On the distribution of the number stranded in bulk-arrival, bulk-service queues of the M/G/1 form

Aykut Kahraman

M & T Bank, Buffalo, NY 14620

Abhijit Gosavi *

Missouri University of Science and Technology

219 Engineering Management, Rolla, MO 65401

Phone: (573)341-4624

Email: gosavia@mst.edu

Fax: (573)341-6567

February 3, 2011

Abstract

Bulk-arrival queues with single servers that provide bulk service are widespread in the real world, e.g., elevators in buildings, people-movers in amusement parks, air-cargo delivery planes, and automated guided vehicles. Much of the literature on this topic focusses on the development of the theory for waiting time and number in such queues. We develop the theory for the number *stranded*, i.e., the number of customers left behind after each service, in queues of the M/G/1 form, where there is single server, the arrival process is Poisson, the service is of a bulk nature, and the service time is a random variable. For the homogenous Poisson case, in our model the service time can have any given distribution. For the non-homogenous Poisson arrivals, due to a technicality, we assume that the service time is a discrete random variable. Our analysis is

*Corresponding author

not only useful for performance analysis of bulk queues but also in designing server capacity when the aim is to reduce the frequency of stranding. Past attempts in the literature to study this problem have been hindered by the use of Laplace transforms, which pose severe numerical difficulties. Our approach is based on using a discrete-time Markov chain, which bypasses the need for Laplace transforms and is numerically tractable. We perform an extensive numerical analysis of our models to demonstrate their usefulness. To the best of our knowledge, this is the first attempt in the literature to study this problem in a comprehensive manner providing numerical solutions.

Keywords: Queueing; bulk queues; downside risk

1 Introduction

Bulk-arrival, bulk-service systems are ubiquitous in the real world. Elevators in buildings (Chen, 2005) form the commonest example, where a server of a fixed capacity arrives after a random amount of time to serve a pool of customers. If the capacity of the server that arrives is less than the number of customers waiting, the server leaves behind some customers. Such customers are typically referred to have been *stranded.* A system in which a large number of customers are frequently stranded can lead to customers becoming displeased with it. Other settings where the bulk-service queue is found are trains (or people-movers) in amusement parks, cargo-delivering airplanes, and in the manufacturing setting, where the machine or equipment can serve multiple units at the same time, e.g., an automated guided vehicle (AGV) that picks up jobs at one machine to deliver them at another. In an amusement park, excessive delays at the end of the day waiting for the train are clearly unhelpful to tired parents. In the air-cargo delivery system, stranded mail can get delayed incurring penalties for the carrier. In a manufacturing system in which AGVs are used for material handling, inadequate capacity of the AGVs increases lead time and hence inventory. Finally, in the manufacturing setting, bulk queues are also seen in front of machines where service is provided simultaneously, e.g., testing centers (Chang et al., 2004) and heat-treatment units.

In many of these systems there tends to be a single server whose capacity is a variable, i.e., the service process is of a bulk nature. It is also common to find bulk-arrivals in such systems, e.g., a family of four arriving together to a station to board a train. The service time, which is the time it takes for the server to return to pick up customers, is usually a random variable. A typical goal in the study of bulk queues is the control of the service of the queue, which is a widely studied topic in non-bulk queues as well; see Teghem (1986) and Ke and Wang (2002). In the models that we study, we assume the server capacity to be a fixed variable to be optimized. The service time can have any given continuous distribution if the arrival process is a homogenous Poisson process. If the arrival process is of a non-homogenous Poisson type, our model will be capable of handling a discrete random variable for the service time. Our primary goal in this paper is to develop a distribution for the number *left behind* when the server leaves. This will be done under a variety of conditions for the server departure that we describe later. To use the standard notation from the literature, our models for bulk-arrival, bulk-service queues are of the following forms: $M_h^x/G^x/1$ and $M_{nh}^x/G^x/1$, where G^x denotes a service time that is generally distributed and that the service is of a bulk (variable) nature, M^x denotes the fact that the arrival process is Poisson and is of bulk nature, and h and nh denote homogeneous Poisson and non-homogeneous Poisson, respectively.

The server departure is determined by the so-called dispatching rule, which depends on the system being modeled. Two dispatching strategies that we focus on are: (i) A *regular* (R) dispatching strategy in which the server leaves according to its own schedule regardless of the number of customers waiting. (ii) A *holding* (H) strategy in which the server is held as long as the number of customers waiting is less than both of the following: a pre-specified number, B, and the available capacity of the server; the server departs as soon as the number waiting exceeds either of the two quantities.

Given a dispatching strategy, one has to additionally account for what happens with the stranded customers if there are any, i.e., use the appropriate *stranding* strategy. We focus on two types of stranding strategies: (i) 0-stranding where customers stranded in their very first attempt leave to get service elsewhere and (ii) ∞-stranding, where the customers stranded never leave. Additionally, we consider a third type of strategy: (iii) 1-stranding where customers leave after being stranded *twice*.

The literature has studied the problem of bulk queues widely (Bailey, 1954; Miller, 1959; Gaver, 1959; Saaty, 1960; Keilson, 1962; Jaiswal, 1964; Ghare, 1968; Cohen, 1969; Borthakur and Medhi, 1974; Neuts, 1981; Downton, 1986). Mejia-Tellez and Worthington (1994) model the queue length in bulk-arrival, bulk-service queues. Chen (2005) discusses a fuzzy bulk queue and provides some interesting examples of cable cars and elevators. Armero and Conesa (2004) model a bulk-arrival queueing system for the analysis of a make-to-stock production system. Bar-Lev et al. (2007) study a bulk-service queue with variable batch size for a group testing center that has applications in a medical testing center. Chang et al. (2004) develop performance measures for finite-buffer bulk-arrival, bulk-service queues using transform methods. See Medhi (2003) for a formal description of the bulk-service models and additional references.

Contributions of this paper: We now discuss the gaps in the literature that we seek to fill with this paper along with our contributions. Firstly, much of the existing literature in bulk queueing is directed towards estimating the waiting time or number. While this is an important task, our focus is on developing the *distribution of the number stranded*, which is an equally important performance measure for bulk queues. The importance of this measure stems from the fact that when the server capacity is designed for a bulk queue, the probability of stranding becomes an important design issue. The probability of stranding more than K customers (where K is a pre-specified number) is a *downside risk* for the designer, and the designer is expected to keep this risk under control. To the best of our knowledge, this paper is the first to address this issue in bulk queueing.

Secondly, a commonly advocated approach in the analysis of bulk queues is the use of the Laplace transforms. However, these techniques require the computation of the so-called complex zeroes, which "frequently becomes very challenging" (Thonemann and Brandeau, 1996). We sidestep the issue of transforms by directly computing the steady-state distribution of the underlying Markov chain from the elements of the associated semi-Markov kernels; this leads to a numerically tractable approach. In other words, we provide a computational tool for solving a problem, which, although *theoretically* solvable with Laplace transforms, is computationally very challenging.

Finally, our models are developed in the context of a wide array of stranding and dispatching strategies (discussed above), which are applicable in real-world systems. Further, we compare the results of our model with that of a simulation model, which is guaranteed to provide optimal solutions, in order to demonstrate the practical usefulness of our approach.

The rest of this paper is organized as follows. Section 2 presents the mathematical model for performance evaluation. Section 3 discusses some structural properties in the context of the performance evaluation model. Section 4 provides results from computational experiments, while Section 5 concludes this paper.

2 The semi-Markov kernel model

We begin with some general assumptions about our models: (i) The inter-arrival times of the customers (customer group) are i.i.d. and the arrival process is either homogeneous or non-homogeneous Poisson. (ii) The customer group sizes are i.i.d. and assume discrete values. (iii) The service times are i.i.d. random variables. (iv) The dispatch station has an infinite capacity of waiting space, making our queue one of infinite waiting capacity. (iv) The holding time of the server, wherever relevant, does not affect the next service time.

We now present some notation. Let X_m denote the number of customers stranded when the mth departure occurs, T_m denote the time of the mth departure, and U_m denote the number of customers that arrive in one service period after the $(m-1)$th departure. Z will denote the maximum capacity of the server. Since inter-arrival times, customer-size distributions, and service times are i.i.d., values of $\{U_m\}$ are i.i.d. They are also independent of $\{X_m\}$. Let $\mathcal{I} = \{0, 1, 2, \ldots, \}$, $\mathcal{I}^+ = \{1, 2, \ldots, \}$, $\mathbf{X} = \{X_m : m = 0, 1, 2, \ldots\}$ and $\mathbf{T} = \{T_m : m = 0, 1, 2, \ldots\}$. Now since for all $m, T_m \geq 0$ and $j \in \mathcal{I}$, $\mathsf{P}\{X_{m+1} = j, T_{m+1} - T_m \leq t | X_0, \ldots, X_m; T_0, \ldots, T_m\} = \mathsf{P}\{X_{m+1} = j, T_{m+1} - T_m \leq t | X_m\}$, $(\mathbf{X}, \mathbf{T})$ is a Markov renewal process (see Kao (1997); pp. 323). If the system is in state i at the previous departure time, then the probability that the next departure will occur after t units of time *and* that the system will be in state j then will be denoted by $Q(i, j, t)$. Mathematically, $Q(i, j, t) \equiv \mathsf{P}\{\, X_{m+1} = j, T_{m+1} - T_m \leq t | X_m = i\,\}$. $\mathbf{Q}(\mathbf{t})$ denotes a matrix whose element in its ith row and jth column is $Q(i, j, t)$. Let $\mathbf{P}$ denote the transition probability matrix

(TPM) of the discrete-time Markov chain underlying the Markov renewal process, and $P(i,j)$ denote the one-step transition probability of going from state i to state j. The elements of $\mathbf{P}$ can be calculated from $\mathbf{Q}(t)$ as follows (see e.g., Kao (1997)): $P(i,j) = \lim_{t\to\infty} Q(i,j,t)$. The TPM of the underlying Markov chains in the semi-Markov kernels will be constructed for homogeneous Poisson arrivals in subsection 2.1 and for non-homogeneous Poisson arrivals in subsection 2.2. Before plunging into the mathematical details of the models, we present an overview of the underlying scheme in semi-Markov kernel models in queueing.

Constructing the kernel: To construct the semi-Markov process (kernel) underlying the queueing system, we view the system only at the departure instants. The time between the departure instants is determined by the service time. We can construct an discrete-time Markov chain from the kernel as follows. We use the number stranded to denote the state of the Markov chain. Given the current state (i.e., the number stranded at current departure), we will develop an expression that evaluates the probability that the system will occupy any other state at the next departure instant. This can be accomplished provided one has access to the distribution of the time elapsed between successive departure instants, i.e., the service time. For more details of this process, the reader is referred to pages 256-259 of Medhi (2003). This idea will be used in all our models for the different dispatching and stranding strategies. The transition probabilities of going from one state to another will depend on the scenario considered, i.e., the holding and dispatching strategy and whether the arrival process is homogenous or non-homogenous Poisson. Once the transition probabilities are available, one can use the standard invariance equations (Ross, 2003) to determine the steady-state (limiting) probabilities of the underlying Markov chain. These steady-state probabilities will yield the distribution of the number stranded, along with the first and the second moment.

2.1 Homogeneous Poisson arrivals

Let λ denote the Poisson rate of arrival, $A(t)$ the number of customers that arrive in a time interval of length t, and L the customer size. Also, $L(l) \equiv \mathsf{P}(L = l)$, where $l \in \{1, 2, \ldots, L_{\max}\}$. Since the values of L are i.i.d. and finite, the arrival process at the dispatch station forms a *compound Poisson process* allowing us to compute the distribution of $A(t)$ via a recursion (see e.g., Kao (1997); pp. 73). Using $P_n^t \equiv \mathsf{P}(A(t)=n)$, and $P_0^t = exp(-\lambda t)$,

$$P_n^t = \frac{\lambda t}{n}\sum_{j=1}^{l} jL(j)P_{n-j}^t, \text{ where } l=n \text{ if } 1 \le n \le L_{\max} \text{ and } l = L_{\max} \text{ if } n > L_{\max}. \tag{1}$$

We begin with the holding strategy and then present the regular strategy as its special case.

2.1.1 Holding dispatching

D_m will denote the available capacity of the server at the mth departure instant. If D_m is greater than B, a pre-specified number, the server is held until B customers appear; otherwise it leaves with D_m customers. Let $B_m^* = \min(B, D_m)$, and the server departs with B_m^* customers. The server is held as long as there are $(B_m^* - k)$ (with $k > 0$) or fewer customers in the queue. However, since the size of the customer can range from 1 to $L_{\max}$, the number of customers waiting at the station just before the mth departure takes values from: $\{B_m^*, (B_m^*+1), (B_m^*+2), \ldots, (B_m^*+L_{\max}-1)\}$; these values can be obtained by adding $k, k+1, k+2, \ldots, L_{\max}$ to (B_m^*-k) for $k = 1, 2, \ldots, L_{\max}$. Thus when the server departs, since it carries away B_m^* customers, the number stranded assumes any value between 0 (i.e., $B_m^* - B_m^*$) and $(L_{\max}-1)$ (i.e., $B_m^* + L_{\max} - 1 - B_m^*$). To determine $Q(i,j,t)$, we need to account for the holding time.

Let S_m denote the total number of customers that arrive during the time the server is held before the mth departure. Let S_m^n denote the total number of customers from the first n groups that arrive during the time the server is held before the mth departure. Defining $\mathsf{P}(S_m^n = i) = \sum_{k=1}^{\min(L_{\max},i)} \mathsf{P}(S_{n-1} = i-k)L(k)$, with boundary condition: $\mathsf{P}(S_1 = i) = L(i)$ for $i \in \{1, 2, \ldots L_{\max}\}$, we can compute the *pmf* of S_m with: $\mathsf{P}(S_m = i) = \sum_{n=1}^{i} \mathsf{P}(S_m^n = i),\ i > 0$. Now, for $i < B$ and $0 < j < L_{\max}$:

$$\begin{aligned} H_m(i,j,t) &= \sum_{d=i+1}^{B} \mathsf{P}(D_m = d)\left(\sum_{r=0}^{d-i-1} \mathsf{P}(A(t)=r)G_m(d-i-r,j)\right) \\ &+ \sum_{d=B+1}^{Z} \mathsf{P}(D_m = d)\left(\sum_{r=0}^{B-i-1} \mathsf{P}(A(t)=r)G_m^*(d-i-r,j,r)\right), \text{ where} \end{aligned}$$

$G_m(k,j) = \sum_{n=j+1}^{\min(L_{\max},(k+j))} L(n)\mathsf{P}(S_m = k+j-n)$ and $G_m^*(k,j,r) = \sum_{n=j+1+r-B}^{\min(L_{\max},(k+j))} L(n)\mathsf{P}(S_m = k+j-n)$.

∞-stranding strategy: There are four main cases; also, we define $\Gamma_k = \sum_{i=k}^{Z} \mathsf{P}(D_m = i)$.

Case 1: If $(j-i) > 0$,

Case 1a: If $j < L_{\max}$, $i < B$,
$Q(i,j,t) = \int_0^t \left[\sum_{n=j-i}^{Z+j-i} \mathsf{P}(A(u)=n)\mathsf{P}(D_m = n-j+i) + H_m(i,j,u)\right] d\mathsf{P}\{T_{m+1} - T_m \le u | X_m = i\}$.

Case 1b: Else, $Q(i,j,t) = \int_0^t \sum_{n=j-i}^{Z+j-i} \mathsf{P}(A(u)=n)\mathsf{P}(D_m = n-j+i) d\mathsf{P}\{T_{m+1} - T_m \le u | X_m = i\}$.

Case 2: If $(j-i) \le 0$, $(i-j) \le Z$, and $j \ne 0$,

Case 2a: If $j < L_{\max}$, $i < B$,
$Q(i,j,t) = \int_0^t \left[\sum_{n=0}^{Z+j-i} \mathsf{P}(A(u)=n)\mathsf{P}(D_m = n-j+i) + H_m(i,j,u)\right] d\mathsf{P}\{T_{m+1} - T_m \le u | X_m = i\}$.

Case 2b: Else, $Q(i,j,t) = \int_0^t \sum_{n=0}^{Z+j-i} \mathsf{P}(A(u)=n)\mathsf{P}(D_m = n-j+i) d\mathsf{P}\{T_{m+1} - T_m \le u | X_m = i\}$.

Case 3: If $(j-i) \le 0$, $(i-j) \le Z$, and $j = 0$,

Case 3a: If $i < B$,
$Q(i,j,t) = \int_0^t \left[\sum_{n=0}^{Z+j-i} \mathsf{P}(A(u)=n)\Gamma_{n-j+i} - \sum_{n=1}^{L_{\max}-1} H_m(i,n,u)\right] d\mathsf{P}\{T_{m+1} - T_m \le u | X_m = i\}$.

Case 3b: Else, $Q(i,j,t) = \int_0^t \sum_{n=0}^{Z+j-i} \mathsf{P}(A(u)=n)\Gamma_{n-j+i} d\mathsf{P}\{T_{m+1} - T_m \le u | X_m = i\}$.

Case 4: If $(j-i) < 0$ and $(i-j) > Z$, then $Q(i,j,t) = 0$.

0-stranding strategy:

Case 1: If $j > 0$,

Case 1a: If $j < L_{\max}$,

$$Q(i,j,t) = \int_0^t \left[\sum_{n=j}^{Z+j} \mathsf{P}(A(u)=n)\mathsf{P}(D_m = n-j) - H_m(0,j,u)\right] d\mathsf{P}\{T_{m+1} - T_m \le u | X_m = i\}.$$

Case 1b: Else, $Q(i,j,t) = \int_0^t \sum_{n=j}^{Z+j} \mathsf{P}(A(u)=n)\mathsf{P}(D_m = n-j) d\mathsf{P}\{T_{m+1} - T_m \le u | X_m = i\}$.

Case 2: If $j = 0$,
$Q(i,j,t) = \int_0^t \left[\sum_{n=0}^{Z} \mathsf{P}(A(u)=n)\Gamma_n - \sum_{n=1}^{L-1} H_m(0,n,u)\right] d\mathsf{P}\{T_{m+1} - T_m \le u | X_m = i\}$.

2.1.2 Regular dispatching

The relevant expressions for regular dispatching, where the server is not held but leaves after the service is complete, can be developed from those for the holding (dispatching) case by setting $H_m(.,.,.) = 0$. The expressions can be derived from the corresponding cases for the holding strategy. When the server is not held, a 1-stranding strategy that we discuss next may be relevant for regular dispatching.

1-stranding strategy: Here the state will have to be defined as a 2-tuple: $\hat{i} = (i_{fs}, i_{ss})$, where for the associated departure, i_{fs} denotes the number stranded for the first time and i_{ss} denotes the number shipped via another company. Then, the elements of $\mathbf{Q(t)}$ can be determined via the following three cases.

Case 1: If $j_{ss} > 0$, $j_{ss} \le i_{fs}$ and $(i_{fs} - j_{ss}) \le Z$,
$Q(\hat{i},\hat{j},t) = \int_0^t \mathsf{P}(A(u) = j_{fs})\mathsf{P}(D_m = i_{fs} - j_{ss}) d\mathsf{P}\{T_{m+1} - T_m \le u | X_m = \hat{i}\}$.

Case 2: If $j_{ss} = 0$, $i_{fs} \le Z$ and $j_{fs} > 0$,
$Q(\hat{i},\hat{j},t) = \int_0^t \sum_{n=i_{fs}}^{Z} \mathsf{P}(A(u) = n - i_{fs} + j_{fs})\mathsf{P}(D_m = n) d\mathsf{P}\{T_{m+1} - T_m \le u | X_m = \hat{i}\}$.

Case 3: If $j_{ss} = 0$, $i_{fs} \le Z$ and $j_{fs} = 0$,
$Q(\hat{i},\hat{j},t) = \int_0^t \sum_{n=0}^{Z-i_{fs}} \mathsf{P}(A(u) = n)\Gamma_{n+i_{fs}} d\mathsf{P}\{T_{m+1} - T_m \le u | X_m = \hat{i}\}$.

2.2 Non-homogeneous Poisson arrivals

We now work out the details for the case of non-homogeneous Poisson arrivals. Let $\lambda(t)$ denote the intensity function. Let $\bar{\lambda}(t_i, t_j) \equiv \int_{t_i}^{t_j} \lambda(\tau) d\tau$ with $t_j \ge t_i$. Then $\mathsf{P}(A(t_j - t_i))$ can be computed with Equation (1) using $P_0^{t_j - t_i} = exp(-\bar{\lambda}(t_i, t_j))$ and $P_n^{t_j - t_i} = \frac{\bar{\lambda}(t_i,t_j)}{n} \sum_{j=1}^{l} jL(j) P_{n-j}^{t_j - t_i}$ in Equation (1). Because we will need to introduce time into the state space, we are forced to assume that the random service time will be finite and discrete; this is not a very strong assumption for an air-cargo or amusement park system where typically the service time is measured as multiples of time periods of fixed duration, e.g., 1 hour. Let $T_s(t) = \mathsf{P}(\text{Service Time } = \Psi t)$ where Ψ denotes a time period of fixed duration and $t \in \mathcal{I}^+ = \{1, 2, 3, \ldots, \}$.

2.2.1 Holding dispatching

Since time is incorporated into the state space, H_m has to be re-defined.

$$\begin{aligned} H_m(i,j,t_i,t_j) &= \sum_{d=i+1}^{B} \mathsf{P}(D_m = d) \left(\sum_{r=0}^{d-i-1} \mathsf{P}(A(t_j - t_i) = r) G_m(d-i-r,j) \right) \\ &+ \sum_{d=B+1}^{Z} \mathsf{P}(D_m = d) \left(\sum_{r=0}^{B-i-1} \mathsf{P}(A(t_j - t_i) = r) G^*_m(d-i-r,j,r) \right). \end{aligned} \quad (2)$$

∞-stranding strategy: Here the state will be defined as a 2-tuple: $\hat{i} = (i, t_i)$, where i denotes the number stranded at the end of the associated epoch and t_i the time at which the epoch ends.

Case 1: If $(j - i) > 0$,

Case 1a: If $j < L_{\max}$, $i < B$,
$P(\hat{i}, \hat{j}) = \mathsf{P}(T_s = t_j - t_i) \left[\sum_{n=j-i}^{Z+j-i} \mathsf{P}(A(t_j - t_i) = n) \mathsf{P}(D_m = n - j + i) + H_m(i,j,t_i,t_j) \right]$.

Case 1b: Else,
$P(\hat{i}, \hat{j}) = \mathsf{P}(T_s = t_j - t_i) \sum_{n=j-i}^{Z+j-i} \mathsf{P}(A(t_j - t_i) = n) \mathsf{P}(D_m = n - j + i)$.

Case 2: If $(j - i) \leq 0$, $(i - j) \leq Z$, and $j \neq 0$,

Case 2a: If $j < L_{\max}$, $i < B$,
$P(\hat{i}, \hat{j}) = \mathsf{P}(T_s = t_j - t_i) \left[\sum_{n=0}^{Z+j-i} \mathsf{P}(A(t_j - t_i) = n) \mathsf{P}(D_m = n - j + i) + H_m(i,j,t_i,t_j) \right]$.

Case 2b: Else, $P(\hat{i}, \hat{j}) = \mathsf{P}(T_s = t_j - t_i) \sum_{n=0}^{Z+j-i} \mathsf{P}(A(t_j - t_i) = n) \mathsf{P}(D = n - j + i)$.

Case 3: If $(j - i) \leq 0$, $(i - j) \leq Z$, and $j = 0$, Case 3a: If $i < B$,

$P(\hat{i},\hat{j}) = \mathsf{P}(T_s = t_j - t_i)\left[\sum_{n=0}^{Z+j-i} \mathsf{P}(A(t_j - t_i) = n)\Gamma_{n-j+i} - \sum_{n=1}^{L_{\max}-1} H_m(i,j,t_i,t_j)\right].$

Case 3b: Else, $P(\hat{i},\hat{j}) = \mathsf{P}(T_s = t_j - t_i)\sum_{n=0}^{Z+j-i} \mathsf{P}(A(t_j - t_i) = n)\Gamma_{n-j+i}$.

Case 4: If $(j-i) < 0$ and $(i-j) > Z$, then $P(\hat{i},\hat{j}) = 0$.

0-stranding strategy: Case 1: If $j > 0$,

Case 1a: If $j < L_{\max}$, $P(\hat{i},\hat{j}) = \mathsf{P}(T_s = t_j - t_i)\left[\sum_{n=j}^{Z+j} \mathsf{P}(A(t_j - t_i) = n)\mathsf{P}(D_m = n-j) - H_m(0,j,t_i,t_j)\right].$

Case 1b: Else, $P(\hat{i},\hat{j}) = \mathsf{P}(T_s = t_j - t_i)\left[\sum_{n=j}^{Z+j} \mathsf{P}(A(t_j - t_i) = n)\mathsf{P}(D_m = n-j) - H_m(0,j,t_i,t_j)\right].$

Case 2: If $j = 0$, $P(\hat{i},\hat{j}) = \mathsf{P}(T_s = t_j - t_i)\left[\sum_{n=0}^{Z} \mathsf{P}(A(t_j - t_i) = n)\Gamma_n - \sum_{n=1}^{L_{\max}-1} H_m(0,n,t_i,t_j)\right].$

2.2.2 Regular dispatching

Like before, expressions for the regular strategy can be obtained from those for the holding strategy by setting $H_m(.,.,.) = 0$. In addition, we consider a 1-stranding strategy.

1-stranding strategy: Here, the state will have to be as a three-tuple: $\hat{i} = (i_{fs}, i_{ss}, t_i)$, where t_i denotes the time of the associated departure.

Case 1: If $j_{ss} > 0$, $j_{ss} \leq i_{fs}$ and $(i_{fs} - j_{ss}) \leq Z$,

$P(\hat{i},\hat{j}) = \mathsf{P}(T_s = t_j - t_i)\mathsf{P}(A(t_j - t_i) = j_{fs})\mathsf{P}(D_m = i_{fs} - j_{ss}).$

Case 2: If $j_{ss} = 0$, $i_{fs} \leq Z$ and $j_{fs} > 0$,

$P(\hat{i},\hat{j}) = \mathsf{P}(T_s = t_j - t_i)\sum_{n=i_{fs}}^{Z} \mathsf{P}(A(t_j - t_i) = n - i_{fs} + j_{fs})\mathsf{P}(D_m = n).$

Case 3: If $j_{ss} = 0$, $i_{fs} \leq Z$ and $j_{fs} = 0$,

$P(\hat{i},\hat{j}) = \mathsf{P}(T_s = t_j - t_i)\sum_{n=0}^{Z-ifs} \mathsf{P}(A(t_j - t_i) = n)\Gamma_{n+i_{fs}}.$

3 Performance evaluation and structural properties

Three performance measures that are of interest to us are: the expected number of stranded customers ($\mathsf{E}[X]$), the variance of the number stranded ($\mathsf{Var}[X]$), and the probability that the number stranded exceeds a given threshold K ($\mathsf{P}(X > K)$), which is also called the downside risk. Let $\vec{\Pi}$ denote the row vector of steady-state probabilities associated with the Markov chain, whose nth element is denoted by Π_n. The elements of $\vec{\Pi}$ can be obtained by solving the following system of invariance equations (Ross, 2003): $\vec{\Pi}\mathbf{P} = \vec{\Pi}; \sum_{n \in \mathcal{S}} \Pi_n = 1$. The performance measures can then be computed using: $\mathsf{E}[X] = \sum_{n \in \mathcal{S}} n\Pi_n$, $\mathsf{Var}[X] = \sum_{n \in \mathcal{S}} n^2\Pi_n - \mathsf{E}^2[X]$, and $\mathsf{P}(X > K) = \sum_{n=K+1}^{|\mathcal{S}|} \Pi_n$.

We now prove some structural results, the proofs of which are in the Appendix.

Ergodicity: In order to compute the steady-state probabilities, it is necessary to show that the Markov chain is ergodic. We present the following result for ∞-stranding, which can be extended easily to other stranding strategies.

Theorem 1 *Suppose the system is operating under the ∞-stranding strategy. Let U_m denote the number of customers that arrive in the service period before the mth departure and D_m denote the available capacity before the mth departure. If $\mathsf{E}[U_m] < \mathsf{E}[D_m]$, then $\mathbf{X}$ is ergodic for all the dispatching strategies considered.*

Proof The fact that $P(i, i) \neq 0$ follows for the regular strategy from Cases 2 and 3 and for the holding strategy from Cases 2a, 2b, 3a, and 3b. Hence $\{X_m\}$ is aperiodic. That it is irreducible can be proved on an element-by-element basis. Theorem 2 of Pakes (1969) shows that a Markov chain is ergodic if the following conditions are satisfied: (i) $|\psi_j| < \infty$ and (ii) $\limsup_{j\to\infty} \psi_j < 0$, where $\psi_j = \mathsf{E}[X_{m+1} - X_m | X_m = j]$. At the mth departure, where $m \in \{1, 2, \ldots\}$, it follows from the nature of the queue that $X_{m+1} = X_m + U_m - D_m + B_m$, where B_m is defined for regular and holding strategies as follows:

Regular strategy: $B_m = \max(0, D_m - X_m - U_m)$.

Holding strategy:

$$B_m = \begin{cases} \min(B, D_m) - X_m - U_m \text{ if } X_m + U_m < \min(B, D_m) \\ 0 \text{ if } X_m + U_m < B \text{ and } X_m + U_m \geq D_m \\ \max(0, D_m - X_m - U_m) \text{ if } X_m + U_m \geq B \end{cases}$$

Since $\mathsf{E}[U_m]$ and $\mathsf{E}[D_m]$ are bounded, $\mathsf{E}[B_m|X_m = j]$ is also bounded for both strategies. Hence, condition (i) is satisfied. By definition of B_m in the regular strategy, if $j > D_m - U_m$, $B_m = 0$. Also, for holding strategy, if $j > \max(D_m - U_m, B - U_m)$, $B_m = 0$. Therefore, it is true for all four strategies that $\limsup_{j\to\infty} \psi_j = \mathsf{E}[U_m] - \mathsf{E}[D_m]$. Since $\mathsf{E}[U_m] < \mathsf{E}[D_m]$, condition (ii) follows. ∎

A threshold property: We will now show that the problem structure has an interesting threshold property that should appeal to managers interested in optimizing the system. This threshold property will ensure that given an upper limit on the downside risk, one can determine an optimal capacity that minimizes costs or the downside risk.

Consider a server with a given maximum capacity. Consider a 2-state chain in which the system will be said to be in state 1 (state 2) if K or less (more than K) customers are stranded when the server takes off. Let $\mathbf{L}$ ($\mathbf{L}'$) denote the TPM of the chain when the maximum capacity of the server is C (C'). Clearly then, if $L(i, j)$ denotes the transition probability of going from state i to state j when the TPM is $\mathbf{L}$, $L(1, 1) = \sum_{i<K,j<K} P(i, j)$ and $L(2, 1) = \sum_{i\geq K,j<k} P(i, j)$.

Assumption 1 *If $C' > C$, then for some $\epsilon_1 > 0$ and some $\epsilon_2 > 0$.*

$$L'(1, 1) = L(1, 1) + \epsilon_1 \text{ and } L'(2, 1) = L(2, 1) + \epsilon_2. \tag{3}$$

Since the transition probabilities can be evaluated from the elements of the Markov chain, whether the system satisfies Assumption 1 can be easily verified. Also, it is not hard to show that the chain defined by $\mathbf{L}$ or $\mathbf{L}'$ is a Markov chain.

Theorem 2 *Let Assumption 1 hold. If the maximum capacity of the server is increased, the probability that K or more customers will be stranded decreases for any $K > 0$.*

Proof For the proof, the following lemma is needed.

Lemma 1 *If* $b_1 \neq 0$, $a_1 > b_1$, $a < b$, $b + b_1 > 0$, *and* $a_1 > 0$,

$$\frac{a + a_1}{b + b_1} > \frac{a}{b}, \text{ where } a, b, a_1 \text{ and } b_1 \text{ are scalars.}$$

Proof If $b_1 < 0$, the result follows from the following: $\frac{a+a_1}{b+b_1} = \frac{a}{b+b_1} + \frac{a_1}{b+b_1} > \frac{a}{b} + \frac{a_1}{b+b_1} > \frac{a}{b}$. If $b_1 > 0$, the following argument is used. Since $a_1 > b_1$ and $b > a$, one has that: $ab + a_1 b > ab + ab_1$, which implies that $b(a + a_1) > a(b + b_1)$, from which the result follows by rearrangement of terms around the inequality. ∎

Now, let π_i and π'_i denote the limiting probabilities of state i associated with $\mathbf{L}$ and $\mathbf{L}'$ respectively. Then one needs to prove that $\pi'_1 > \pi_1$. Now, from $\pi\mathbf{L} = \pi$ and $\sum_i \pi_i = 1$, it follows that $\pi_1 = \pi_1 L(1,1) + \pi_2 L(2,1)$, $\pi_2 = \pi_1 L(1,2) + \pi_2 L(2,2)$ and $\pi_1 + \pi_2 = 1$, from which one has that $\pi_1 = \frac{L(2,1)}{1-L(1,1)+L(2,1)}$. Similarly,

$$\pi'_1 = \frac{L'(2,1)}{1 - L'(1,1) + L'(2,1)} = \frac{L(2,1) + \epsilon_2}{1 - L(1,1) - \epsilon_1 + L(2,1) + \epsilon_2} \text{ (from (3)).}$$

Using the above and setting $a = L(2,1)$, $b = 1 - L(1,1) + L(2,1)$, $a_1 = \epsilon_2$ and $b_1 = \epsilon_2 - \epsilon_1$, from Lemma 1, one has that $\pi'_1 = \frac{L(2,1)+\epsilon_2}{1-L(1,1)-\epsilon_1+L(2,1)+\epsilon_2} > \frac{L(2,1)}{1-L(1,1)+L(2,1)} = \pi_1$. ∎

A north-west corner structure: The north-west corner truncation of an infinite-dimensional TPM to a finite one, in order to compute the steady-state probabilities, is a well-known procedure (Seneta, 1981). It works well when the elements in the TPM taper off becoming negligible as one proceeds to the eastern and southern sides of the infinite-dimensional matrix. Now, we show that such a property is actually displayed by the Markov chain that we develop in this paper.

Theorem 3 *For the* ∞*-stranding strategy and any dispatching strategy with homogeneous Poisson arrivals in bulk, consider the infinite-sized TPM,* $\mathbf{P}$. *For any finite value of* $i, j \in \mathcal{I}^+$ *and non-zero probability* $P(i,j)$, *there exists a finite* $j_i > i$ *and* $j_i > j$ *such that* $\forall j^* \in \mathcal{I}^+$ *and* $j^* > j_i$, $P(i, j^*) < P(i,j)$.

Proof

$$\begin{aligned}
\text{Since } \lim_{n\to\infty} P_n^u &= \lim_{n\to\infty}\left[\frac{\lambda u}{n}\sum_{j=1}^{L_{\max}} jL(j)P_{n-j}^u\right] \quad \text{(by Equation (1))}\\
&\le \lim_{n\to\infty}\left[\frac{\lambda u}{n}\sum_{j=1}^{L_{\max}} j\right] \quad \text{(since } L(j) \text{ and } P_{n-j}^u \text{ are probabilities)}\\
&\le \lim_{n\to\infty}\left[\frac{\lambda u}{n}(L_{\max})^2\right] \le 0,
\end{aligned} \tag{4}$$

we have that $\lim_{n\to\infty} P_n^u \equiv \lim_{n\to\infty} \mathsf{P}(A(u)=n) = 0$.

For homogeneous Poisson arrivals, if $j > i$ in the case of regular dispatching and if $j > i$ and $j > B$ in the case of holding:

$$Q(i,j,t) = \int_0^t \sum_{n=j-i}^{Z+j-i} \mathsf{P}(A(u)=n)\mathsf{P}(D=n-j+i)d\mathsf{P}\{T_{m+1}-T_m \le u|X_n = i\}.$$

Since $P(i,j) = \lim_{t\to\infty} Q(i,j,t)$ and $\lim_{n\to\infty} \mathsf{P}(A(u)=n) = 0$, $\lim_{j\to\infty} P(i,j) = 0$. The result follows from the definition of a limit of a sequence. ∎

The result can be easily extended to other stranding strategies and non-homogeneous Poisson arrivals. If one assumes the customer size to be 1 and homogeneous Poisson arrivals, a better structural property can be proved. We define $P_\tau(i,j) \equiv Q(i,j,\tau)$, where τ is sufficiently large.

Theorem 4 *Consider a system working under the assumption of ∞-stranding strategy and regular dispatching strategy with single, homogeneous Poisson arrivals. For any finite value of $i \in I^+$, there exists a finite $j_i \ge i$ such that $P_\tau(i,j)$ decreases monotonically for all $j \ge j_i$ and any $\tau > 0$. For the holding strategy, the following holds. For any finite value of $i \in I^+$, there exists a finite $j_i \ge i$ such that $P_\tau(i,j)$ decreases monotonically for all $j \ge j_i$, $j \ge B$ and any $\tau > 0$.*

Proof Let $j_i = \lceil \lambda\tau + i - 1\rceil$ (where $\lceil a \rceil$ denotes the smallest integer greater than or equal to a). Now, for any i, consider a $j \ge j_i$. Since $j_i \ge i$, $j \ge i$. In addition, for the case of holding and the cancellation strategies, assume that $j \ge B$. Then Condition 1 holds for the regular strategy, Condition 2 for the cancellation strategy, and Condition 4 for the holding strategy. Hence, it can be written that:

$$P_\tau(i,j) = \int_0^\tau \sum_{n=j-i}^{Z+j-i} \mathsf{P}(A(t)=n)\mathsf{P}(D=n-j+i)d\mathsf{P}\{T_{m+1}-T_m \le u|X_n=i\} \text{ and}$$

$$P_\tau(i,j+1) = \int_0^\tau \sum_{n=j+1-i}^{Z+j+1-i} \mathsf{P}(A(t)=n)\mathsf{P}(D=n-j-1+i)d\mathsf{P}\{T_{m+1}-T_m \le u|X_n=i\}.$$

From the above, it follows that:

$$\begin{aligned} P_\tau(i,j) - P_\tau(i,j+1) &= \int_0^\tau \Bigg[\sum_{n=j-i}^{Z+j-i} \mathsf{P}(A(t)=n)\mathsf{P}(D=n-j+i) \\ &- \sum_{n=j+1-i}^{Z+j+1-i} \mathsf{P}(A(t)=n)\mathsf{P}(D=n-j-1+i)\Bigg] d\mathsf{P}\{T_{m+1}-T_m \le u|X_n=i\} \\ &= \int_0^{\tau_\epsilon} \sum_{n=0}^{Z} \mathsf{P}(D=n)\left[\mathsf{P}(A(t)=n+j-i) - \mathsf{P}(A(t)=n+j+1-i)\right] \\ &\times\ d\mathsf{P}\{T_{m+1}-T_m \le u|X_n=i\}. \end{aligned} \tag{5}$$

Now $j \ge j_i = \lceil \lambda\tau + i - 1 \rceil \ge \lambda\tau + i - 1$. Hence $j - i + 1 \ge \lambda\tau$. Then, for all $t \le \tau$, $1 \ge \frac{\lambda\tau}{j-i+1}$. Using this, one has that for all $t \le \tau$,

$$\begin{aligned} \mathsf{P}(A(t)=n+j-i) - \mathsf{P}(A(t)=n+j+1-i) &= \frac{e^{-\lambda t}(\lambda t)^{n+j-i}}{(n+j-i)!} - \frac{e^{-\lambda t}(\lambda t)^{n+j-i+1}}{(n+j-i+1)!} \\ &= \frac{e^{-\lambda t}(\lambda t)^{n+j-i}}{(n+j-i)!}\left(1 - \frac{\lambda t}{n+j-i+1}\right) \\ &\ge 0. \end{aligned}$$

The above together with (5) implies that $P_\tau(i,j) - P_\tau(i,j+1) \ge 0$ for $j \ge j_i$ in the case of the regular strategy and for the holding strategies if in addition $j \ge B$. ∎

A finite estimator: For the purposes of numerical integration, required in computing the transition probabilities from the semi-Markov kernel's elements in the homogeneous case, one needs to approximate the upper limit of the integral by a finite quantity. (Note that for the non-homogeneous case, we assume finite values for the service time, ruling out the need for finite estimation.) To show the κ-approximateness of this process, for any $\kappa > 0$, the following lemma will be useful.

Lemma 2 *For any $\epsilon > 0$, there exists a large enough $\tau_\epsilon > 0$ such that for all $\tau \geq \tau_\epsilon$, $\int_\tau^\infty \mathsf{P}(A(\tau) = n)d\tau < \epsilon \quad \forall n$, where $f(\tau)$ is the pdf of the service time.*

Proof For a given $\epsilon > 0$, there exists a $\tau_\epsilon > 0$ such that $\int_{\tau_\epsilon}^\infty f(\tau)d\tau < \epsilon$.

$$\text{Since } 0 \leq \mathsf{P}(A(\tau) = n) \leq 1, \text{ one has that } \int_{\tau_\epsilon}^\infty \frac{e^{-\lambda\tau}(\lambda\tau)^n}{n!} f(\tau)d\tau < \epsilon. \qquad \blacksquare$$

Now for any case of the semi-Markov kernel and for any dispatching strategy, in general:

$$Q(i, j, \infty) = \sum_{n=0}^{\bar{n}} \int_0^\infty R(i, j, n)\mathsf{P}(A(t) = n)f(\tau)d\tau \tag{6}$$

where $\bar{n}$, a finite number, and $R(i, j, n)$, a function, both depend on the associated case. The expression (6) follows from the fact that for any case, $d\mathsf{P}\{.\} = f(\tau)d\tau$. Then, using sufficiently large but finite values for the upper limit in the integrals in (6), one can develop an approximation for $Q(i, j, \infty)$. The approximation will be defined as follows for any given value of $\kappa > 0$:

$$Q_\kappa(i, j) \equiv \sum_{n=0}^{\bar{n}} \int_0^{\tau_{\epsilon_n}} R(i, j, n)\mathsf{P}(A(\tau) = n)f(\tau)d\tau, \tag{7}$$

where $\epsilon_n = \kappa/(\bar{n} + 1)$ and τ_{ϵ_n} is selected as defined in Lemma 2 (using $\epsilon = \epsilon_n$). The latter implies that for all $\tau \geq \tau_{\epsilon_n}$,

$$\int_\tau^\infty \mathsf{P}(A(\tau) = n)f(\tau)d\tau < \epsilon_n \qquad \forall n. \tag{8}$$

Theorem 5 *For a given value of $\kappa > 0$ and for every i and j, there exists a $Q_\kappa(i, j)$ such that $|Q_\kappa(i, j) - \lim_{t\to\infty} Q(i, j, t)| < \kappa$.*

Proof The notation $\lim_{t\to\infty} Q(i, j, t) = Q(i, j, \infty)$ will be used. It is also true that for any finite n

$$R(i, j, n) \leq 1, \tag{9}$$

since $R(i, j, n)$ is always a probability. Then for any $i, j \in \mathcal{S}$,

$$\begin{aligned} |Q(i, j, \infty) - Q_\kappa(i, j)| \quad = \quad & \Bigg| \sum_{n=0}^{\bar{n}} R(i, j, n) \int_0^{\tau_{\epsilon_n}} \mathsf{P}(A(\tau) = n)f(\tau)d\tau + \\ & R(i, j, n) \int_{\tau_{\epsilon_n}}^\infty \mathsf{P}(A(\tau) = n)f(\tau)d\tau - R(i, j, n) \int_0^{\tau_{\epsilon_n}} \mathsf{P}(A(\tau) = n)f(\tau)d\tau \Bigg| \end{aligned}$$

$$
\begin{aligned}
& \text{(from Equations (6) and (7))} \\
= & \left|\sum_{n=0}^{\bar{n}} R(i,j,n) \int_{\tau_{\epsilon_n}}^{\infty} \mathsf{P}(A(\tau)=n) f(\tau) d\tau \right| \\
\leq & \sum_{n=0}^{\bar{n}} \epsilon_n \text{ (from Equations (9) and (8))} \\
= & \kappa \quad \blacksquare
\end{aligned}
$$

Optimization of server capacity: As stated above, in a commercial system, bulk queueing models are useful to determine the optimal capacity of the server. This is an involved topic in itself, and hence beyond the scope of this paper. Therefore, our discussion is very brief. The simplest optimization model that can be constructed in this context is one to optimize the capacity subject to a constraint on the downside risk:

$$\text{Minimize } Z \text{ such that } \mathsf{P}(X > K) < p_{\max}, \text{ where } K \in I^+ \text{ and } p_{\max} \in (0,1).$$

Since the downside risk can be evaluated for any given capacity using the models presented above, it is not very difficult to optimize the capacity of the server.

4 Computational Results

We now describe the results of our computational experiments. We begin with some notation.

Notation: Since, we have developed the models for numerous combinations of dispatching and stranding strategies for homogenous and non-homogenous Poisson arrivals, in order to increase readability, we have developed the following notation to define each case studied:

Dispatching strategy|Stranding Strategy|Nature of arrivals,

where R = regular dispatching, H = holding dispatching, h = homogeneous Poisson arrival, and nh = non-homogeneous Poisson arrival. The stranding strategy is denoted either by ∞, 0, or 1. For the homogeneous case, we used the uniform distribution (denoted by $U(a,b)$) for the service time, although we note that our model works in general for any distribution. For the non-homogeneous case, the time horizon over which arrivals can occur before a service occurs (i.e., the inter-arrival time) is divided into four intervals, and λ_k denotes the mean arrival rate in the kth interval. Let $\bar{L} = (L(1), \ldots, L(L_{\max}))$, $T_s(t) = \mathsf{P}(\text{Service Time } = \Psi t)$, and $\bar{T_s} = (T_s(1), \ldots, T_s(ST))$ where $L(l) = \mathsf{P}(L = l)$ and $L_{\max}$ denotes the maximum size of the customer.

Input parameters and results In order to test the validity of our models numerically, it is prudent to perform an extensive computational investigation over a variety of *pmfs* for the server capacity (ϕ). Accordingly, we have chosen twenty one different *pmfs* with the server capacity varying from 8 to 40. These distributions are described in Table 1.

Tables 2 and 3 denote the input parameters and the results, respectively, for the case of holding strategy with ∞-stranding and homogenous Poisson arrivals. We studied ten different cases described in Table 2. The deviation of the value produced by the Markov chain model from that produced by the simulation model is expressed in percentage as follows:

$$\text{Deviation (in \%)} \ = \frac{|Value_{MC} - Value_{SIM}|}{|Value_{SIM}|} \times 100$$

where $Value_M$ denotes the value obtained from a model M, and MC denotes the Markov chain model and SIM the simulation model. As is clear from the results in Table 3, the deviation of the Markov chain model is less than 3% in each case. Also, the Markov chain model takes a significantly shorter length of time on the computer. For the downside risk, we have not shown the deviation, because it was insignificant up to five places after the decimal point.

For the non-homogenous case, we used $\Psi = 1$ in all our computations. Assumption 1 and the north-west-corner properties were verified in all of these experiments. Results involving the holding strategy with ∞-stranding and other forms of stranding when the arrival process is nonhomogeneous Poisson, regular holding strategy with 1-stranding and homogenous arrivals, and some other computations related to regular holding can be found in the online supplement to this paper.

5 Conclusions

This paper presented Markov chain models to determine the distribution of the number stranded after the server leaves in a bulk-service, bulk-arrival queue. Studies of bulk queues continue to attract research attention because of their ubiquitous nature. However, the existing literature has focussed on the distribution of the customer waiting time in such queues, ignoring the distribution of the number stranded, which plays a role in determining whether server capacity is sufficient. Also, the literature has developed Laplace transform techniques which could potentially be useful in determining this distribution. In this paper, however, we used a discrete-time Markov chain and presented results which show that this technique elegantly scales up to the complexity of this problem without having to generate the complex zeroes required in Laplace transforms. We presented an extensive analysis of this kind of a queue under a variety of dispatching and stranding strategies. Finally, we performed an extensive numerical study to show the usefulness of our models and mathematical results. Potential future work could be related to retrial queues (Artalejo et al., 2007, 2008) in systems of the nature studied in this paper.

References

C. Armero and D. Conesa. Statistical performance of multi-class bulk production queueing system. *European Journal of Operational Research*, 158:649–661, 2004.

J.R. Artalejo, A. Corral-Gomez, and M.J. Lopez-Herrero. Algorithmic approximations for the busy period distributions of the M/M/c retrial queue. *European Journal of Operational Research*, 176: 1687–1702, 2007.

J.R. Artalejo, A. Economou, and A. Gómez-Corral. Algorithmic analysis of the Geo/Geo/c retrial queue. *European Journal of Operational Research*, 189:1042–1056, 2008.

N. Bailey. On queuing processes with bulk service. *J. Roy. Stat. Soc. B*, 16:80–87, 1954.

S.K. Bar-Lev, M. Parlar, D. Perry, W. Stadje, and F.A. Van der Duyn Schouten. Applications of bulk queues to group testing models with complete information. *European Journal of Operational Research*, 183:226–237, 2007.

A. Borthakur and J. Medhi. A queuing system with arrivals and departures in batches of variable size. *Cah. Cent. Etud. Rech. Opnl.*, 16:117–126, 1974.

S. Chang, D. Choi, and T-S. Kim. Performance analysis of a finite-buffer bulk-arrival and bulk-service queue with variable server capacity. *Stochastic Analysis and Applications*, 22(5):1151–1173, 2004.

S-P Chen. Parametric nonlinear programming approach to fuzzy queues with bulk service. *European Journal of Operational Research*, 162:434–444, 2005.

J. Cohen. The single server queue. *North-Holland, London*, 1969.

F. Downton. Waiting time in bulk service queues. *J. Roy. Stat. Soc. B0*, 17:256–261, 1986.

D. Gaver. Embedded Markov chain analysis of a waiting line process in continuous time. *Ann. Math. Stat.*, 30:698–720, 1959.

P. Ghare. Multichannel queuing system with bulk service. *Operations Research*, 16:189–192, 1968.

N. Jaiswal. A bulk-service queuing problem with variable capacity. *J. Roy. Stat. Soc.B*, 26:143–148, 1964.

E.P.C. Kao. *An introduction to stochastic processes.* Duxbury Press, Belmont, CA, USA, 1997.

J-C Ke and K-H Wang. A recursive method for the n policy g/m/1 queueing system with finite capacity. *European Journal of Operational Research*, 142:577–594, 2002.

J. Keilson. The general bulk queue as a Hilbert problem. *Applied Research Laboratory, Sylvania Electronic Systems, Waltha,, Mass*, 2:344–358, 1962.

J. Medhi. *Stochastic Models in Queueing Theory.* Academic Press, Amsterdam, second edition, 2003.

J Mejia-Tellez and David Worthington. Practical methods for queue length behavior for bulk service queues of the form $M/G^{0,C}/1$ and $M(t)/G^{0,C}/1$. *European Journal of Operational Research*, 73(1):103–113, 1994.

R. Miller. A contribution to the theory of bulk queues. *J. Roy. Stat. Soc.B*, 21:320–337, 1959.

M. Neuts. *Matrix-Geometric Solutions in Stochastic Models.* Johns Hopkins University Press, Baltimore, 1981.

A. G. Pakes. Some conditions for ergodicity and recurrence of Markov chains. *Operations Research*, 17:1058–1061, 1969.

S.M. Ross. *Introduction to Probability Models.* Academic Press, San Diego, CA, USA, 2003.

T. Saaty. Time dependent solution of the many server Poisson queue. *Operations Research*, 8: 755–772, 1960.

E. Seneta. *Non-negative matrices and Markov chains.* Springer, New York, NY, USA, 1981.

J. Teghem. Control of the service process in a queueing system. *European Journal of Operational Research*, 23(2):141–158, 1986.

U. Thonemann and M. Brandeau. Designing a single-vehicle automated guided vehicle system with multiple load capacity. *Transportation Science*, 30(4):351–363, 1996.

Table 1: Various patterns for ϕ.

	Z	$\mathsf{P}(D=0), \mathsf{P}(D=1), \ldots, \mathsf{P}(D=Z)$
ϕ_{11}	9	$\{0, 0, 0, 0.1, 0.1, 0.1, 0.1, 0.2, 0.2, 0.2\}$
ϕ_{12}	14	$\{0, 0, 0.05, 0.05, 0.05, 0.05, 0.05, 0.08, 0.08, 0.08, 0.08, 0.09, 0.09, 0.09, 0.16\}$
ϕ_{13}	12	$\{0, 0, 0, 0, 0, 0.08, 0.09, 0.12, 0.15, 0.19, 0.18, 0.1, 0.09\}$
ϕ_{14}	25	$\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.1, 0.3, 0.2, 0.2, 0.1, 0.05, 0.05\}$
ϕ_{15}	20	$\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.1, 0.3, 0.2, 0.2, 0.1, 0.1\}$
ϕ_{16}	15	$\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0.1, 0.1, 0.1, 0.1, 0.2, 0.2, 0.2\}$
ϕ_{17}	16	$\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0.05, 0.05, 0.1, 0.3, 0.25, 0.15, 0.05, 0.05\}$
ϕ_{18}	20	$\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.1, 0.1, 0.2, 0.2, 0.2, 0.1, 0.1\}$
ϕ_{19}	10	$\{0, 0, 0, 0, 0, 0.1, 0.3, 0.25, 0.25, 0.05, 0.05\}$
ϕ_{20}	25	$\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.3, 0.1, 0.2, 0.2, 0.1, 0.05, 0.05\}$
ϕ_{21}	30	$\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.3, 0.1, 0.2, 0.2, 0.1, 0.05, 0.05\}$
ϕ_{22}	40	$\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.1, 0.2, 0.2, 0.2, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05\}$
ϕ_{23}	25	$\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1\}$
ϕ_{24}	10	$\{0, 0, 0, 0, 0.1, 0.1, 0.1, 0.1, 0.2, 0.2, 0.2\}$
ϕ_{25}	12	$\{0, 0, 0, 0, 0, 0, 0.1, 0.1, 0.2, 0.2, 0.2, 0.1, 0.1\}$
ϕ_{26}	8	$\{0, 0, 0.05, 0.05, 0.1, 0.1, 0.1, 0.3, 0.3\}$
ϕ_{27}	11	$\{0, 0, 0, 0, 0, 0, 0, 0.1, 0.2, 0.2, 0.25, 0.25\}$
ϕ_{28}	20	$\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.05, 0.05, 0.05, 0.05, 0.1, 0.1, 0.1, 0.2, 0.2, 0.1\}$
ϕ_{29}	20	$\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.1, 0.1, 0.1, 0.1, 0.2, 0.2, 0.2\}$
ϕ_{30}	16	$\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.1, 0.15, 0.15, 0.2, 0.2, 0.1, 0.1\}$
ϕ_{31}	10	$\{0, 0, 0, 0, 0, 0, 0.1, 0.2, 0.3, 0.3, 0.1\}$

Table 2: System parameters for $H|\infty|h$. Here $U(a,b)$ denotes a uniform distribution with parameters a and b.

System	λ	$L_{\max}$, $\bar{L}$	ϕ	U(a,b)	B
1	0.8	5, (0.1,0.1,0.2,0.3,0.3)	ϕ_{16}	2,4	6
2	1.2	5, (0.1,0.1,0.2,0.3,0.3)	ϕ_{16}	1,3	6
3	1.2	6, (0,0.1,0.1,0.2,0.3,0.3)	ϕ_{16}	1,3	6
4	1.2	6, (0,0.1,0.1,0.2,0.3,0.3)	ϕ_{16}	1,3	12
5	1.1	6, (0,0.1,0.1,0.2,0.3,0.3)	ϕ_{16}	1,3	8
6	1.2	6, (0,0.1,0.1,0.2,0.3,0.3)	ϕ_{28}	2.1,2.5	10
7	1.2	6, (0,0.1,0.1,0.2,0.3,0.3)	ϕ_{28}	2.1,2.5	13
8	1.2	4, (0.1,0.1,0.5,0.3)	ϕ_{28}	2.1,2.5	13
9	1.5	4, (0.1,0.1,0.5,0.3)	ϕ_{28}	2.1,2.5	13
10	1.5	4, (0.1,0.1,0.5,0.3)	ϕ_{28}	2.8,3	15

Table 3: Performance parameters for $H|\infty|h$

System	$\mathsf{E}(X)$			$\mathsf{Var}(X)$			$\mathsf{P}(X > 5)$
	Markov chain	Simulation	Deviation in %	Markov Chain	Simulation	Deviation in %	
1	2.85213	2.85515	0.105773777	31.894472	31.918182	0.074283679	0.21766
2	3.209972	3.207042	0.091361448	38.742991	38.622138	0.312911212	0.236145
3	18.486579	18.510549	0.129493728	554.806878	555.340573	0.096102289	0.629807
4	19.396082	19.596766	1.02406693	556.55507	560.120996	0.636634946	0.656585
5	9.498974	9.542315	0.454197959	190.962767	191.166136	0.106383382	0.469216
6	5.395274	5.417871	0.417082651	85.424777	85.312782	0.131275757	0.331734
7	5.654459	5.712131	1.00964071	86.095061	86.177634	0.095817205	0.343167
8	0.53269	0.548185	2.826600509	2.732738	2.761575	1.044222952	0.032794
9	1.19049	1.215353	2.045743089	8.912699	9.043056	1.441514904	0.092643
10	4.135253	4.186771	1.230494813	47.93385	48.08815	0.320869071	0.287316

ONLINE SUPPLEMENT

Aykut Kahraman
M & T Bank, Buffalo, NY 14620
Abhijit Gosavi *
Missouri University of Science and Technology
219 Engineering Management, Rolla, MO 65401
Phone: (573)341-4624
Email: gosavia@mst.edu
Fax: (573)341-6567

February 3, 2011

*Corresponding author

Table 1: System parameters for $H|\infty|nh$.

System	$\lambda_{1,2,3,4}$	ST, $\bar{T}_s$	$L_{\max}$, $\bar{L}$	ϕ	B
1	3.5,2,3,2.5	3, (0.3,0.4,0.3)	4, (0.2,0.2,0.3,0.3)	ϕ_{29}	6
2	3.5,2,3,2.5	3, (0.3,0.4,0.3)	4, (0.2,0.2,0.3,0.3)	ϕ_{29}	14
3	3,2,3,2.5	3, (0.3,0.4,0.3)	6, (0,0.1,0.1,0.2,0.3,0.3)	ϕ_{29}	10
4	3,2,3,2.5	3, (0.3,0.4,0.3)	6, (0,0.1,0.1,0.2,0.3,0.3)	ϕ_{29}	20
5	1,2,1,2.5	3, (0.3,0.4,0.3)	5, (0.1,0.1,0.3,0.3,0.2)	ϕ_{30}	8
6	1,2,1,2.5	3, (0.5,0.4,0.1)	5, (0.1,0.1,0.3,0.3,0.2)	ϕ_{30}	8
7	4,1,1.5,2.5	3, (0.3,0.4,0.3)	4, (0.2,0.2,0.4,0.2)	ϕ_{28}	12
8	4,1,1.5,2.5	3, (0.3,0.4,0.3)	4, (0.2,0.2,0.4,0.2)	ϕ_{28}	16
9	4,1,3.5,2.5	3, (0.3,0.4,0.3)	3, (0.2,0.4,0.4)	ϕ_{28}	10
10	4,1,3.5,2.5	3, (0.3,0.4,0.3)	3, (0.2,0.4,0.4)	ϕ_{28}	15

Table 2: Performance parameters for $H|\infty|nh$.

	$\mathsf{E}(X)$			$\mathsf{Var}(X)$			$\mathsf{P}(X > 5)$
System	Markov chain	Simulation	Deviation in %	Markov Chain	Simulation	Deviation in %	
1	11.405022	11.457133	0.454834556	263.40109	264.254321	0.322882516	0.480214
2	11.518538	11.834147	2.66693493	263.59816	267.656048	1.516083059	0.483129
3	16.796941	17.317612	3.006598138	512.57867	528.747027	3.057862489	0.555509
4	18.548689	19.908419	6.829924566	522.826619	548.244631	4.636253702	0.592038
5	32.015368	32.689056	2.06089769	1345.601733	1384.918841	2.838946719	0.737113
6	5.841444	6.014764	2.881576068	86.273881	90.1262	4.274360841	0.319314
7	3.678307	3.779234	2.670567634	47.204557	47.796337	1.238128353	0.231251
8	3.969427	4.16795	4.76308497	47.629028	49.192915	3.179089916	0.238955
9	3.549695	3.613171	1.756794793	44.840221	45.102211	0.580880614	0.228175
10	3.724137	3.882396	4.076322972	45.073558	46.192856	2.423097632	0.232927

Table 3: System parameters for $H|0|nh$.

System	$\lambda_{1,2,3,4}$	ST, $\bar{T}_s$	$L_{\max}$, $\bar{L}$	ϕ	B
1	5,3,1,4	3, (0.3,0.4,0.3)	4, (0.2,0.2,0.3,0.3)	ϕ_{29}	8
2	5,3,1,4	3, (0.3,0.4,0.3)	4, (0.2,0.2,0.3,0.3)	ϕ_{29}	15
3	2,1,7,9	3, (0.3,0.4,0.3)	4, (0.2,0.2,0.3,0.3)	ϕ_{29}	15
4	4.3,5,2	3, (0.3,0.4,0.3)	3, (0.3,0.3,0.4)	ϕ_{29}	10
5	4.3,5,2	3, (0.3,0.4,0.3)	3, (0.3,0.3,0.4)	ϕ_{29}	16
6	4.3,5,2	3, (0.3,0.4,0.3)	3, (0.3,0.3,0.4)	ϕ_{29}	5
7	1,3,4,2	3, (0.3,0.4,0.3)	3, (0.3,0.3,0.4)	ϕ_{31}	5
8	1,3,4,2	3, (0.3,0.4,0.3)	3, (0.3,0.3,0.4)	ϕ_{31}	7
9	2,5,3,1	3, (0.4,0.4,0.2)	3, (0.7,0.2,0.1)	ϕ_{31}	7
10	2,5,3,1	3, (0.4,0.4,0.2)	3, (0.7,0.2,0.1)	ϕ_{31}	9

Table 4: Performance parameters for $H|0|nh$.

System	$\mathsf{E}(X)$			$\mathsf{Var}(X)$			$\mathsf{P}(X>5)$
	Markov chain	Simulation	Deviation in %	Markov Chain	Simulation	Deviation in %	
1	4.495751	4.481804	0.311191654	50.82339	50.395535	0.848993864	0.294938
2	4.607165	4.626356	0.41481892	50.023035	49.295476	1.475914342	0.294151
3	11.669102	12.281352	4.985200326	179.657423	180.959205	0.719378713	0.529258
4	2.345025	2.363675	0.789025564	21.47938	21.215161	1.245425382	0.170225
5	2.467095	2.543642	3.009346441	21.074645	21.242542	0.790380925	0.170225
6	2.345025	2.3257	0.830932622	21.47938	20.861082	2.963882698	0.170225
7	4.119728	4.142097	0.540040467	30.041506	29.145071	3.075768798	0.306965
8	4.201805	4.261275	1.395591695	29.477994	28.598176	3.07648292	0.306965
9	1.686065	1.686004	0.003618022	9.226588	8.977472	2.774901442	0.116786
10	1.812272	1.822315	0.551112184	8.965558	8.73023	2.695553267	0.116786

Table 5: System parameters for $H|0|h$.

System	λ	L_{max}, L	ϕ	U(a,b)	B
1	2.5	5, (0.1,0.1,0.2,0.3,0.3)	ϕ_{16}	3,4	8
2	2.5	5, (0.1,0.1,0.2,0.3,0.3)	ϕ_{16}	3,4	13
3	6.5	5, (0.1,0.1,0.2,0.3,0.3)	ϕ_{16}	1.8,2	13
4	3.5	4, (0.1,0.1,0.5,0.3)	ϕ_{16}	1.8,2	10
5	3.5	4, (0.1,0.1,0.5,0.3)	ϕ_{16}	1.8,2	15
6	3	4, (0.1,0.1,0.5,0.3)	ϕ_{16}	2.4,3.2	6
7	3	4, (0.1,0.1,0.5,0.3)	ϕ_{16}	2.4,3.2	11
8	3.2	4, (0.1,0.1,0.5,0.3)	ϕ_{28}	2.4,3.2	11
9	3.2	4, (0.1,0.1,0.5,0.3)	ϕ_{28}	3.9,4.1	12
10	3.2	4, (0.1,0.1,0.5,0.3)	ϕ_{28}	3.9,4.1	18

Table 6: Performance parameters for $H|0|h$.

System	$\mathsf{E}(X)$			$\mathsf{Var}(X)$			$\mathsf{P}(X>5)$
	Markov chain	Simulation	Deviation in %	Markov Chain	Simulation	Deviation in %	
1	19.031039	19.038386	0.038590456	132.507161	132.571047	0.048190009	0.897352
2	19.055288	19.066944	0.061131978	131.645493	131.266976	0.288356608	0.897352
3	31.802778	31.870677	0.213045365	181.929295	185.102881	1.714498436	0.987104
4	8.087027	8.095265	0.101763191	53.710772	53.614221	0.180084683	0.610311
5	8.248227	8.29712	0.589276761	51.413858	50.855585	1.097761436	0.610311
6	12.869171	12.872472	0.02564387	82.148042	82.14927	0.00149484	0.79744
7	12.881389	12.887331	0.046107297	81.856774	81.780276	0.093540892	0.79744
8	10.820414	10.825239	0.044571764	81.496422	81.488641	0.00954857	0.700387
9	21.760903	21.751715	0.042240347	129.155606	129.30493	0.115482062	0.941763
10	1.771145	21.751715	0.08932629	128.729827	129.30493	0.444764944	0.941763

Table 7: System parameters for $R|1|h$.

System	λ	$L_{\max}$, L	ϕ	U(a,b)
1	3	3, (0.2,0.4,0.4)	ϕ_{16}	2,4
2	2	3, (0.3,0.4,0.3)	ϕ_{16}	2.4
3	2.8	3, (0.3,0.4,0.3)	ϕ_{18}	3,5
4	1.5	3, (0.1,0.3,0.6)	ϕ_{17}	3,5
5	1.5	3, (0.1,0.3,0.6)	ϕ_{17}	2,6
6	1.7	4, (0.1,0.1,0.5,0.3)	ϕ_{17}	2,6
7	2.4	3, (0.2,0.3,0.5)	ϕ_{17}	3.8,4.2
8	1.3	3, (0.2,0.3,0.5)	ϕ_{19}	2.7,3.3
9	2.1	3, (0.2,0.3,0.5)	ϕ_{19}	2.7,3.3
10	1.1	3, (0.2,0.3,0.5)	ϕ_{19}	4.9,5.1

Table 8: Performance parameters for the number stranded for $R|1|h$.

	$\mathsf{E}(X)$			$\mathsf{Var}(X)$			$\mathsf{P}(X > 5)$
System	Markov chain	Simulation	Deviation in %	Markov Chain	Simulation	Deviation in %	
1	14.10566	14.128283	0.160125615	61.247546	61.472835	0.366485457	0.887688
2	5.690404	5.690848	0.007802001	37.197586	37.183736	0.037247468	0.478011
3	20.299113	20.39521	0.471174359	72.176467	73.84217	2.255761173	0.968452
4	11.661313	11.655966	0.045873504	57.054576	57.241558	0.326654281	0.807972
5	11.251906	11.274114	0.19698222	69.040913	69.665638	0.896747691	0.756556
6	18.80123	18.86536	0.339935204	106.804546	108.904963	1.928669679	0.926993
7	21.689144	21.721046	0.732198146	58.134926	58.946585	1.376939818	0.99446
8	7.134525	7.134343	0.002551041	26.383153	26.384551	0.005298555	0.660551
9	14.186164	14.187568	0.009895988	38.680524	38.845421	0.424495335	0.955618
10	12.112389	12.118175	0.047746463	34.165844	34.193498	0.080875025	0.914335

Table 9: System parameters for $R|1|nh$.

System	$\lambda_{1,2,3,4}$	ST, T_s	$L_{\max}$, L	ϕ
1	1.2,1.8,2.1,1.1	3, (0.4,0.5,0.1)	3, (0.3,0.5,0.2)	ϕ_{24}
2	0.8,2.5,2,1.3	3, (0.5,0.4,0.1)	3, (0.3,0.4,0.3)	ϕ_{24}
3	1.8,3.1,2.6,1.9	3, (0.5,0.4,0.1)	3, (0.3,0.5,0.2)	ϕ_{24}
4	2.1,1.8,5.1,0.6	3, (0.5,0.4,0.1)	3, (0.3,0.5,0.2)	ϕ_{25}
5	1,1.5,3,1.5	3, (0.6,0.2,0.2)	3, (0.3,0.5,0.2)	ϕ_{26}
6	1.8,1.1,1.9.1.2	3, (0.6,0.2,0.2)	3, (0.3,0.5,0.2)	ϕ_{26}
7	2.1,1.1,3,0	3, (0.2,0.5,0.3)	3, (0.3,0.5,0.2)	ϕ_{26}
8	2,1.5,1,1.5	3, (0.2,0.5,0.3)	3, (0.3,0.4,0.3)	ϕ_{27}

Table 10: Performance parameters for the number stranded for $R|1|nh$.

	$\mathsf{E}(X)$			$\mathsf{Var}(X)$			$\mathsf{P}(X > 5)$
System	Markov chain	Simulation	Deviation in %	Markov Chain	Simulation	Deviation in %	
1	1.338446	1.342466	0.299448925	6.993925	7.077183	1.176428531	0.116578
2	1.428162	1.433461	0.369664748	7.79405	7.931625	1.734512159	0.125493
3	3.729076	3.73749	0.225124348	21.574447	21.694233	0.552155958	0.337544
4	2.935519	2.946359	0.367911717	20.123144	20.397506	1.345076207	0.258947
5	3.430557	3.432075	0.044229803	19.470659	19.558191	0.447546504	0.306968
6	1.740234	1.743122	0.16567974	9.159452	9.264735	1.136384365	0.153051
7	3.506154	3.512323	0.175638744	17.31922	17.410772	0.525835385	0.329247
8	2.7323597	2.756014	0.858304784	15.645689	15.896856	1.579979085	0.215685

Table 11: System parameters for $R|\infty|h$.

System	λ	L_{max}, $\bar{L}$	ϕ	U(a,b)
1	0.8	3, (0.2,0.4,0.4)	ϕ_{11}	2,4
2	0.95	3, (0.2,0.4,0.4)	ϕ_{11}	2.4
3	0.6	3, (0.2,0.4,0.4)	ϕ_{11}	3,5
4	0.6	3, (0.3,0.3,0.4)	ϕ_{12}	4,6
5	1.5	3, (0.3,0.3,0.4)	ϕ_{12}	2,3
6	0.9	3, (0.3,0.3,0.4)	ϕ_{12}	3,6
7	0.6	3, (0.1,0.2,0.7)	ϕ_{12}	4.6
8	0.5	5, (0,0.1,0.1,0.5,0.3)	ϕ_{13}	3,5
9	0.65	5, (0,0.1,0.1,0.5,0.3)	ϕ_{13}	1,3
10	0.8	4, (0,0,0.5,0.5)	ϕ_{13}	1,3

Table 12: Performance parameters for $R|\infty|h$.

	$\mathsf{E}(X)$			$\mathsf{Var}(X)$			$\mathsf{P}(X > 5)$
System	Markov chain	Simulation	Deviation in %	Markov Chain	Simulation	Deviation in %	
1	4.927	4.925521	0.03002728	51.552025	51.625358	0.14204841	0.362064
2	29.167278	28.966603	0.692780579	1029.820651	1003.254301	2.648017554	0.786818
3	4.771326	4.802577	0.650713148	48.642077	49.224674	1.18354669	0.354974
4	2.843547	2.842245	0.045808859	25.748025	25.776266	0.109562029	0.231795
5	10.007151	10.015923	0.087580545	171.186114	171.338288	0.088814941	0.532383
6	23.73356	23.757196	0.099489856	746.152106	754.593818	1.118709403	0.730283
7	10.119194	10.102449	0.165751888	177.285295	178.349601	0.596752667	0.530825
8	26.206271	25.98545	0.849787092	912.014622	895.99532	1.787877865	0.739774
9	2.16185	2.162237	0.017898131	20.220988	20.204012	0.084022916	0.175514
10	2.413671	2.415984	0.095737389	22.286747	22.318914	0.144124396	0.196836

Table 13: System parameters for $R|0|h$.

System	λ	L_{max}, $\bar{L}$	ϕ	U(a,b)
1	0.8	3, (0.2,0.4,0.4)	ϕ_{11}	2,4
2	1.1	3, (0.2,0.4,0.4)	ϕ_{11}	1,5
3	1.8	5, (0,0.1,0.1,0.5,0.3)	ϕ_{11}	2,6
4	0.6	5, (0,0.1,0.1,0.5,0.3)	ϕ_{12}	4,6
5	0.7	5, (0,0.1,0.2,0.3,0.2),0.2	ϕ_{12}	1,5
6	1.5	5, (0.1,0.2,0.3,0.2,0.2)	ϕ_{12}	1,3
7	3	5, (0.1,0.2,0.3,0.2,0.2)	ϕ_{14}	2,4
8	1.3	5, (0.1,0.2,0.3,0.2,0.2)	ϕ_{14}	4,7
9	1.6	6, (0,0.1,0.25,0.25,0.3,0.1)	ϕ_{14}	4,7
10	2.2	6, (0,0.1,0.25,0.25,0.3,0.1)	ϕ_{15}	4,6

Table 14: Performance parameters for $R|0|h$

System	E(X)			Var(X)			P($X > 5$)
	Markov chain	Simulation	Deviation in %	Markov Chain	Simulation	Deviation in %	
1	1.14523	1.14542	0.016587802	5.314004	5.317751	0.070462118	0.094266
2	2.466295	2.4766	0.416094646	14.218175	14.405369	1.299473828	0.22457
3	22.256316	22.281209	0.111721945	187.818136	189.604431	0.942116696	0.912308
4	4.743565	4.749437	0.123635707	37.029403	37.101037	0.193078161	0.389528
5	1.706088	1.706074	0.000820597	12.660871	12.708793	0.377077508	0.150163
6	3.190777	3.190396	0.011942091	25.073607	25.0659	0.030746951	0.272478
7	9.203577	9.207623	0.043941851	96.373744	96.347795	0.026932635	0.570788
8	4.79256	4.785072	0.156486674	47.649846	47.555008	0.199427997	0.359695
9	15.155145	15.189051	0.223226586	158.503343	159.893038	0.869140406	0.750442
10	27.251937	27.442193	0.693297361	210.761449	219.469518	3.967780619	0.949418

Table 15: System parameters for $R|\infty|nh$.

System	T	$\lambda_{1,2,3,4}$	ST, T_s	L_{max}, L	ϕ
1	4	1,4,2,5	3, (0.3,0.4,0.3)	3, (0.2,0.4,0.4)	ϕ_{20}
2	4	3,5,3,8	3, (0.3,0.4,0.3)	3, (0.4,0.5,0.1)	ϕ_{20}
3	4	1,3,2,2	3, (0.3,0.4,0.3)	4, (0.2,0.3,0.3,0.2)	ϕ_{20}
4	4	2,6,5,4	3, (0.4,0.4,0.2)	4, (0.2,0.3,0.3,0.2)	ϕ_{21}
5	4	3,7,4,6	3, (0.4,0.4,0.2)	4, (0.2,0.3,0.3,0.2)	ϕ_{21}
6	4	4,3,2,6	3, (0.4,0.4,0.2)	5, (0.4,0.1,0.2,0.2,0.1)	ϕ_{21}
7	4	5,4,1,7	3, (0.4,0.4,0.2)	5, (0.4,0.1,0.2,0.2,0.1)	ϕ_{21}
8	4	3,7,2,10	3, (0.4,0.4,0.2)	5, (0.4,0.1,0.2,0.2,0.1)	ϕ_{22}
9	4	2,9,4,8	3, (0.4,0.4,0.2)	5, (0.4,0.1,0.2,0.2,0.1)	ϕ_{22}
10	4	2,9,4,8	3, (0.3,0.4,0.3)	4, (0.2,0.3,0.3,0.2)	ϕ_{22}

Table 16: Performance parameters for $R|\infty|nh$.

System	E(X)			Var(X)			P($X > 5$)
	Markov chain	Simulation	Deviation in %	Markov Chain	Simulation	Deviation in %	
1	1.489237	1.489863	0.042017286	15.628299	15.625457	0.018188268	0.10448
2	4.378734	4.385148	0.146266443	60.895978	61.467172	0.929266764	0.268625
3	0.40056	0.399002	0.390474233	3.251019	3.250689	0.010151694	0.027553
4	5.102523	5.114937	0.242700936	93.38255	93.23185	0.16164004	0.273086
5	17.045342	17.183964	0.806693962	555.590373	567.625038	2.120178673	0.541502
6	2.750245	2.753933	0.133917564	42.22657	42.023644	0.482885301	0.168573
7	5.895933	5.899954	0.068153074	115.670372	115.472645	0.171232763	0.29856
8	5.875185	5.882264	0.12034482	131.856832	132.487782	0.476232593	0.275581
9	7.570653	7.605456	0.457605698	188.723621	190.434858	0.898594416	0.324423
10	14.147141	14.059614	0.622541984	432.781127	426.85313	1.388767373	0.480358

Table 17: System parameters for $R|0|nh$.

System	$\lambda_{1,2,3,4}$	ST, T_s	L_{max}, L	ϕ
1	8,4,13,0	3, (0.3,0.4,0.3)	3, (0.1,0.4,0.5)	ϕ_{23}
2	3,5,3,8	3, (0.3,0.4,0.3)	3, (0.1,0.4,0.5)	ϕ_{20}
3	1,3,2,2	3, (0.3,0.4,0.3)	4, (0.2,0.3,0.3,0.2)	ϕ_{20}
4	2,6,5,4	3, (0.4,0.4,0.2)	4, (0.2,0.3,0.3,0.2)	ϕ_{21}
5	3,7,4,6	3, (0.4,0.4,0.2)	4, (0.2,0.3,0.3,0.2)	ϕ_{21}
6	9,5,12,3	3, (0.4,0.4,0.2)	5, (0.4,0.1,0.2,0.2,0.1)	ϕ_{21}
7	5,2,11,7	3, (0.4,0.4,0.2)	5, (0.4,0.1,0.2,0.2,0.1)	ϕ_{21}
8	3,7,2,10	3, (0.4,0.4,0.2)	5, (0.4,0.1,0.2,0.2,0.1)	ϕ_{22}
9	5,12,6,9	3, (0.3,0.4,0.3)	5, (0.4,0.1,0.2,0.2,0.1)	ϕ_{22}
10	2,9,4,8	3, (0.3,0.4,0.3)	4, (0.2,0.3,0.3,0.2)	ϕ_{22}

Table 18: Performance parameters for $R|0|nh$.

	$\mathsf{E}(X)$			$\mathsf{Var}(X)$			$\mathsf{P}(X > 5)$
System	Markov chain	Simulation	Deviation in %	Markov Chain	Simulation	Deviation in %	
1	12.756387	12.747199	0.07207858	200.709964	200.432454	0.138455622	0.571072
2	5.995404	5.998659	0.054262128	73.326584	73.415495	0.121106587	0.363651
3	0.329454	0.328084	0.417575987	2.505785	2.499718	0.242707377	0.022374
4	2.312325	2.314167	0.079596676	29.263576	29.225208	0.131283924	0.154039
5	3.75131	3.753925	0.069660422	51.830118	51.879282	0.094766153	0.231482
6	11.059622	11.067367	0.069980511	197.203253	197.663121	0.232652403	0.496192
7	8.188784	8.183437	0.065339294	146.386549	146.347673	0.026564139	0.390419
8	2.824246	2.825207	0.034015207	47.215069	47.202654	0.026301487	0.163701
9	11.364887	11.373046	0.071739796	213.618866	213.912856	0.13743447	0.482915
10	4.068633	4.067844	0.019396024	65.939404	65.873396	0.100204337	0.23062