Operations Research Letters 9 (1990) 249–256
North-Holland

July 1990

RECURSIVE COMPUTATION OF STEADY-STATE PROBABILITIES IN PRIORITY QUEUES

Israel CIDON

IBM, Thomas J. Watson Research Center, P.O. Box 704, Yorktown Heights, NY 10598, USA

Moshe SIDI

Electrical Engineering Department, Technion – Israel Institute of Technology, Haifa 32000, Israel

Received May 1988
Revised September 1989

Exact recursive formulas are derived for the state probabilities in priority queueing systems (preemptive and non-preemptive). The derivation is based only on the general structure of the generating function involved, and thus is simpler and more general than previous methods. Furthermore, applications of the method to other queueing systems are discussed.

priority queues * recursive computation * steady-state probabilities

1. Introduction

Queueing models are frequently used in the process of evaluating the performance of practical systems. The most common performance measures for the behavior of a system are the moments (especially first and second moments) of the buffers occupancy. However, in many cases the buffer occupancy probabilities themselves are needed. Frequently, the designer is interested in the probability that the queue length will exceed some predetermined thresholds. In such a case there is a need to calculate the buffer occupancy probabilities up to some large number. A typical example is the performance evaluation of a communication store and forward node that has an hierarchy of memories and each of them has different speed and size. Messages are stored in the fastest memory available to the node. In order to compute the probability of arriving packets to be stored at some level of the memory, we need to know the probabilities of exceeding the corresponding thresholds.

In many queueing systems it is relatively simple to derive the generating function of the number of customers in the system. If this function is not too complex, one would be able to derive the first two moments of the number of customers. However, in most cases, obtaining the probability distribution might prove to be a challenging task. An example of such a system is a priority (preemptive or non-preemptive) queueing system (Jaiswal [5]). The goal of this paper is to introduce a simple derivation of recursive formulas for the state probabilities in a large class of queueing systems including priority queueing systems. The details of the recursion are given in Sections 2 and 3 for the preemptive and the non-preemptive single-server systems, respectively. In Section 4 we show that the same technique can be applied to multi-server priority systems, and we also discuss the applications of the method to systems other than priority systems.

White and Christie [12] derived complicated expressions for the probability distribution for the number of customers in a preemptive priority queueing system. Recursive computation of the *joint* state probabilities for the two priority classes, has been presented by Marks [6] and Miller [7], for the preemptive and the non-preemptive priority (with the same exponential service distribution for both classes of customers in [6]) systems. The method of [6] is by far more complicated and hard to generalize than the method of [7] and the method that we present in this paper. Miller [7] derived his recursive formulas by using Neut's theory of matrix-geometric invariant probability vector. In both [6] and [7], it is not obvious how to obtain marginal probabilities. Note that recursive computation of joint probabilities does not yield marginal probabilities, since (in general) infinite summations should be taken on the joint probability distribution to obtain the marginals, while recursive computations yield only finite number of terms (as large as one wants, but not infinite).

The derivation of the recursive formulas in this paper is completely different than that of [6] and [7]. It is based only on the general structure of the underlying generating function. Consequently, it is simply applied to both the marginal and the joint probabilities. Generally, recursive computations have many advantages over other numerical computation techniques. They are easy to program and their accuracy depends only on the accuracy of computing the basic built in functions (such as additions, multiplications, square roots etc.). For a further discussion on accuracy checks, see [7].

In the following, we first give a very short review of standard (non-recursive) techniques to obtain the corresponding probabilities from a generating function. Then we describe the basic idea of the recursive approach.

Let N be a random variable that represents the number of customers in a queueing system in steady-state. Let $g_n = \Pr(N=n)$, $n = 0, 1, 2, \ldots$, and $G(Z)$ be the generating function of N,

$$G(Z) = \sum_{n=0}^{\infty} g_n Z^n, \quad |Z| < 1, \tag{1}$$

and $G(Z)$ is analytic in the disk $|Z| < 1$. For a given $G(Z)$, one can obtain the occupancy probabilities (at least in principal) by $g_n = G^{(n)}(0)/n!$ $(n = 0, 1, \ldots)$ where $G^{(n)}(0) = \mathrm{d}^n G(Z)/\mathrm{d}Z^n \mid_{Z=0}$. In most cases, the computation of successive differentiations, would be quite cumbersome (the reader is invited to try this approach for the relatively simple generating function in (10)). To alleviate this difficulty, one can use the Cauchy integral formula

$$G^{(n)}(0) = \frac{n!}{2\pi \mathrm{i}} \int_{\Gamma} \frac{G(t)}{t^{n+1}}\, \mathrm{d}t, \quad n = 0, 1, 2, \ldots ,$$

where $\Gamma = \gamma \mathrm{e}^{\mathrm{i}\tau}$, $0 < \gamma < 1$, $0 \leqslant \tau \leqslant 2\pi$. In general, the computation above requires numerical integration and therefore the accuracy of this computation depends on the exact integration method and its parameters (mainly the granularity of the discrete division). A better way to approximate g_n, $n = 0, 1, 2, \ldots$, to some desired accuracy that does not require numerical integration, has been introduced by Jagerman [3]. Yet, both these methods, as well as methods that are based on inversion techniques for Laplace transforms [4], yield only approximate results since they use approximation sequence or truncation of infinite sums.

Exact recursive computation

The basic idea of the recursive approach is the following (see Gross and Harris [2, p.262]). Suppose that $G(Z)$ is given by the quotient of two functions $A(Z)$ and $B(Z)$,

$$G(Z) = A(Z)/B(Z), \tag{2}$$

and assume that the coefficients a_i, b_i, $i = 1, 2, \ldots, n$, of the Taylor expansion are known for both $A(Z)$ and $B(Z)$, where

$$A(Z) = \sum_{i=0}^{\infty} a_i Z^i, \qquad B(Z) = \sum_{i=0}^{\infty} b_i Z^i. \tag{3}$$

Then if $b_0 > 0$ it is easy to see that g_n, $n = 0, 1, 2, \ldots$, are computed via the following recursion:

$$g_0 = a_0/b_0, \qquad g_n = \left(a_n - \sum_{i=0}^{n-1} b_{n-i} g_i \right) / b_0, \quad n > 0. \tag{4}$$

Therefore, in order to compute g_n for some n, we only need to be able to compute a_i, b_i, $i = 0, 1, \ldots, n$, namely to determine the first $n+1$ coefficients of the Taylor series of $A(Z)$ and $B(Z)$ in (3). In Sections 2 and 3 we show how to compute these coefficients in a recursive manner for priority queueing systems.

Note that in some examples $A(Z)$ and similarly $B(Z)$ might be expressed as a product of several terms, namely $A(Z) = \prod_{k=1}^{K} A_k(Z)$. In such cases it is sufficient to determine the first $n+1$ coefficients, $(a_0^k, a_1^k, \ldots, a_n^k)$ of the Taylor series of each $A_k(Z)$. Then the coefficients of $A(Z)$ are obtained by the convolution

$$a_i = \sum_{L_i} \prod_{k=1}^{K} a_{i_k}^k, \quad 0 \leqslant i \leqslant n \quad \text{with } L_i = \left\{ (i_1, i_2, \ldots, i_K) \colon i_k \geqslant 0 \text{ for } k = 1, 2, \ldots, K; \ \sum_{k=1}^{K} i_k = i \right\}. \tag{5}$$

The same derivation applies to multi-dimensional generating functions. We demonstrate this for a two-dimensional generating function. Assume that $G(X, Y)$ is a two-dimensional generating function which is given by the quotient of two functions $A(X, Y)$ and $B(X, Y)$. Thus let

$$G(X, Y) = A(X, Y)/B(X, Y), \tag{6}$$

and assume that the coefficients $a_{i,j}$, $b_{i,j}$ $(i = 0, 1, 2, \ldots, n;\ j = 0, 1, 2, \ldots, m)$ of the two-dimensional Taylor expansion are known for both $A(X, Y)$ and $B(X, Y)$:

$$A(X, Y) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} a_{i,j} X^i Y^i, \qquad B(X, Y) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} b_{i,j} X^i Y^j. \tag{7}$$

Then if $b_{0,0} > 0$ it is easy to see that $g_{n,m}$, n, $m = 0, 1, 2, \ldots$, are computed via the following recursion:

$$g_{0,0} = a_{0,0}/b_{0,0}, \qquad g_{n,m} = \left(a_{n,m} - \sum_{\substack{i=0 \\ (i,j) \neq (n,m)}}^{n} \sum_{j=0}^{m} b_{n-i,m-j} g_{i,j} \right) \Big/ b_{0,0} \quad (n + m > 0). \tag{8}$$

Therefore, in order to compute $g_{n,m}$ for some n and m, we only need to be able to compute $a_{i,j}$, $b_{i,j}$ $(i = 0, 1, \ldots, n;\ j = 0, 1, \ldots, m)$, namely to determine the first $(n+1)(m+1)$ coefficients of the two-dimensional Taylor expansion of $A(X, Y)$ and $B(X, Y)$ in (7). Later we will show how to compute these coefficients (again in a recursive manner) for priority queueing systems. Note that the same method can be used to higher dimensions generating functions as well. Again, if $A(X, Y) = \prod_{k=1}^{K} A_k(X, Y)$, we have that

$$a_{i,j} = \sum_{L_{i,j}} \prod_{k=1}^{K} a_{i_k, j_k}^k \quad \text{with } L_{i,j} = \left\{ ((i_1, j_1), (i_2, j_2), \ldots, (i_K, j_K)) \colon i_k \geqslant 0,\ j_k \geqslant 0 \right.$$
$$\left. \text{for } k = 1, 2, \ldots, K; \ \sum_{k=1}^{K} i_k = i; \ \sum_{k=1}^{K} j_k = j \right\}. \tag{9}$$

2. Preemptive priority queueing system

Consider the two-class preemptive priority queueing system with a single server and two classes of customers, each class having Poisson arrivals and exponentially distributed service times. The high priority class (arrival rate λ_1, service rate μ_1) has preemptive head of the line priority over the low priority customers (arrival rate λ_2, service rate μ_2). We are interested in computing the steady state occupancy probabilities of the high and the low priority customers. Denote $\rho_i = \lambda_i/\mu_i$ $(i = 1, 2)$, $p_0 = 1 - \rho_1 - \rho_2$ and $\lambda = \lambda_1 + \lambda_2$. The high priority class behaves as an M/M/1 queueing system and therefore the steady state probability of having n high priority customers in the system is $(1-\rho_1)\rho_1^n$. For the low priority class it is known (Miller [9]) that

$$G(Z)=p_0/(1-\alpha(Z)-\rho_2 Z), \tag{10}$$

where

$$\alpha(Z)=\left(\mu_1+\lambda-\lambda_2 Z-\sqrt{(\mu_1+\lambda-\lambda_2 Z)^2-4\lambda_1\mu_1}\right)/(2\mu_1). \tag{11}$$

The function $G(Z)$ is of the form (2) with

$$A(Z)=p_0, \qquad B(Z)=1-\alpha(Z)-\rho_2 Z. \tag{12}$$

In order to use the recursive method of (4) we have to compute a_i, b_i $(i=0, 1, 2,\ldots, n)$. For the a_i's we have $a_0=p_0$, $a_i=0$ $(i=1, 2,\ldots)$. To compute the b_i's we first expand $\beta(Z)=\sqrt{(\mu_1+\lambda-\lambda_2 Z)^2-4\lambda_1\mu_1}$ as a Taylor series $\beta(Z)=\sum_{i=0}^{\infty}\beta_i Z^i$ and determine β_i $(i=0, 1, 2,\ldots, n)$. We do this again in a recursive manner by using the relation $(\sum_{i=0}^{\infty}\beta_i Z^i)^2=(\mu_1+\lambda-\lambda_2 Z)^2-4\lambda_1\mu_1$. This gives

$$\beta_0=\sqrt{(\mu_1+\lambda)^2-4\lambda_1\mu_1}, \qquad \beta_1=-\lambda_2(\mu_1+\lambda)/\beta_0, \tag{13a}$$

$$\beta_2=\left(\lambda_2^2-\beta_1^2\right)/(2\beta_0), \qquad \beta_i=-\sum_{j=1}^{i-1}\beta_j\beta_{i-j}/(2\beta_0), \quad i=3, 4,\ldots. \tag{13b}$$

Therefore, for the expansion of $\alpha(Z)=\sum_{i=1}^{\infty}\alpha_i Z^i$, we have

$$\alpha_0=(\mu_1+\lambda-\beta_0)/(2\mu_1), \quad \alpha_1=-(\lambda_2+\beta_1)/(2\mu_1), \quad \alpha_i=-\beta_i/(2\mu_1), \quad i=2, 3,\ldots. \tag{14}$$

From (12) we then have that

$$b_0=1-\alpha_0, \qquad b_1=-\alpha_1-\rho_2, \qquad b_i=-\alpha_i, \quad i=2, 3,\ldots, \tag{15}$$

and now we apply recursion (4) to obtain the steady state probabilities of having n low priority customers in the system.

To derive the recursive relations for the joint probability distribution let $g_{n,m}$ be the steady-state probabilities of having n high priority customers and m low priority customers in the system. Then, $G(X, Y)=\sum_{n=0}^{\infty}\sum_{m=0}^{\infty}g_{n,m}X^n Y^m$ is given by (see [9])

$$G(X, Y)=\frac{p_0}{1-\alpha(Y)-\rho_2 Y}\,\frac{1-\alpha(Y)}{1-X\alpha(Y)}, \tag{16}$$

where $\alpha(Y)$ is given in (11). Therefore, we have

$$a_{0,0}=p_0(1-\alpha_0), \quad a_{0,j}=-p_0\alpha_j \ (j\geqslant 1), \quad a_{i,j}=0 \ (i\geqslant 1, \ j\geqslant 0), \tag{17}$$

$$b_{1,j}=-\sum_{l=0}^{j}b_l\alpha_{j-l}, \quad b_{0,j}=b_j \ (j\geqslant 0), \quad b_{i,j}=b_j \ (i\geqslant 2, \ j\geqslant 0), \tag{18}$$

where b_j is given in (15), and now we apply recursion (8).

3. Non-preemptive priority queueing system

In this section we consider the system of Section 2 when preemption is not allowed. Note that unlike [6], we do not assume that $\mu_1=\mu_2$.

3.1. High priority customers

From [9] we find that the generating function of the occupancy probabilities of high priority customers is given by

$$G(Z)=\frac{p_0[\mu_2+\lambda_1(1-Z)]+\lambda_2}{(1-\rho_1 Z)[\mu_2+\lambda_1(1-Z)]}. \tag{19}$$

So in this case,

$$a_0=p_0(\mu_2+\lambda_1)+\lambda_2, \quad a_1=-\lambda_1 p_0, \quad a_i=0 \ (i\geqslant 2), \tag{20}$$

$$b_0=\mu_2+\lambda_1, \quad b_1=-\rho_1(\mu_2+\lambda_1)-\lambda_1, \quad b_2=\lambda_1\rho_1, \quad b_i=0 \ (i\geqslant 3), \tag{21}$$

and now we apply recursion (4) to obtain g_n. Due to the simplicity of this recursion, it follows immediately (see [7]) that

$$g_n=p_0\rho_1^n+\lambda_2\rho_1^n\{1-[\mu_1/(\lambda_1+\mu_2)]^{n+1}\}/(\lambda_1+\mu_2-\mu_1).$$

3.2. Low priority customers

Again from [9] we have that the generating function of the occupancy probabilities of low priority customers is

$$G(Z)=\frac{p_0[\lambda-\lambda_2 Z+\mu_2-\mu_1\alpha(Z)]}{[\lambda_2(1-Z)+\mu_2][1-\alpha(Z)-\rho_2 Z]}. \tag{22}$$

In this case,

$$a_0=p_0(\lambda+\mu_2-\mu_1\alpha_0), \quad \alpha_1=-p_0(\lambda_2+\mu_1\alpha_1), \quad a_i=-p_0\mu_1\alpha_i \ (i\geqslant 2), \tag{23}$$

$$b_0^1=\lambda_2+\mu_2, \quad b_1^1=-\lambda_2, \quad b_i^1=0 \ (i\geqslant 2), \tag{24}$$

$$b_0^2=1-\alpha_0, \quad b_1^2=-\alpha_1-\rho_2, \quad b_i^2=-\alpha_i \ (i\geqslant 2). \tag{25}$$

Following (7), $b_i=\sum_{l=0}^{i} b_l^1 b_{i-l}^2$ $(i\geqslant 0)$ and we apply recursion (4) to obtain g_n.

3.3. Joint probability distribution

From [9] we have

$$G(X, Y)=G_1(X, Y)+G_2(X, Y), \tag{26}$$

where

$$G_1(X, Y)=\frac{A_1}{B_1}=\frac{A_{11}A_{12}}{B_{11}B_{12}B_{13}}, \qquad G_2(X, Y)=\frac{A_2}{B_2}, \tag{27}$$

with

$$A_{11}=\rho_2 p_0[\lambda+\mu_2-\lambda_2 Y-\mu_1\alpha(Y)], \qquad A_{12}=\mu_1 Y(X-1)-\mu_2 X(Y-1), \tag{28}$$

$$B_{11}=\lambda_1(1-X)+\lambda_2(1-Y)+\mu_2, \qquad B_{12}=\lambda_1 X(1-X)+\lambda_2 X(1-Y)+\mu_1(X-1), \tag{29a}$$

$$B_{13}=1-\alpha(Y)-\rho_2 Y, \tag{29b}$$

$$A_2=\mu_1 p_0(X-1), \qquad B_2=B_{12}. \tag{30}$$

Using the Taylor expansion of $\alpha(Y)$ from (14), it is straightforward to compute the Taylor expansions of A_{11} and B_{13} up to any arbitrary number. Using (9), one recursively computes the tailor expansion of the

Table 1
Non-preemptive priority

$\lambda_1 = 0.8$, $\lambda_2 = 0.2$, $\mu_1 = 1$, $\mu_2 = 2$			
n	Probability for n high priority	Probability for n low priority	Probability for n total
0	1.714 E −01	2.487 E −01	1.000 E −01
1	1.575 E −01	1.140 E −01	8.175 E −02
2	1.318 E −01	7.643 E −02	6.897 E −02
3	1.071 E −01	5.929 E −02	5.986 E −02
4	8.620 E −02	4.884 E −02	5.298 E −02
5	6.910 E −02	4.151 E −02	4.751 E −02
6	5.531 E −02	3.598 E −02	4.296 E −02
7	4.426 E −02	3.160 E −02	3.908 E −02
8	3.541 E −02	2.804 E −02	3.571 E −02
9	2.833 E −02	2.507 E −02	3.275 E −02
10	2.266 E −02	2.255 E −02	3.010 E −02
11	1.813 E −02	2.038 E −02	2.774 E −02
12	1.450 E −02	1.849 E −02	2.560 E −02
13	1.160 E −02	1.683 E −02	2.367 E −02
14	9.284 E −03	1.537 E −02	2.192 E −02
15	7.427 E −03	1.407 E −02	2.032 E −02

product terms A_1 and B_1 of (27). Using (8), it is easy to recursively compute the Taylor series coefficients of the quotients $G_1(X, Y)$ and $G_2(X, Y)$ of (26) and then the coefficients of $G(X, Y)$

3.4. Numerical examples

Numerical examples that are derived via the recursions developed in this section, are given in Tables 1–3. Table 1 corresponds to high load of high priority customers and low load of low priority customers. Table 2 corresponds to high load of low priority customers and low load of high priority customers. Table 3 corresponds to the same load for the two types of customers. The first two columns in each table

Table 2
Non-preemptive priority

$\lambda = 0.1$, $\lambda_2 = 1.6$, $\mu_1 = 1$, $\mu_2 = 2$			
n	Probability for n high priority	Probability for n low priority	Probability for n total
0	8.619 E −01	1.057 E −01	1.000 E −01
1	1.224 E −01	9.051 E −02	8.772 E −02
2	1.397 E −02	7.860 E −02	7.743 E −02
3	1.479 E −03	6.910 E −02	6.882 E −02
4	1.518 E −04	6.134 E −02	6.155 E −02
5	1.537 E −05	5.486 E −02	5.532 E −02
6	1.546 E −06	4.932 E −02	4.990 E −02
7	1.550 E −07	4.451 E −02	4.513 E −02
8	1.552 E −08	4.028 E −02	4.090 E −02
9	1.553 E −09	3.652 E −02	3.712 E −02
10	1.554 E −10	3.315 E −02	3.372 E −02
11	1.554 E −11	3.012 E −02	3.065 E −02
12	1.554 E −12	2.738 E −02	2.787 E −02
13	1.554 E −13	2.490 E −02	2.536 E −02
14	1.554 E −14	2.266 E −02	2.308 E −02
15	1.554 E −15	2.062 E −02	2.101 E −02

Table 3
Non-preemptive priority

$\lambda_1 = 0.4$, $\lambda_2 = 0.8$, $\mu_1 = 1$, $\mu_2 = 2$			
n	Probability for n high priority	Probability for n low priority	Probability for n total
0	5.333 E −01	2.678 E −01	2.000 E −01
1	2.688 E −01	1.608 E −01	1.462 E −01
2	1.168 E −01	1.082 E −01	1.108 E −01
3	4.826 E −02	7.949 E −02	8.693 E −02
4	1.956 E −02	6.176 E −02	7.005 E −02
5	7.868 E −03	4.966 E −02	5.754 E −02
6	3.154 E −03	4.078 E −02	4.790 E −02
7	1.263 E −03	3.393 E −02	4.024 E −02
8	5.054 E −04	2.848 E −02	3.403 E −02
9	2.022 E −04	2.405 E −02	2.891 E −02
10	8.088 E −05	2.041 E −02	2.465 E −02
11	3.235 E −05	1.738 E −02	2.107 E −02
12	1.294 E −05	1.484 E −02	1.806 E −02
13	5.176 E −06	1.270 E −02	1.550 E −02
14	2.070 E −06	1.089 E −02	1.332 E −02
15	8.283 E −07	9.358 E −03	1.147 E −02

corresponds to marginal probabilities of high and low priority customers. The third column in the tables is obtained from the joint probability distribution by computing $\Sigma_{m=0}^{n} g_{m,n-m}$. Notice that the recursions developed in this paper for the joint probability distributions are similar to those in [6] and hence the computational effort is similar. Miller [6] also discusses ways of monitoring the accuracy of the numerical results. In our examples we did not face any numerical instabilities.

4. Other systems

The method described in this paper is easily generalized to handle multi-server priority systems (Gail et al. [1], Miller [8]). The basic difference is that in a multi-server system, the probability distribution is expressed in terms of a vector of generating functions of the form (see [1]):

$$\boldsymbol{G}(Z) = \boldsymbol{B}^{-1}(Z)\boldsymbol{A}(Z), \tag{31}$$

where

$$\boldsymbol{G}(Z) = \left(G^1(Z),\ G^2(Z),\ \ldots,\ G^K(Z)\right)^{\mathrm{T}}, \qquad \boldsymbol{A}(Z) = \left(A^1(Z),\ A^2(Z),\ \ldots,\ A^K(Z)\right)^{\mathrm{T}},$$

$$\boldsymbol{B}(Z) = \begin{pmatrix} B^{11}(Z) & B^{12}(Z) & \cdots & B^{1K}(Z) \\ B^{21}(Z) & B^{22}(Z) & \cdots & B^{2K}(Z) \\ \vdots & & & \\ B^{K1}(Z) & B^{K2}(Z) & \cdots & B^{KK}(Z) \end{pmatrix}.$$

We are interested in computing the coefficients g_i^k $(0 \leqslant i \leqslant n)$ of $G^k(Z)$ for $1 \leqslant k \leqslant K$. The expansion of each element in the vector $\boldsymbol{A}(Z)$ (namely, the coefficients a_i^m for $1 \leqslant m \leqslant K$, $0 \leqslant i \leqslant n$) and each element in the matrix $\boldsymbol{B}(Z)$ (namely, the coefficients b_i^{km} for $1 \leqslant k \leqslant K$, $1 \leqslant m \leqslant K$, $0 \leqslant i \leqslant n$) is done in the same way that we expanded $A(Z)$ and $B(Z)$ in (12) or in (22)–(25). Once these coefficients are known, the coefficients g_i^k $(1 \leqslant k \leqslant K;\ 0 \leqslant i \leqslant n)$ are computed recursively, assuming that the matrix $\boldsymbol{b}_0$ defined below is invertible:

$$\boldsymbol{g}_0 = \boldsymbol{b}_0^{-1}\boldsymbol{a}_0, \qquad \boldsymbol{g}_i = \boldsymbol{b}_0^{-1}\left[\boldsymbol{a}_i - \sum_{l=0}^{i-1} \boldsymbol{b}_{i-l}\boldsymbol{g}_l\right] \quad (i \geqslant 1), \tag{32}$$

where

$$\boldsymbol{g}_i = \left(g_i^1,\ g_i^2,\ \dots\ ,\ g_i^K\right)^{\mathrm{T}}, \qquad \boldsymbol{a}_i = \left(a_i^1,\ a_i^2,\ \dots\ ,\ a_i^K\right)^{\mathrm{T}},$$

$$\boldsymbol{b}_i = \begin{pmatrix} b_i^{11} & b_i^{12} & \cdots & b_i^{1K} \\ b_i^{21} & b_i^{22} & \cdots & b_i^{2K} \\ \vdots & & & \\ b_i^{K1} & b_i^{K2} & \cdots & b_i^{KK} \end{pmatrix}.$$

We also remark that our method is not limited to priority queueing systems. We have applied this method in other systems such as the discrete-time tandem in Morrison [10], the two interfering queues in Sidi and Segall [11] and M/G/1 systems in which the Laplace transform of the service time is some rational function.

Acknowledgement

The authors thank Dr. Inder Gopal for helpful discussions which lead to the formulation of the problem.

References

[1] H.R. Gail, S.L. Hantler and B.A. Taylor, "Analysis of a preemptive priority multiserver queue", *Proceedings of the third International Conference on Data Communication Systems and Their Performance*, Rio de Janeiro, June 1987, 341–356.
[2] D. Gross and C.M. Harris, *Fundamentals of Queueing Theory* (Second Edition), Wiley, New York, 1985.
[3] D.L. Jagerman, "Obtaining coefficients of a power series", Unpublished note, AT&T Bell Labs., Holmdel.
[4] D.L. Jagerman, "An inversion technique for the Laplace transform", *Bell System Tech. J.* **61**, 1995–2002 (1982).
[5] N.K. Jaiswal, *Priority Queues*, Academic Press, New York, 1968.
[6] B.I. Marks, "State probabilities of M/M/1 priority queues", *Oper. Res.* **21**, 974–987 (1973).
[7] D.G. Miller, "Computation of steady-state probabilities for M/M/1 priority queues", *Oper. Res.* **29**, 945–958 (1981).
[8] D.G. Miller, "Steady-state algorithm analysis of M/M/c two-priority queues with heterogeneous rates", *Applied Probability – Computer Science: The Interface*, R. Disney and T. Ott (eds.), 1982, 207–225.
[9] R.G. Miller, "Priority queues", *Ann. Math. Statist.* **31**, 86–103 (1960).
[10] J.A. Morrison, "Two discrete-time queues in tandem", *IEEE Trans. Comm.* COM-**27**, 563–573 (1979).
[11] M. Sidi and A. Segall, "Two interfering queues in packet-radio networks", *IEEE Trans. Comm.* COM-**31**, 123–129 (1983).
[12] H. White and L. Christie, "Queueing with preemptive priorities or with breakdown", *Oper. Res.* **6**, 79–95 (1958).