Hindawi Publishing Corporation
Advances in Operations Research
Volume 2016, Article ID 3478709, 8 pages
http://dx.doi.org/10.1155/2016/3478709

Research Article

Selective Trunk with Multiserver Reservation

Bacel Maddah[1] and Muhammad El-Taha[2]

[1] *Department of Industrial Engineering and Management, Faculty of Engineering and Architecture, American University of Beirut, Beirut 1107 2020, Lebanon*
[2] *Department of Mathematics and Statistics, University of Southern Maine, 96 Falmouth Street, Portland, ME 04104-9300, USA*

Correspondence should be addressed to Bacel Maddah; bacel.maddah@aub.edu.lb

Received 3 November 2015; Revised 28 January 2016; Accepted 29 March 2016

Academic Editor: Yiqiang Zhao

We consider a queueing model that is primarily applicable to traffic control in communication networks that use the Selective Trunk Reservation technique. Specifically, consider two traffic streams competing for service at an n-server queueing system. Jobs from the *protected* stream, stream 1, are blocked only if all n servers are busy. Jobs from the *best effort* stream, stream 2, are blocked if $n-r,\ r \geq 1$, servers are busy. Blocked jobs are diverted to a secondary group of $c-n$ servers with, possibly, a different service rate. We extend the literature that studied this system for the special case of $r = 1$ and present an explicit computational scheme to calculate the joint probabilities of the number of primary and secondary busy servers and related performance measures. We also argue that the model can be useful for bed allocation in a hospital.

1. Introduction

In this paper, we consider a simple and effective traffic control technique for communication networks with two priority traffic classes. The method, referred to as Selective Trunk Reservation (STR), assumes two traffic streams competing for service at an n-server queueing system. Jobs from stream 1, the *protected* stream, are blocked if all n servers are busy. Jobs from stream 2, the *best effort* stream, are blocked if $n-r,\ r \geq 1$, servers are busy. Blocked jobs are sent to a secondary group of $c-n$ servers with, possibly, a different service rate from that of the primary servers. Jobs from both streams are lost when all c servers are busy.

El-Taha and Heath [1] consider a similar problem but they restrict attention to the case $r = 1$ and describe a procedure to evaluate joint probabilities for $r > 1$. Their paper includes significant enhancements of earlier work (e.g., [1–7]). This paper is an extension of El-Taha and Heath [1] work and a response to inquiries about implementing the STR method of [1] for all $r \geq 1$. In particular, we explicitly derive the joint probabilities for all $r \geq 1$ and provide an efficient algorithm for computing them.

The block diagram of the model in this paper is shown in Figure 1. We derive the joint probabilities of the number of primary and secondary busy servers. From these probabilities, performance measures, including overflow probabilities of the server groups, are derived. We then apply our results to the specific case of two Poisson traffic streams and two heterogeneous groups of exponential servers.

One alternative application of the model in this paper, which came to our attention recently, is for bed allocation in a hospital ward. The literature contains several examples of bed allocation with interesting similarities to the classic STR problem of communication. Examples of these works include [8–13]. In many of these works (e.g., [8, 10]), two types of patients are considered, serious and nonserious, which arrive according to a Poisson process and have an exponentially distributed duration of stay (service time), similar to the two streams of arrivals in the STR problem. More interestingly, this literature discusses the setting of a "cut-off occupancy" in terms of a number of beds in the ward reserved for serious patients only, which is in the same spirit of STR. Esogbue and Singh [10] further consider "buffer accommodation" which mimics the secondary servers we

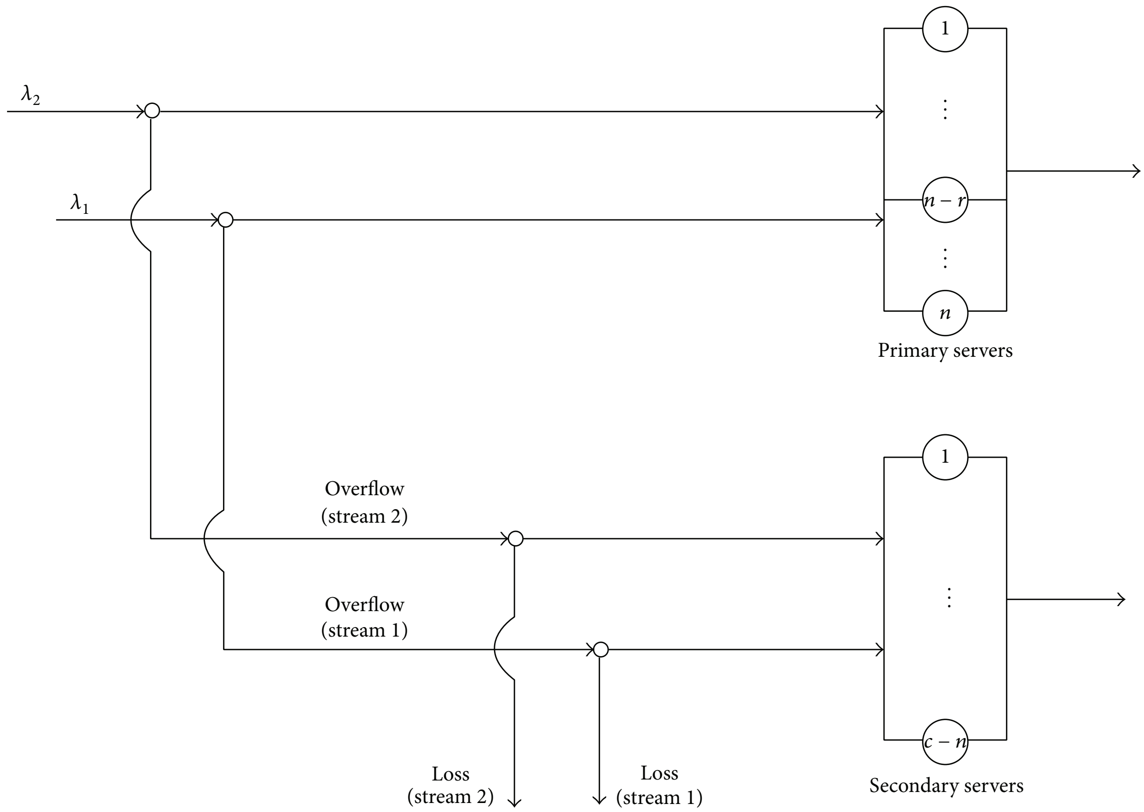

Figure 1: Block diagram of the STR model.

consider in this paper. However, one distinctive difference between STR in communication and bed allocation in hospitals is that, in the latter, the service times depend on the type of arrival. This require state expansion, to track patient type, in Markovian STR models, such as the one in this paper. We believe that our efficient computational schemes can be tailored for such expanded-state bed allocation problems.

The paper is organized as follows. In Section 2, we present an efficient iterative method to find the joint probability distribution of the number of busy primary and secondary servers. In Section 3, we provide an application that considers two Poisson streams, based on the analysis of Section 2, and we illustrate the application of the method by a numerical example. In Section 4, we draw some conclusions and discuss possible extensions. Finally, in the Appendix we provide an algorithm that is used for the development of a computer program of the present model.

2. Joint Stationary Probabilities

Formally, we describe the above model as a two-dimensional stochastic process $\mathbf{X} \equiv \{(X_1(t), X_2(t));\ t \geq 0\}$ with finite state space $\mathcal{S}$ such that $X_1(t)$ takes integer values in $\{0, \dots, n\}$, $n \geq 1$; $X_2(t)$ takes integer values in $\{0, \dots, c-n\}$, $c \geq n+1$. Here, $X_1(t)$ represents the number of primary busy servers and $X_2(t)$ represents the number of secondary busy servers.

Utilizing a similar notation to El-Taha and Heath [1], define $\beta_1(i, j)$ to be the state (i, j) stream 1 arrival rate and $\beta_2(i, j)$ to be the stream 2 arrival rate when the system state is (i, j). Let $\beta(i, j) = \beta_1(i, j) + \beta_2(i, j)$. Define $\gamma_1(i)$ to be the primary group service rate when i primary servers are busy and $\gamma_2(j)$ to be the secondary group service rate when j secondary servers are busy. Let $p_{i,j}$ denote the state (i, j) joint stationary probability, $i = 0, \dots, n$, $j = 0, \dots, c-n$. The transition rate out of state (i, j) is denoted by $\alpha(i, j) = \beta(i, j) + \gamma_1(i) + \gamma_2(j)$. Then, the state probabilities for any $r \geq 1$ satisfy the global balance equations:

$$\begin{aligned} \alpha(i, j)\, p_{i,j} &= \beta(i-1, j)\, p_{i-1,j} + \gamma_1(i+1)\, p_{i+1,j} \\ &\quad + \gamma_2(j+1)\, p_{i,j+1}, \\ &\qquad i = 0, 1, \dots, n-r-1;\ j = c-n, \dots, 0, \end{aligned} \tag{1}$$

$$\begin{aligned} \alpha(n-r, j)\, p_{n-r,j} &= \beta(n-r-1, j)\, p_{n-r-1,j} \\ &\quad + \beta_2(n-r, j-1)\, p_{n-r,j-1} \\ &\quad + \gamma_1(n-r+1)\, p_{n-r+1,j} \\ &\quad + \gamma_2(j+1)\, p_{n-r,j+1}; \\ &\qquad j = c-n, \dots, 0, \end{aligned} \tag{2}$$

$$\begin{aligned} \alpha(i,j)\,p_{i,j} = {} & \beta_1(i-1,j)\,p_{i-1,j} \\ & + \beta_2(i,j-1)\,p_{i,j-1} \\ & + \gamma_1(i+1)\,p_{i+1,j} \\ & + \gamma_2(j+1)\,p_{i,j+1}; \\ & i = n-r+1,\ldots,n-1;\; j = c-n,\ldots,0, \end{aligned} \tag{3}$$

$$\begin{aligned} \alpha(n,j)\,p_{n,j} = {} & \beta_1(n-1,j)\,p_{n-1,j} \\ & + \beta(n,j-1)\,p_{n,j-1} \\ & + \gamma_2(j+1)\,p_{n,j+1}; \\ & j = c-n,\ldots,0. \end{aligned} \tag{4}$$

The normalization equation is $\sum_{i=0}^{n}\sum_{j=0}^{c-n} p_{i,j} = 1$. The relations in (1)–(4) are graphically illustrated in Figure 2. Equations (1)–(4) expand similar flow balance equations in El-Taha and Heath [1] by allowing for $r > 1$ in (3). Equations (1), (2), and (4) are the same as those in [1]. We present these equations here for completeness.

The remainder of this section is devoted to solving (1)–(4) efficiently via an iterative scheme. The first step, in (5), expresses each $p_{i,j}$, $i = 1,\ldots,n-r$, $j = 0,\ldots,c-n$, in terms of the probabilities $p_{0,k}$, $k = j,\ldots,c-n$:

$$p_{i,j} = \sum_{k=0}^{\min\{i,c-n-j\}} (-1)^k G(i,j,k)\,p_{0,j+k}, \qquad i = 1,\ldots,n-r,\; j = 0,\ldots,c-n, \tag{5}$$

where, for each $j = c-n,\ldots,0$,

$$G(i,j,k) = \begin{cases} 1 & i=0,\; k=0; \\ \gamma_1(i)^{-1}\left[\alpha(i-1,j)\,G(i-1,j,k) - \beta(i-2,j)\,G(i-2,j,k) + \gamma_2(j+1)\,G(i-1,j+1,k-1)\right], & k = 0,\ldots,\min\{i,c-n-j\},\; i = 1,\ldots,n-r; \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

Equation (5) can be proved in a similar manner to Lemma 2.1 of El-Taha and Heath [1]. In our main result, Theorem 1, (5) is utilized in obtaining $\{p_{i,j},\ i = 0,\ldots,n-r;\ j = 0,\ldots,c-n\}$, in terms of $p_{n-k,c-n}$, $k = 0,1,\ldots,r$. The $p_{n-k,c-n}$ probabilities are obtained by solving a system of $r+1$ linear equations formulated by applying Theorem 1 in combination with the limiting state probabilities of the number of busy primary servers. Finally, all joint probabilities are computed by appealing again to Theorem 1 and by replacing $p_{n-k,c-n}$, $k = 0,1,\ldots,r$, by their computed values.

Theorem 1 is our main contribution with respect to El-Taha and Heath [1], as it gives explicit expressions of the limiting probabilities $p_{i,j}$ in terms of a smaller set of these probabilities, $p_{n-k,c-n}$, $k = 0,1,\ldots,r$. The main result in [1] is a special case of Theorem 1 with $r = 1$.

Theorem 1. *For all (i,j), $i = 0,\ldots,n$ and $j = 0,\ldots,c-n$,*

$$p_{i,j} = \sum_{m=0}^{r} B_m(i,j)\,p_{n-r+m,c-n}, \tag{7}$$

where

$$B_m(i,j) = \sum_{k=0}^{\min(i,c-n-j)} (-1)^k G(i,j,k)\,B_m(0,j+k), \qquad m = 0,\ldots,r;\; i = 1,\ldots,n-r-1;\; r \le n-2;\; j = c-n-1,\ldots,0; \tag{8}$$

and the "boundary" conditions are

$$B_m(0,j) = G(n-r,j,0)^{-1}\left[B_m(n-r,j) - \sum_{k=1}^{\min(n-r,c-n-j)} (-1)^k G(n-r,j,k)\,B_m(0,j+k)\right], \qquad m = 0,\ldots,r;\; j = c-n-1,\ldots,0, \tag{9}$$

$$\begin{aligned} B_m(n-r,j) = {} & \beta_2(n-r,j)^{-1} \\ & \cdot\left[\alpha(n-r,j+1)\,B_m(n-r,j+1)\right. \\ & - \beta(n-r-1,j+1)\,B_m(n-r-1,j+1) \\ & - \gamma_1(n-r+1)\,B_m(n-r+1,j+1) \\ & \left. - \gamma_2(j+2)\,B_m(n-r,j+2)\right], \\ & m = 0,\ldots,r;\; j = c-n-1,\ldots,0, \end{aligned} \tag{10}$$

$$\begin{aligned} B_m(i,j) = {} & \beta_2(i,j)^{-1}\left[\alpha(i,j+1)\,B_m(i,j+1)\right. \\ & - \beta_1(i-1,j+1)\,B_m(i-1,j+1) \\ & \left. - \gamma_1(i+1)\,B_m(i+1,j+1) - \gamma_2(j+2)\,B_m(i,j+2)\right], \\ & m = 0,\ldots,r;\; i = n-r+1,\ldots,n-1;\; j = c-n-1,\ldots,0, \end{aligned} \tag{11}$$

$$\begin{aligned} B_m(n,j) = {} & \beta(n,j)^{-1}\left[\alpha(n,j+1)\,B_m(n,j+1)\right. \\ & - \beta_1(n-1,j+1)\,B_m(n-1,j+1) \\ & \left. - \gamma_2(j+2)\,B_m(n,j+2)\right], \\ & m = 0,\ldots,r;\; j = c-n-1,\ldots,0, \end{aligned} \tag{12}$$

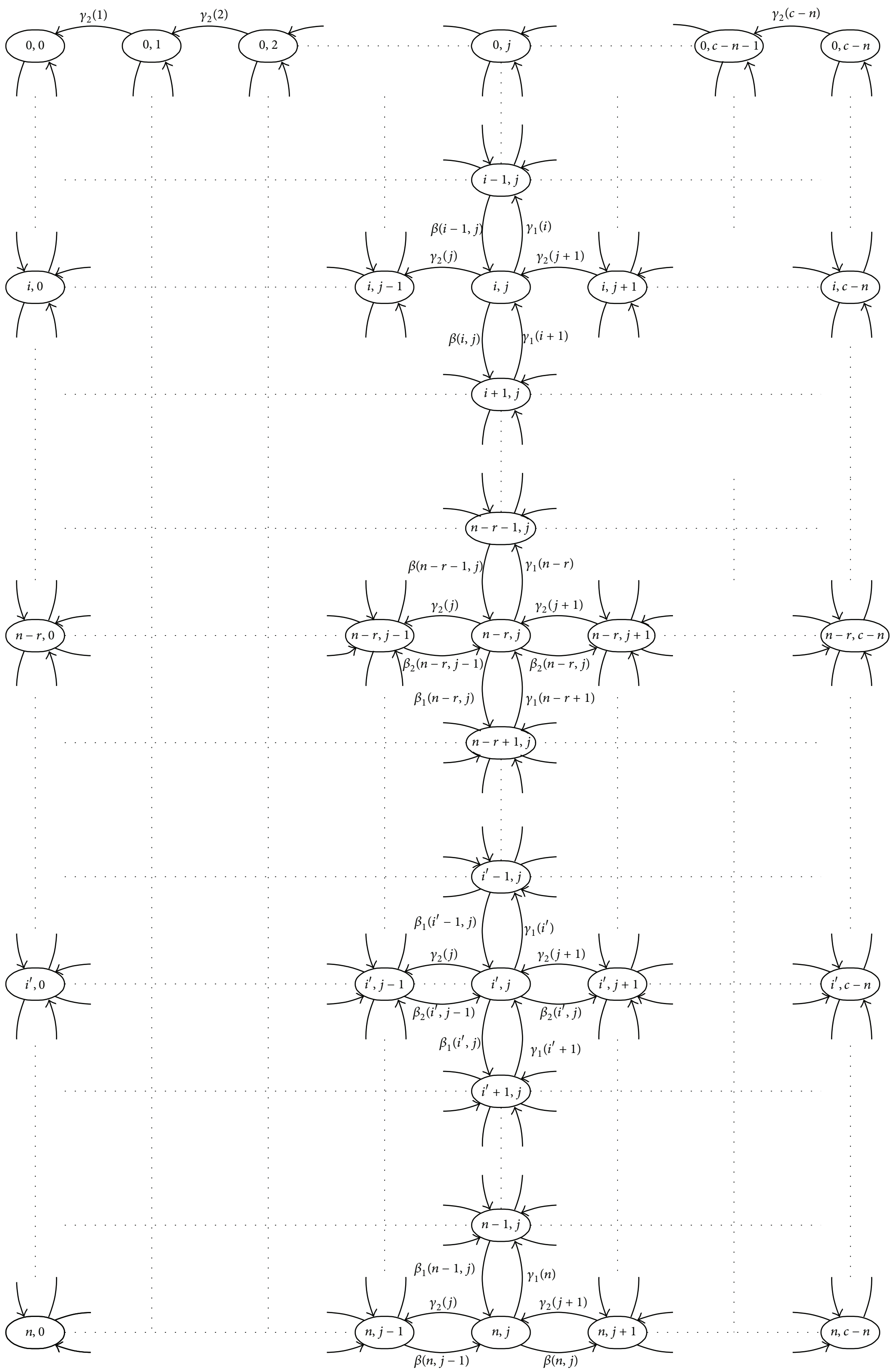

FIGURE 2: Transition diagram of the model.

$$B_m(i,c-n) = \begin{cases} \dfrac{G(i,c-n,0)}{G(n-r,c-n,0)}; & m=0;\ i=0,\ldots,n-r \\ 1; & i=n-r+m;\ m=1,\ldots,r; \\ 0; & \textit{otherwise}. \end{cases} \quad (13)$$

Proof. The joint probabilities are derived iteratively. First consider $j=c-n$. By (5),

$$p_{i,c-n} = G(i,c-n,0)\,p_{0,c-n}; \quad i=0,1,\ldots,n-r, \quad (14)$$

which implies

$$p_{n-r,c-n} = G(n-r,c-n,0)\,p_{0,c-n}, \quad (15)$$

and then

$$p_{0,c-n} = G(n-r,c-n,0)^{-1}\,p_{n-r,c-n}. \quad (16)$$

It follows that

$$\begin{aligned} p_{i,c-n} &= \frac{G(i,c-n,0)}{G(n-r,c-n,0)}p_{n-r,c-n} \\ &= B_0(i,c-n)\,p_{n-r,c-n}, \quad i=0,\ldots,n-r, \end{aligned} \quad (17)$$

where $B_0(i,c-n)$ is given by (13). Note that, for $i=n-r+1,\ldots,n$ and all $m=0,\ldots,r$, $B_m(i,c-n)=0$ except $B_m(n-r+m,c-n)=1$. Therefore, the theorem is valid for $j=c-n$.

Next, we consider the general case. Rewrite (3) as

$$\begin{aligned} p_{i,j-1} &= \beta_2(i,j-1)^{-1}\left[\alpha(i,j)\,p_{i,j} - \beta_1(i-1,j)\,p_{i-1,j}\right. \\ &\quad \left. - \gamma_1(i+1)\,p_{i+1,j} - \gamma_2(j+1)\,p_{i,j+1}\right]. \end{aligned} \quad (18)$$

Using a change of variable $(j-1\to j)$, we obtain

$$\begin{aligned} p_{i,j} &= \beta_2(i,j)^{-1}\left[\alpha(i,j+1)\,p_{i,j+1}\right. \\ &\quad - \beta_1(i-1,j+1)\,p_{i-1,j+1} - \gamma_1(i+1)\,p_{i+1,j+1} \\ &\quad \left. - \gamma_2(j+2)\,p_{i,j+2}\right], \end{aligned} \quad (19)$$

where $i=n-r+1,\ldots,n-1$; and $j=c-n-1,\ldots,0$.

Now, suppose that the theorem holds for $j+1, j+2,\ldots,c-n$. Then for $i=n-r+1,\ldots,n-1$; and $j=c-n-1,\ldots,0$

$$\begin{aligned} p_{i,j} &= \beta_2(i,j)^{-1} \\ &\quad \cdot \left\{\alpha(i,j+1)\sum_{m=0}^{r}B_m(i,j+1)\,p_{n-r+m,c-n}\right. \\ &\quad - \beta_1(i-1,j+1)\sum_{m=0}^{r}B_m(i-1,j+1)\,p_{n-r+m,c-n} \\ &\quad - \gamma_1(i+1)\sum_{m=0}^{r}B_m(i+1,j+1)\,p_{n-r+m,c-n} \\ &\quad \left. - \gamma_2(j+2)\sum_{m=0}^{r}B_m(i,j+2)\,p_{n-r+m,c-n}\right\}, \end{aligned} \quad (20)$$

which leads to

$$\begin{aligned} p_{i,j} &= \beta_2(i,j)^{-1}\left\{\sum_{m=0}^{r}\left[\alpha(i,j+1)\,B_m(i,j+1)\right.\right. \\ &\quad - \beta_1(i-1,j+1)\,B_m(i-1,j+1) \\ &\quad - \gamma_1(i+1)\,B_m(i+1,j+1) \\ &\quad \left.\left. - \gamma_2(j+2)\,B_m(i,j+2)\right]p_{n-r+m,c-n}\right\} = \sum_{m=0}^{r}B_m(i, \\ &\quad j)\,p_{n-r+m,c-n}, \end{aligned} \quad (21)$$

where $B_m(i,j)$ is given by (11).

Similarly, by manipulating (2) and (4) one can show that $p_{n-r,j} = \sum_{m=0}^{r}B_m(n-r,j)p_{n-r+m,c-n}$, $j=0,\ldots,c-n$, where $B_m(n-r,j)$ is given by (10) and that $p_{n,j} = \sum_{m=0}^{r}B_m(n,j)p_{n-r+m,c-n}$, $j=0,\ldots,c-n$, where $B_m(n,j)$ is given by (12).

Next, we evaluate $p_{0,j}$, $j=c-n-1,\ldots,0$. It follows from the above equations and (5) that

$$\begin{aligned} p_{n-r,j} &= \sum_{m=0}^{r}B_m(n-r,j)\,p_{n-r+m,c-n} \\ &= \sum_{k=0}^{\min(n-r,c-n-j)}(-1)^k G(n-r,j,k)\,p_{0,j+k} \\ &= G(n-r,j,0)\,p_{0,j} \\ &\quad + \sum_{k=1}^{\min(n-r,c-n-j)}(-1)^k G(n-r,j,k)\,p_{0,j+k}. \end{aligned} \quad (22)$$

Solving for $p_{0,j}$, using the induction argument, and simplifying,

$$\begin{aligned} p_{0,j} &= G(n-r,j,0)^{-1}\left[p_{n-r,j} - \sum_{k=1}^{\min(n-r,c-n-j)}(-1)^k\right. \\ &\quad \left.\cdot G(n-r,j,k)\,p_{0,j+k}\right] = G(n-r,j,0)^{-1} \\ &\quad \cdot\left[\sum_{m=0}^{r}B_m(n-r,j)\,p_{n-r+m,c-n}\right. \\ &\quad - \sum_{m=0}^{r}\sum_{k=1}^{\min(n-r,c-n-j)}(-1)^k G(n-r,j,k)\,B_m(0,j+k) \\ &\quad \left.\cdot p_{n-r+m,c-n}\right] = G(n-r,j,0)^{-1}\sum_{m=0}^{r}\left[B_m(n-r,j)\right. \\ &\quad \left. - \sum_{k=1}^{\min(n-r,c-n-j)}(-1)^k G(n-r,j,k)\,B_m(0,j+k)\right] \\ &\quad \cdot p_{n-r+m,c-n} = \sum_{m=0}^{r}B_m(0,j)\,p_{n-r+m,c-n}, \end{aligned} \quad (23)$$

where $B_m(0,j)$ is given by (9).

Now, for $i=1,2,\ldots,n-r-1$, using (5),

$$\begin{aligned} p_{i,j} &= \sum_{k=0}^{\min(i,c-n-j)}(-1)^k G(i,j,k)\,p_{0,j+k} \\ &= \sum_{k=0}^{\min(i,c-n-j)}(-1)^k G(i,j,k) \end{aligned}$$

$$
\begin{aligned}
&\cdot \sum_{m=0}^{r} B_m(0, j+k)\, p_{n-r+m,c-n} \\
&= \sum_{m=0}^{r} \left[\sum_{k=0}^{\min(i,c-n-j)} (-1)^k G(i,j,k)\, B_m(0, j+k) \right] \\
&\cdot p_{n-r+m,c-n} = \sum_{m=0}^{r} B_m(i,j)\, p_{n-r+m,c-n},
\end{aligned} \tag{24}
$$

where $B_m(i,j)$ is given by (8). The induction argument completes the proof of the theorem. □

Now, we proceed to evaluate $p_{n-r+m,c-n}$, $m = 0,\ldots,r$. In the following theorem, these are obtained based on the marginal limiting probabilities of the number of busy primary servers, $p_k^{(n)}$, $k = 0,1,\ldots,n$, which have a simple closed form.

Theorem 2. *The probabilities $p_{n-r+m,c-n}$, $m = 0,\ldots,r$, are the solution to the following system of linear equations:*

$$
\sum_{m=0}^{r} C(m,k)\, p_{n-r+m,c-n} = p_k^{(n)}, \tag{25}
$$

where $p_k^{(n)} = (\prod_{i=1}^{k} \beta(i-1)\gamma_1(i)^{-1})/(1 + \sum_{l=1}^{n}\prod_{i=1}^{l}\beta(i-1)\gamma_1(i)^{-1})$, $C(m,k) = \sum_{j=0}^{c-n} B_m(k,j)$, and $B_m(k,j)$, $k = 0,1,\ldots,n$, $j = 0,1,\ldots,c-n$ are as given in Theorem 1.

Proof. The proof follows by noting that $p_k^{(n)} = \sum_{j=0}^{c-n} p_{k,j}$, $k = 0,1,\ldots,n$. In addition, the probabilities $p_k^{(n)}$ satisfy the flow balance equations:

$$
\beta(k)\, p_k^{(n)} = \gamma_1(k+1)\, p_{k+1}^{(n)}, \quad k = 0,1,\ldots,n. \tag{26}
$$

Utilizing $p_0^{(n)}$ as a normalizing constant and utilizing the expression in Theorem 1 of $p_{k,j}$, $k = 0,1,\ldots,n$, $j = 0,\ldots,c-n$, complete the proof. □

3. Application

Similar to El-Taha and Heath [1], we present an application for the special case of two, state-independent, Poisson arrival streams, a *real time* (protected) stream with arrival rate λ_1 and a *best effort* stream with arrival rate λ_2. Both primary and secondary groups of servers have exponentially distributed service times with constant service rates μ_1 and μ_2, respectively. The transition rates are then obtained from the general model as

$$
\begin{aligned}
\beta(i,j) &= \lambda_1 + \lambda_2, \\
&\qquad i = 0,\ldots,n-r-1,\ i = n;\ j = 0,\ldots,c-n, \\
\beta_1(i,j) &= \lambda_1, \quad i = n-r,\ldots,n-1,\ j = 0,\ldots,c-n, \\
\beta_2(i,j) &= \lambda_2, \\
&\qquad i = n-r,\ldots,n-1\ ,\ j = 0,\ldots,c-n-1, \\
\gamma_1(i) &= i\mu_1, \quad i = 0,\ldots,n, \\
\gamma_2(j) &= j\mu_2, \quad j = 0,\ldots,c-n, \\
\beta(k) &= \begin{cases} \lambda_1 + \lambda_2, & k = 0,\ldots,n-r-1, \\ \lambda_1, & k = n-r,\ldots,n-1. \end{cases}
\end{aligned} \tag{27}
$$

The joint probabilities, $p_{i,j}$, are obtained from Theorem 1 and subsequent results of Section 2. The overflow probabilities, the loss probabilities of the protected and the best effort streams, and the mean number of busy primary and secondary servers are then obtained from the joint probabilities, $p_{i,j}$. For simplicity, one may use the distribution of the number of busy primary servers, $p_i^{(n)}$, as defined in Theorem 2, in obtaining some of these measures of performance.

We have developed a C++ program for obtaining the joint probabilities and other measures of performance. A summary of the program algorithm is presented in the Appendix. To solve the system of linear equations in (25), we use the LU decomposition technique with partial pivoting as presented by Press et al. [14] (pp. 43–48). Press et al. [14] report that this method is three times faster than the classical Gauss-Jordan elimination method. In addition, the method is stable and yield results with high precision for systems of linear equations of dimension in the order of couple of hundreds. Therefore, our computational scheme is expected to be fast and robust.

Example. To illustrate the application of our method, consider an example with $(c,n,r) = (16,10,3)$ and arrival and service rates $(\lambda_1,\lambda_2,\mu_1,\mu_2) = (10.0, 10.0, 3.0, 1.0)$. Joint probabilities, generated by our C++ program, are presented as follows:

$p_{i,j}$; $i = 0,1,\ldots,10$; $j = 0,\ldots,6$

$$
\begin{matrix}
0.000354 & 0.000457 & 0.000392 & 0.000272 & 0.000156 & 0.000070 & 0.000019 \\
0.002208 & 0.002936 & 0.002604 & 0.001875 & 0.001134 & 0.000544 & 0.000161 \\
0.006793 & 0.009354 & 0.008607 & 0.006461 & 0.004130 & 0.002145 & 0.000715 \\
0.013680 & 0.019626 & 0.018835 & 0.014818 & 0.010054 & 0.005703 & 0.002184 \\
0.020103 & 0.030334 & 0.030609 & 0.025395 & 0.018390 & 0.011492 & 0.005178 \\
0.022623 & 0.036486 & 0.039187 & 0.034594 & 0.026899 & 0.018671 & 0.010206 \\
0.019626 & 0.034914 & 0.040776 & 0.038837 & 0.032662 & 0.025321 & 0.017494 \\
0.012305 & 0.026207 & 0.034800 & 0.036657 & 0.033678 & 0.029067 & 0.026934 \\
0.003575 & 0.008741 & 0.012867 & 0.014727 & 0.014597 & 0.013846 & 0.014834 \\
0.000944 & 0.002585 & 0.004160 & 0.005131 & 0.005459 & 0.005659 & 0.006872 \\
0.000201 & 0.000630 & 0.001130 & 0.001528 & 0.001762 & 0.001999 & 0.003019
\end{matrix} \tag{28}
$$

```
INITIALIZE:
    input c, n, r, α(i, j), β_1(i, j), β_2(i, j), β(i, j), γ_1(i), γ_2(j) ∀0 ≤ i ≤ n, ∀0 ≤ j ≤ c − n;
STEP 1. CASE OF (j = c − n)
    G(0, c − n, 0) = 1;
    for (i = 1; i ≤ n − r; i + +)
        G(i, c − n, 0) = γ_1(i)^{−1}[α(i − 1, c − n)G(i − 1, c − n, 0)
            −β(i − 2, c − n)G(i − 2, c − n, 0)];
    for (i = 0; i ≤ n − r; i + +){
        B_0(i, c − n) = G(i, c − n, 0)/G(n − r, c − n, 0);
    for (m = 1; m ≤ r; m + +){
        B_m(n − r + m, c − n) = 1;
    }
    }
STEP 2. CASE OF (j ≤ c − n − 1)
    for (j = c − n − 1, j ≥ 0; j − −){
        G(0, j, 0) = 1;
            for (i = 1; i ≤ n − r; i + +){
        for (k = 0; k ≤ min(i, c − n − j); k + +)
                G(i, j, k) = γ_1(i)^{−1}[α(i − 1, j)G(i − 1, j, k)
                −β(i − 2, j)G(i − 2, j, k) + γ_2(j + 1)G(i − 1, j + 1, k − 1)]
        }
        for (m = 0; m ≤ r; m + +){
        B_m(n, j) = β(n, j)^{−1}[α(n, j + 1)B_m(n, j + 1)
            −β_1(n − 1, j + 1)B_m(n − 1, j + 1) − γ_2(j + 2)B_m(n, j + 2)];
        B_m(n − r, j) = β_2(n − r, j)^{−1}[α(n − r, j + 1)B_m(n − r, j + 1)
            −β(n − r − 1, j + 1)B_m(n − r − 1, j + 1) − γ_1(n − r + 1)B_m(n − r + 1, j + 1)
            −γ_2(j + 2)B_m(n − r, j + 2)];
            for (i = n − r + 1; i ≤ n − 1; i + +){
        B_m(i, j) = β_2(i, j)^{−1}[α(i, j + 1)B_m(i, j + 1)
            −β_1(i − 1, j + 1)B_m(i − 1, j + 1) − γ_1(i + 1)B_m(i + 1, j + 1)
            −γ_2(j + 2)B_m(i, j + 2)];
        }
        B_m(0, j) = G(n − r, j, 0)^{−1}[B_m(n − r, j)
            − Σ_{k=1}^{min(n−r,c−n−j)} (−1)^k G(n − r, j, k)B_m(0, j + k)];
        for (i = 1; i ≤ n − r − 1; i + +)
                B_m(i, j) = Σ_{k=0}^{min(i,c−n−j)} (−1)^k G(i, j, k)B_m(0, j + k)
        }
        }
    }
COMPUTE JOINT PROBABILITIES:
    input β(k), γ_1(k + 1) ∀k < n;
      for (k = n − r; k ≤ n; k + +)
    p_k^{(n)} = Π_{i=1}^{k} β(i − 1)/γ_1(i)[1 + Σ_{k=1}^{n} Π_{i=1}^{k} β(i − 1)/γ_1(i)]^{−1};
    }
Compute: C(m, k) = Σ_{j=0}^{c−n} B_m(k, j)
Solve the system of equations
Σ_{m=0}^{r} C(m, k) p_{n−r+m,c−n} = p_k^{(n)}
(Note; the unknowns are p_{n−r+m,c−n}, m = 0, . . . , r, k = n − r, . . . , n.)
    for (j = c − n; j ≥ 0; j − −)
        for (i = 0; i ≤ n; i + +)
            p_{i,j} = Σ_{m=0}^{r} B_m(i, j) p_{n−r+m,c−n}
END
```

Algorithm 1

Measures of performance are easily obtained from the above joint probabilities or the distribution of the number of busy primary servers, $p_i^{(n)}$. These measures of performance are computed as follows: protected stream overflow probability, $P_1 = p_n^{(n)} = \sum_{j=0}^{c-n} p_{n,j} = 0.010$; best effort stream overflow probability, $P_2 = \sum_{i=n-r}^{n} p_i^{(n)} = \sum_{i=n-r}^{n} \sum_{j=0}^{c-n} p_{i,j} = 0.324$; protected stream loss probability, $Q_1 = p_{n,c-n} = 0.003$; best effort stream loss probability, $Q_2 = \sum_{i=n-r}^{n} p_{i,c-n} = 0.052$; mean number of busy primary servers, $L_1 = \sum_{i=1}^{n} i p_i^{(n)} = \sum_{i=1}^{n} \sum_{j=0}^{c-n} i p_{i,j} = 5.553$; and mean number of busy secondary servers, $L_2 = \sum_{i=0}^{n} \sum_{j=1}^{c-n} j p_{i,j} = 2.795$. These results have been validated via discrete-event simulation.

Note finally that the example here is selected for illustrative purposes only. Our computational scheme can be further utilized in determining an optimal r value that meets a specific criterion such as a specified overflow probability for *protected traffic* and studying the effect of increasing the *best effort* traffic on the *protected traffic* quality of service. For details, see Examples 2 and 3 in El-Taha and Heath [1].

4. Conclusion

We have extended the results of El-Taha and Heath [1] to explicitly allow multiple reserved channels, $r > 1$. As in [1], we present an efficient iterative technique to evaluate the joint probabilities of busy primary and secondary servers. A new component of the present model is the system of $r + 1$ linear equations given in (25). By using effective numerical methods and with the availability of computers with high processing speed and storage capacity, this part of the computational scheme will not be of concern even for large values of r. We have illustrated the application of the method for the case of two Poisson arrival streams competing for service on two sets of heterogeneous servers with constant service rates. It is important to note that the present model may be applicable for more general situations such as arrivals from a finite source and state dependent service rates. This may be an area for further extensions and applications of the results of this paper. Another area for future work is to tailor the current model to the problem of bed allocation in a hospital ward, with service rates depending on the type of arrival, which requires state expansion of the Markov chain.

Appendix

For a c-server system with n primary servers and $m = c - n$ secondary servers, Algorithm 1 computes the joint stationary probabilities $p_{i,j}$.

Competing Interests

The authors declare that they have no competing interests.

References

[1] M. El-Taha and J. R. Heath, "Traffic overflow in loss systems with selective trunk reservation," *Performance Evaluation*, vol. 41, no. 4, pp. 295–306, 2000.

[2] J. M. Akinpelu, "The overload performance of engineered networks with nonhierarchical and hierarchical routing," *The Bell System Technical Journal*, vol. 63, no. 7, 1984.

[3] A. Girard, "Blocking probability of noninteger groups with trunk reservation," *IEEE Transactions on Communications*, vol. 33, no. 2, pp. 113–120, 1985.

[4] F. Le Gall and J. Bernussou, "Blocking probabilities for trunk reservation policy," *IEEE Transactions on Communications*, vol. 35, no. 3, pp. 313–318, 1987.

[5] L. Paratz, "Traffic theory of circuit reservation," *Australian Telecommunication Research*, vol. 20, no. 2, pp. 51–60, 1986.

[6] E. A. Van Doorn, "On the overflow process from a finite Markovian queue," *Performance Evaluation*, vol. 4, no. 4, pp. 233–240, 1984.

[7] J. F. E. Wallin and B. Sanders, "The calculation of overflow moments in loss systems with selective trunk reservation," *Performance Evaluation*, vol. 15, no. 3, pp. 195–202, 1992.

[8] J. P. Young, *A queueing theory approach to the control of hospital in-patient census [Ph.D. dissertation]*, School of Engineering, The Johns Hopkins University, Baltimore, Md, USA, 1962.

[9] P. Kolesar, "A markovian model for hospital admission scheduling," *Management Science*, vol. 16, no. 6, pp. 384–396, 1970.

[10] A. O. Esogbue and A. J. Singh, "A stochastic model for an optimal priority bed distribution problem in a hospital ward," *Operations Research*, vol. 24, no. 5, pp. 884–898, 1976.

[11] E. P. C. Kao and G. G. Tung, "Bed allocation in a public health care delivery system," *Management Science*, vol. 27, no. 5, pp. 507–520, 1981.

[12] L. V. Green and V. Nguyen, "Strategies for cutting hospital beds: the impact on patient service," *Health Services Research*, vol. 36, no. 2, pp. 421–442, 2001.

[13] J. K. Cochran and K. T. Roche, "A multi-class queuing network analysis methodology for improving hospital emergency department performance," *Computers & Operations Research*, vol. 36, no. 5, pp. 1497–1512, 2009.

[14] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery, *Numerical Recipes in C: The Art of Scientific Computing*, Cambridge University Press, Cambridge, UK, 1992.