(\alpha_1 \otimes \tilde{D}_1)\ (\alpha_2 \otimes \tilde{D}_2)\ \ldots\ (\alpha_K \otimes \tilde{D}_K) \right]. \tag{3}$$

Let $\tilde{L}_l = (\tilde{D}_0)^{l-1}\tilde{L}$, for $l \geq 2$. Then, as a result of our prior discussion, we find

$$\begin{aligned} \tilde{B}_1 &= D_0, && (4)\\ \tilde{B}_0 &= \left[(\alpha_1 \otimes \tilde{D}_1^f)\ (\alpha_2 \otimes \tilde{D}_2^f)\ \ldots\ (\alpha_K \otimes \tilde{D}_K^f)\right], && (5)\\ \tilde{B}_l &= T_{ser}^0 \otimes \begin{bmatrix} (D_0)^{l-1} \\ \mathbf{O}_{(a-1)m,m} \end{bmatrix}, && (6)\\ \tilde{A}_l &= T_{ser}^0 \otimes \tilde{L}_l, && (7) \end{aligned}$$

for $l \geq 2$ and where $\mathbf{O}_{x,y}$ is a zero matrix with x rows and y columns. Notice, only m rows of each am rows of $\tilde{B}_l$ and $\tilde{A}_l$, for $l \geq 2$, differ from zero, due to the structure of $\tilde{D}_0$ (see Section 4). It remains to calculate $\tilde{A}_0$ and $\tilde{A}_1$. In order to make the one time unit correction as discussed before, we need to shift the probabilities of A_1 related to the MMAP[K] states $j > m$ to A_0. Hence,

$$\begin{aligned} \tilde{A}_0 &= T_{ser} \otimes I_{am} + T_{ser}^0 \otimes \begin{bmatrix} \mathbf{O}_{m,m_1am} & \mathbf{O}_{m,m_2am} & \cdots & \mathbf{O}_{m,m_Kam} \\ (\alpha_1 \otimes \tilde{D}_1^r) & (\alpha_2 \otimes \tilde{D}_2^r) & \ldots & (\alpha_K \otimes \tilde{D}_K^r) \end{bmatrix}, && (8)\\ \tilde{A}_1 &= T_{ser}^0 \otimes \begin{bmatrix} (\alpha_1 \otimes \tilde{D}_1^f) & (\alpha_2 \otimes \tilde{D}_2^f) & \ldots & (\alpha_K \otimes \tilde{D}_K^f) \\ \mathbf{O}_{(a-1)m,m_1am} & \mathbf{O}_{(a-1)m,m_2am} & \cdots & \mathbf{O}_{(a-1)m,m_Kam} \end{bmatrix}, && (9) \end{aligned}$$

where I_l is the unity matrix of dimension l. The remainder of the procedure is identical to Section 2, that is, we simply replace A_l and B_l by $\tilde{A}_l$ and $\tilde{B}_l$ in all the formulas in Section 2.2 and 2.3 (and in some cases m by am and m_{tot} by am_{tot}).

4. The MMAP[K]/PH[K]/1 queue with Batch Arrivals

Consider a MMAP[K] arrival process characterized by a set of $m \times m$ matrices D_C where C is a string of integers between 1 and K, that is, $C = c_1 \ldots c_b$ with $1 \leq c_l \leq K$ and $1 \leq l \leq b$. Let b_{max} be the maximum batch size of the MMAP[K] arrival process. We state that a string C_1 *extends* a string $C_2 = c_1^2 \ldots c_b^2$ if C_1 is of the form $C_1 = c_1^1 \ldots c_l^1 c_1^2 \ldots c_b^2$ for some integer $l \geq 1$. Let $\mathcal{C} = \{C \mid \exists C_1 \text{for which} C_1$ *extends* C and $D_{C_1} \neq 0\}$ and let $|\mathcal{C}|$ be the number of strings in the set $\mathcal{C}$. The empty string $\emptyset$ not considered as a member of $\mathcal{C}$. Notice, $|\mathcal{C}| \leq \sum_{b=1}^{b_{max}-1} K^b = \frac{K^{b_{max}}-1}{K-1} - 1$. Finally, define a as $|\mathcal{C}| + 1$.

Next, we construct a new MMAP[K] arrival process that is identical to the first, except that each time instance in which a batch of $b \leq b_{max}$ customers occurs, is replaced by b time instances each holding one customer (in the same order as in the batch). The new MMAP[K] is characterized by the $am \times am$ matrices $\tilde{D}_0, \tilde{D}_1, \ldots, \tilde{D}_K$. The matrix $\tilde{D}_0$ is equal to zero except for the $m \times m$ block in the upper left corner which equals D_0. In order to describe the matrices $\tilde{D}_k$, with $1 \leq k \leq K$, we start by labeling the am states of the arrival process as follows. The first m states are labeled as the empty string $\emptyset$. The remaining $m|\mathcal{C}|$ states are grouped into $|\mathcal{C}|$ sets of m states and each set is labeled by a string $C \in \mathcal{C}$ such that each set of m states has a unique label.

Let $(\tilde{D}_k)_{C_1,C_2}$ be the $m \times m$ submatrix of $\tilde{D}_k$ that holds the probabilities of making a transition from the m states labeled C_1 to the m states with label C_2, while a type k customer is generated. Then, we define $(\tilde{D}_k)_{C_1,C_2} = I_m$ provided that $C_1 = kc_1^2 \ldots c_b^2$, where $C_2 = c_1^2 \ldots c_b^2$. Notice, $C = k$ is considered identical to $C = k\emptyset$. The other $m \times m$ submatrices $(\tilde{D}_k)_{C_1,C_2}$, for $C_1 \neq \emptyset$, are equal to zero. The submatrices $(\tilde{D}_k)_{\emptyset,C_2}$ are equal to D_C, with $C = kC_2$.

In conclusion, to obtain the delay distribution of a type k customer in a MMAP[K]/PH[K]/1 queue, where the MMAP[K] allows for batch arrivals to occur, we simply construct the MMAP[K] arrival process characterized by the matrices $\tilde{D}_0, \tilde{D}_1, \ldots, \tilde{D}_K$ and apply the procedure described by equations (2) to (9). Obviously, the dimension of the matrices $\tilde{A}_l$ and $\tilde{B}_l$ should not exceed a few hundred, otherwise the procedure is too time and memory consuming. It is important to note that the new MC characterized by $\tilde{P}$ might contain a very high number of obvious transient states due to the construction of the new MMAP[K]. Indeed, the states (k, s, j), for $1 \leq s \leq m_k$, at level $i > 0$ are transient if the j-th component of $\tilde{\theta}\tilde{D}_k$ equals zero, where $\tilde{\theta}$ is the stochastic stationary vector of $(\tilde{D}_0 + \ldots + \tilde{D}_K)$. Thus, these states are easy to identify and can be removed without any difficulties, thereby reducing the computation time significantly.

5. Numerical Examples

The idea used in this paper originated while analyzing the performance of FS-ALOHA(++), a random access algorithm used in telecommunication systems [16, 1]. Demonstrating how FS-ALOHA can be evaluated using a MMAP[K]/PH[K]/1 queue would lead us too far. Therefore, we present a rather arbitrary example that illustrates the strength of the algorithmic procedure presented.

Consider a single server queue with three correlated input sources A, B and C; their customers are referred to as type one, two and three. Each sources generates zero or one customer during a time instance. The superposition of these three correlated sources is assumed to be a 3 state MMAP[3]. The three states are traversed one by one and the sojourn time in each state is geometrically distributed with a mean of 1000 time units. While in state one, source A generates a customer with probability 1/5, source C with probability 1/100, while source B is silent. In state two, source A and C generate a customer with probability 1/100, while source B generates a customer with probability 1/28. Finally, in state three, source B generates a customer with probability 1/100, source C with probability 1/20, while source A is silent. Given that we are in state $1 \leq j \leq 3$, the three sources A, B and C are independent (e.g., the probability that a type one and type three customer are generated while in state two is $9.643\ 10^{-5}$). In this example, the majority of the arriving customers while in state j, are customers of type j. We further assume that the batches are ordered, that is, whenever a batch arrival occurs, the type one customer is first, followed by the type two customer and finally the type three customer. Ordering the batches reduces $a = 1 + |\mathcal{C}|$ by a factor 2.5. As a result, the MMAP[K] is characterized by the 3×3 non zero matrices $D_0, D_1, D_2, D_3, D_{12}, D_{13}, D_{23}$ and D_{123}. For instance,

$$D_1 = \begin{bmatrix} 1.978\ 10^{-1} & 1.98\ 10^{-4} & 0 \\ 0 & 9.537\ 10^{-3} & 9.546\ 10^{-6} \\ 0 & 0 & 0 \end{bmatrix}.$$

Clearly, $\mathcal{C} = \{23, 2, 3\}$. Using the procedure in the previous section, we construct a new MMAP[3] that is characterized by the 12×12 matrices $\tilde{D}_0$, $\tilde{D}_1, \ldots, \tilde{D}_3$.

The service times are assumed to be as follows. Type one customers have a deterministic service time of two time units. The service time distribution of a type two customers on the other hand, is phase-type with three phases, being three geometric phases with a mean of two, three and two time units. Finally, type three customers

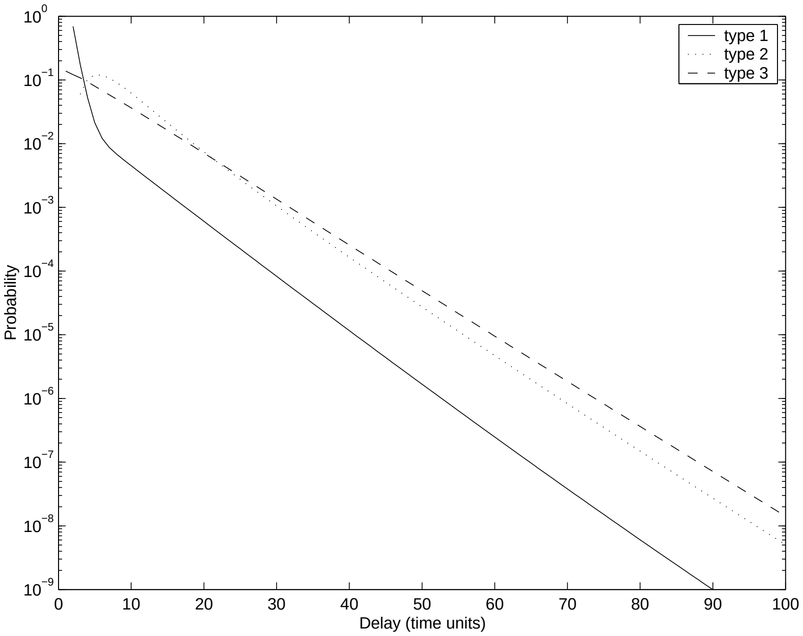

FIGURE 1: Delay distribution of type one, two and three customers

require a geometric service time with a mean of 5 time units. Hence,

$$T_1 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad T_2 = \begin{bmatrix} 1/2 & 1/2 & 0 \\ 0 & 2/3 & 1/3 \\ 0 & 0 & 1/2 \end{bmatrix}, \quad T_3 = [4/5],$$

and $\alpha_1 = [1\ 0]$, $\alpha_2 = [1\ 0\ 0]$, and $\alpha_3 = [1]$. As a result, the matrices $\tilde{A}_l$ are 72×72 matrices. Figure 1 represents the delay distribution of type one, two and three customers, as calculated by equations (2) to (9) and Sections 2.2 and 2.3. The computation time was approximately one minute and thirty seconds on a Sun Enterprise 2170 with two 167 Mhz processors and 3x128 Mbyte RAM. We could further reduce the computation time, if we took the effort to remove the 28 transient states of each level $i > 0$, thereby reducing the dimension of the $\tilde{A}_l$ matrices to 44×44 matrices, instead of 72×72 matrices. These 28 states are easily identified by looking at the zero entries of the vectors $\tilde{\theta}\tilde{D}_k$, for $k \geq 1$.

6. Conclusion

This paper presented an algorithmic procedure to calculate the delay distribution of a type k customer in a first-come-first-serve (FCFS) discrete-time MMAP[K]/PH[K]/1 queueing system. We started by developing a procedure, using matrix analytical methods, for arrival processes that do not allow batch arrivals to occur. Afterwards, we showed that this technique can be generalized to arrival processes that do allow batch arrivals to occur. A numerical example to demonstrate the strength of the procedure was presented within Section 5.

Acknowledgements

B. Van Houdt is a postdoctoral fellow of the FWO Flanders. We would like to thank the reviewers for their valuable comments and suggestions.

Appendix

In this section we present an algebraic proof that the MC defined in Section 2.1 is ergodic if and only if $\rho = \sum_{k=1}^{K} \lambda_k/\mu_k < 1$. Recall that we mean to say that P is ergodic after removing the obvious transient states mentioned at the start of Section 2.2. We start by defining the following $1 \times m_k$ stochastic vectors for $1 \leq k \leq K$:

$$\beta_k = \beta_k \left(T_k + T_k^0 \alpha_k \right). \tag{10}$$

We start by proving the following two equations:

$$\begin{aligned} \beta_k T_k^0 &= \mu_k, & (11) \\ \theta &= \theta \left(\sum_{k=1}^{K} D_k \right) (I - D_0)^{-1}, & (12) \end{aligned}$$

where θ was defined as the stochastic stationary vector of $D = \sum_{k=0}^{K} D_k$. The first equation is obtained from equation (10) by subtracting $\beta_k T_k$ from both sides of the equation, followed by multiplying both sides by $(I - T_k)^{-1} e$ and applying the definition of μ_k. The second equation is easily obtained from $\theta = \theta D$ by subtracting D_0 and multiplying by $(I - D_0)^{-1}$.

Next, we define the $(k,s,j)^{th}$ component, with $1 \leq k \leq K$, $1 \leq s \leq m_k$ and $1 \leq j \leq m$, of the $1 \times m_{tot}$ vector Π_g as

$$\frac{1}{\rho}(\theta D_k)_j \frac{(\beta_k)_s}{\mu_k},$$

where v_j, with v a row or column vector, denotes the j^{th} component of v.

Lemma 1. *The vector* Π_g *is an invariant vector of* $\sum_{l=0}^{\infty} A_l$ *and* Π_g *is stochastic.*

Proof. The sum $\sum_{l=0}^{\infty} A_l$ can be written as $T_{ser} \otimes I_m + T_{ser}^0 \otimes \left((I-D_0)^{-1}L\right)$, where L was defined by equation (3). First, we calulate the $(k',s',j')^{th}$ component of $\Pi_g(T_{ser} \otimes I_m)$. Given the structure of T_{ser} we find

$$\frac{1}{\rho}(\theta D_{k'})_{j'} \frac{\sum_{s=1}^{m_{k'}} (\beta_{k'})_s (T_{k'})_{s,s'}}{\mu_{k'}}.$$

Using equations (10) and (11), we can rewrite this as

$$\frac{1}{\rho}(\theta D_{k'})_{j'} \frac{((\beta_{k'})_{s'} - \mu_{k'}(\alpha_{k'})_{s'})}{\mu_{k'}}. \tag{13}$$

Second, the $(k',s',j')^{th}$ component of $\Pi_g\left(T_{ser}^0 \otimes ((I-D_0)^{-1}L)\right)$ equals

$$\frac{1}{\rho}\sum_{k=1}^{K} \frac{\sum_{s=1}^{m_k}(\beta_k)_s(T_k^0)_s}{\mu_k} \sum_{j=1}^{m} (\theta D_k)_j((I-D_0)^{-1}D_{k'})_{j,j'}(\alpha_{k'})_{s'}.$$

This equation can be simplified using equation (11) to find

$$\frac{1}{\rho}(\theta(D_1 + \ldots + D_K)(I-D_0)^{-1}D_{k'})_{j'}(\alpha_{k'})_{s'}.$$

Or by means of equation (12) we have $\frac{1}{\rho}(\theta D_{k'})_{j'}(\alpha_{k'})_{s'}$. Adding this to equation (13) proofs that Π_g is an invariant vector of $\sum_l A_l$. Π_g is clearly a stochastic vector because β_k is stochastic and $\theta D_k e$ equals λ_k. Q.E.D.

Lemma 2. $\Pi_g\left(\sum_{l=1}^{\infty} lA_l\right)e = 1/\rho$.

Proof. $\left(\sum_{l=1}^{\infty} lA_l\right)e$ can be written as $T_{ser}^0 \otimes ((\sum_{l=1}^{\infty} l(D_0)^{l-1})Le)$. Looking at equation (3), we have $Le = (I-D_0)e$ because the vectors α_k are stochastic and $\left(\sum_{k=0}^{K} D_k\right)e = e$. Moreover, $(\sum_l l(D_0)^{l-1})(I-D_0)$ is equal to $(I-D_0)^{-1}$. Thus, $\left(\sum_{l=1}^{\infty} lA_l\right)e = (T_{ser}^0 \otimes ((I-D_0)^{-1}e))$. As a result, $\Pi_g(\sum_{l=1}^{\infty} lA_l)e$ equals

$$\frac{1}{\rho}\sum_{k=1}^{K} \theta(D_k(I-D_0)^{-1})e \sum_{s=1}^{m_k} \frac{(\beta_k)_s(T_k^0)_s}{\mu_k}.$$

Thus, using equations (11) and (12) results in $1/\rho\ \theta\ e = 1/\rho$.

Q.E.D.

Theorem 1. *The MC defined by the transition matrix P, defined by equation (1), is ergodic if and only if $\rho < 1$.*

Proof. Neuts [9] has shown, provided that the matrix $A = \sum_{l=0}^{\infty} A_l$ is irreducible, that a $GI/M/1$ type MC is ergodic if and only if the product of the stochastic invariant vector of $\sum_l A_l$ with the vector $(\sum_l lA_l)e$ is larger than one. The matrix $\sum_l A_l$ is irreducible, after removing the transient states mentioned at the start of Section 2.2, because $D = D_0 + \ldots + D_K$ and the matrices $(T_k + T_k^0 \alpha_k)$ are irreducible. The entries of Π_g that correspond to the transient states are zero. Moreover, the $(i,j)^{th}$ entry of the matrices $\sum_l A_l$ and $\sum_l lA_l$ equals zero, if state i is not transient, whereas state j is. Therefore, Lemma 1 and 2 suffice to proof the theorem. Q.E.D.

References

[1] Cortizo, D. V., García, J., Blondia, C. and Van Houdt, B. (1999). FIFO by sets ALOHA (FS-ALOHA): a collision resolution algorithm for the contention channel in wireless ATM systems. *Performance Evaluation* **36-37,** 401–427.

[2] He, Q. (1996). Queues with marked customers. *Adv. Appl. Prob.* **28,** 567–587.

[3] He, Q. (2000). Classification of Markov processes of matrix M/G/1 type with a tree structure and its applications to the MMAP[K]/G[K]/1 queue. *Stochastic Models* **16,** 407–434.

[4] He, Q. (2001). The versatility of the MMAP[K] and the MMAP[K]/G[K]/1 queue. *Queueing Systems* **38,** 397–418.

[5] He, Q. and Alfa, A. (1998). The MMAP[K]/PH[K]/1 queues with a last-come-first-serve preemptive service discipline. *Queueing Systems* **28,** 269–291.

[6] He, Q. and Alfa, A. (2000). The discrete time MMAP[K]/PH[K]/1/LCFS-GPR queue and its variants. In *Proc. of the 3rd Int. Conf. on Matrix Analytic Methods.* Leuven (Belgium). pp. 167–190.

[7] He, Q. and Neuts, M. (1998). Markov chains with marked transitions. *Stochastic Processes and their Applications* **74,** 37–52.

[8] Lucantoni, D., Meier-Hellstern, K. and Neuts, M. (1990). A single server queue with server vacations and a class of non-renewal arrival processes. *Adv. Appl. Prob.* **22,** 676–705.

[9] Neuts, M. (1978). Markov chains with applications in queueing theory, which have a matrix geometric invariant probability vector. *Adv. Appl. Prob.* **10,** 185–212.

[10] Neuts, M. (1981). *Matrix-Geometric Solutions in Stochastic Models, An Algorithmic Approach.* John Hopkins University Press.

[11] Neuts, M. (1986). Generalizations of the pollaczek-khinchin integral method in the theory of queues. *Adv. Appl. Prob.* **18,** 952–990.

[12] Ramaswami, V. (1988). Nonlinear matrix equations in applied probability - solution techniques and open problems. *SIAM review* **30,** 256–263.

[13] Takine, T. (2002). Queue length distribution in a FIFO single-server queue with multiple arrival streams having different service time distributions. *To appear in Queueing Systems*.

[14] Takine, T. and Hasegawa, T. (1994). The workload in a MAP/G/1 queue with state-dependent services: its applications to a queue with preemptive resume priority. *Stochastic Models* **10,** 183–204.

[15] Takine, T., Sengupta, B. and Yeung, R. (1995). A generalization of the matrix M/G/1 paradigm for Markov chains with a tree structure. *Stochastic Models* **11,** 411–421.

[16] Van Houdt, B. and Blondia, C. (2002). Robustness of FS-ALOHA. In *Proc of the 4th Int. Conf. on Matrix Analytic Methods (MAM4), to appear.* Adelaide (Australia).