Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2011, Article ID 702834, 18 pages
doi:10.1155/2011/702834

Research Article

Upper Bounds on Performance Measures of Heterogeneous $M/M/c$ Queues

F. S. Q. Alves,[1] H. C. Yehia,[1] L. A. C. Pedrosa,[2] F. R. B. Cruz,[3] and Laoucine Kerbache[4]

[1] *Centro de Estudos da Fala, Acústica, Linguagem e músicA, Departamento de Engenharia Eletrônica, Universidade Federal de Minas Gerais, 31270-901 Belo Horizonte, MG, Brazil*
[2] *Fundação Dom Cabral, 30140-083 Belo Horizonte, MG, Brazil*
[3] *Departamento de Estatística, Universidade Federal de Minas Gerais, 31270-901 Belo Horizonte, MG, Brazil*
[4] *Department of OMIT and the Research Center GREGHEC, HEC School of Management, 78351 Paris, France*

Correspondence should be addressed to F. R. B. Cruz, fcruz@est.ufmg.br

Received 22 February 2011; Accepted 11 May 2011

Academic Editor: Ben T. Nohara

In many real-life queueing systems, the servers are often heterogeneous, namely they work at different rates. This paper provides a simple method to compute tight upper bounds on two important performance measures of single-class heterogeneous multi-server Markovian queueing systems, namely the average number in queue and the average waiting time in queue. This method is based on an expansion of the state space that is followed by an approximate reduction of the state space, only considering the most probable states. In most cases tested, we were able to approximate the actual behavior of the system with smaller errors than those obtained from traditional homogeneous multiserver Markovian queues, as shown by GPSS simulations. In addition, we have correlated the quality of the approximation with the degree of heterogeneity of the system, which was evaluated using its Gini index. Finally, we have shown that the bounds are robust and still useful, even considering quite different allocation strategies. A large number of simulation results show the accuracy of the proposed method that is better than that of classical homogeneous multiserver Markovian formulae in many situations.

1. Introduction

A better understanding of queueing systems is of paramount importance in order to improve their applications, which is the scope of the current study. Markov and semi-Markov processes are among the stochastic-based methods traditionally used for performance evaluation

of multistate systems [1]. In this paper, we have developed a formulation to model a particular type of queueing system, namely, heterogeneous multiserver single queues, seen in Figure 1. Using Kendall notation, we will focus on $M/M_i/c$ queues in this work. For such queues, there is only one class of jobs that arrive to the system accordingly to a Poisson process at a rate λ. The number of parallel servers is c, and the amount of time dedicated to each job on each server is exponentially distributed, with a rate μ_i, where $i = 1, 2, \ldots, c$. The *queue discipline* is first come first served (FCFS), which means that the jobs arriving first to the system will be served first. Concerning the *server allocation*, one possibility is that the first job in the queue will be allocated as soon as any server becomes idle, but there are others. Indeed, there are three different allocation strategies among the most popular that will be investigated in this paper, namely, (i) the fastest server first (FSF) allocation, for which the fastest available server is always allocated first, (ii) the randomly chosen server (RCS) allocation, for which the next job in the queue is randomly sent to any one of the servers that is idle, and (iii) the slowest server first (SSF) allocation, where the job is always first allocated to the slowest free server.

The aims of this paper are twofold. First, we will derive upper bounds on performance measures of single-class heterogeneous multi-server Markovian queueing systems see Figure 1. We will show that these bounds are easy to compute and accurate, which makes them a useful general approximation of the performance measures of these systems. Second, we will investigate the role of three different types of allocation strategies, namely, the FSF, the RCS, and the SSF allocations. As we will show in more detail in the following, the type of allocation strategy that is used for heterogeneous servers will strongly determine the performance measures of the system and hence its modeling.

The paper is organized as follows. In Section 2, we present some results that have been obtained in previous studies. In Section 3, we thoroughly describe our method that is based on the equilibrium equations of queueing systems. In Section 4, we focus on the model used in the simulations and developed to validate the proposed approximation. In Section 5, we present the experimental results. Finally, we draw some conclusions in Section 6 and give some final remarks.

2. Previous Work

Queues are ordinary phenomena that happen all the time. They can be encountered everywhere. Everyone has already joined a queue at least once, for instance, when driving home and lining up in a traffic jam, when paying bills or when buying a snack. Day-to-day queues are very frequent, and they are not even perceived in some cases. Queues happen, for example, throughout manufacturing processes [2–4], in airports, ports, and products distribution systems [5], or in computer and communication systems [6–8]. Queues may cause the quality of the services or the prices of the goods to rise or fall, depending on the efficiency of the distribution and logistics [9]. Thus, organizing queueing systems in order to decrease the line length can be a way to reduce costs and maximize the efficiency of a system.

As mentioned earlier, the focus here is on heterogeneous multi-server single queues. The importance of modeling such systems comes from their similarities with real-life systems. Indeed, many real situations involve servers working at different rates. To illustrate such a situation, let us consider manual assembly, in which human beings can be seen as servers. As noticed by Wang et al. [10], manual assembly is, by definition, carried out by workers, and

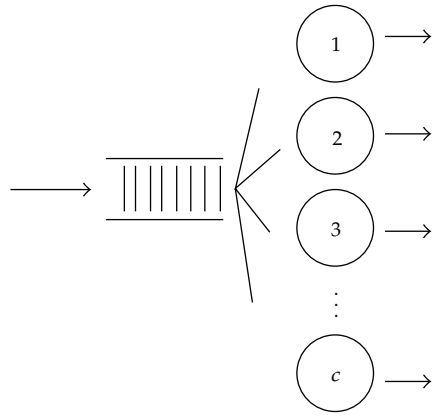

Figure 1: A single-class heterogeneous multiserver queue.

therefore, the system is human-centered and its performance largely depends on humans. In practice it is possible that even if all tasks are equal, the average task completion time may differ from person to person (as it is impossible for each worker to have equal efficiency) [10]. Thus, the individuals may be considered as heterogeneous servers. Considering another example of a real situation, among machines undergoing a process of rapid and constant technological renewal and depreciation, older equipments usually become slower than their latest counterparts, which naturally leads to service heterogeneities. As a final example, we can cite the transportation of goods by trucks with different powers and capacities, leading to heterogeneities in the servers. As a result, myriad applications can be found for heterogeneous multi-server single queues that can be used to gain more insight into these systems, and thus make them more manageable.

In the literature, there exist several studies that have taken into account the heterogeneity of the servers. One of the first studies that were developed for queueing systems considering the differences in the processing capacity of the servers was done by Gumbel [11] back in the 1960s. In his work, a Poisson distribution was used to model the arrivals, and the processing times of the servers were exponentially distributed with different rates for each server. Assuming steady-state conditions, Gumbel gave expressions in closed form for the state probabilities and for the expected length of the queue. Also, he analyzed the error resulting from the assumption that all the processing rates of each server were equal. However, Gumbel only considered the random allocation strategy.

Singh [12] analyzed a $M/M_i/3$ queueing system, composed of three heterogeneous servers. For a given system utilization, ρ, defined as

$$\rho = \frac{\lambda}{\mu_1 + \mu_2 + \mu_3}, \tag{2.1}$$

they showed that there is an optimal combination of servicing rates μ_1, μ_2, and μ_3 to minimize the performance measures of that system. This paper was a followup of a previous paper also by Singh [13], in which they studied a $M/M_i/2$ queueing system with balking.

Gall [14] generalized a factorization method [15] previously used to the case of a $G/G/s$ queueing system with heterogeneous servers. Gall presented three properties, which

allowed the construction of a numerical method to calculate the queue delay. A comparison of the results found using a factorization and a Markovian method in the case of a symmetrical $M/G/s$ system were presented and compared. Gall also compared the average queueing delay to simulation results.

Grassmann and Zhao [16] analyzed a queueing system with heterogeneous severs and general inputs. They showed how to find steady state probabilities for such a system and used different rules to allocate the arriving job presenting heuristic arguments. They used numerical calculations to support their assumption that the influence of the allocation strategy in the state probabilities increases with decreasing traffic intensities. As we will show in this paper, we could also validate this assumption using numerical simulations.

Boxma et al. [17] studied a $M/G_i/2$ queueing system with heterogeneous servers. The first server was exponentially distributed, and the second one was generally distributed. However, they were restricted to only two servers, while in this paper, a general number of c servers is treated. Additionally, Boxma et al. [17] did not investigate different server allocations (e.g., FSF, RCS, and SSF, as done here), as they only considered that when a customer arrives and there is no other customer in the system, the customer receives service from the first server immediately.

Chao and Luh [18] analyzed a finite $M/M/c/N$ queueing system, namely, with c parallel servers and a limited capacity of N jobs, including those in service. They studied the probability of an arriving job to find the system full. Also, finite queueing network systems with heterogeneous servers in series topologies were studied. Chiang et al. [19] focused on open networks of queues with heterogeneous exponential servers, while Biller et al. [20] studied closed networks of heterogeneous Bernoulli queues.

Marmony [21] considered a large-scale queueing system with only one type of job and multiple sever pools and proposed a routing policy assigning the jobs in accordance to the FSF allocation strategy in order to minimize the steady-state queue length. Marmony showed that the heterogeneous servers system was better than its homogeneous counterpart in the quality and efficiency driven regime when the FSF allocation strategy was applied, namely, the Halfin-Whitt many-server heavy traffic regime.

The multiclass multi-server (MCMS) system is a more complex model of a queueing system. The MCMS system presents different types of jobs arriving in a system with multiple servers. If we consider the servers to be heterogeneous, the MCMS system constitutes a generalization of the queueing system of interest in this paper, which only assumes one type of job. Van Harten and Sleptchenko [22] studied such a generalized model and even proposed an exact solution with a specific structure for the MCMS system, which can be reduced in order to give the eigenvalues and eigenvectors of a finite-dimensional matrix. Harten and Sleptchenko defined some sets of multiplicative eigenmodes creating approximations to find the performance measures of the system. Although Harten and Sleptchenko claim that the proposed structure could aggregate the nonidentical servers effects, their approximation was developed considering only equal servers due to the high level of complexity that the heterogeneity of the servers adds to the model.

Finally, it is also worthwhile mentioning that there are a number of papers in the literature focusing on controlling heterogeneous server queues using different allocation strategies in order to optimize their performance measures as shown in the work of Lin and Kumar [23], Koole [24], Walrand [25], Rykov and Efrosinin [26], Shenker and Weinrib [27, 28], and Cruz et al. [29]. In this paper, we show using numerical simulations that the average queue waiting time is the lowest using the FSF allocation strategy compared to the SSF and RCS strategies.

3. Mathematical Formulation

3.1. Preliminaries

The formulation proposed in this paper is fundamentally based on the equilibrium equations of the system that are obtained from the conservation of flow and some approximations. For our convenience and without loss of generality, the indices i of the processing rates μ_i can be rearranged as follows:

$$\mu_1 \leq \mu_2 \leq \cdots \leq \mu_c, \tag{3.1}$$

where μ_1 represents the processing rate of the slowest server in the system and μ_2 is the processing rate of the second slowest server, and so on till μ_c, which is the processing rate of the fastest server.

Considering heavy traffic conditions, it is intuitive that if there is a job in the system, it will most likely be on the slower server. Thus, assuming heavy traffic conditions the probability to find a job in the system in the slowest server is higher than the probability to find it in any other server, because in average, the job will stay longer in a slow server than in a fast one. Another argument that supports the statement that a job in the system is highly probable to be found in the slowest server is the fact that the exponential distribution has no memory. Also, in order to know which server is more likely to be the last one to finish the work at any time t and when all c servers are busy, one must consider that the probability of being the last one does not depend on the knowledge of which server started the work first because of the lack of memory of the exponential distribution. This probability only depends on the service rate μ, regardless of which server first started the work. Thus, we can define an *approximation* of the state diagram of $M/M_i/c$ systems, shown in Figure 2.

Figure 2 represents a birth and death process, in which the birth rate is such that $\lambda_i = \lambda$, for $i = 0, 1, 2, \ldots, \infty$ and the death rate μ_{eq_i} is variable and depends on the state i in which the system currently is. The state space is the set of nonnegative integers (the numbers inside the circles), which represents the number of jobs currently in the system, i. Then, the quantity μ_{eq_i} may be defined as an equivalent *approximate* death rate as

$$\mu_{\text{eq}_i} = \begin{cases} 0, & \text{if } i = 0, \\ \mu_1, & \text{if } i = 1, \\ \mu_1 + \mu_2, & \text{if } i = 2, \\ \vdots & \vdots \\ \mu_1 + \mu_2 + \cdots + \mu_c, & \text{if } i \geq c. \end{cases} \tag{3.2}$$

Indeed, under the assumptions of (3.1), if the system is in state "1" (i.e., there is currently only 1 single job in the system), server "1" (with service rate μ_1) clearly will be more probable to be busy than server "2" (with service rate $\mu_2 > \mu_1$) and so on. We argue that this is a worst-case *approximation* that can be made. Thus, the system may now be modeled using only the most probable possibilities regardless of the others. Such a simplification will allow us to easily compute accurate *upper bounds* for two important performance measures of

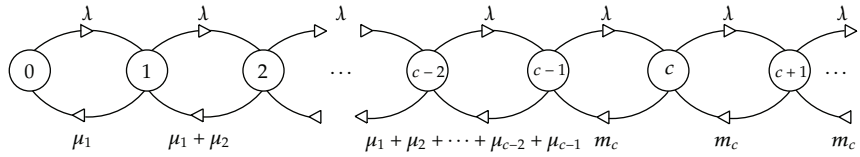

Figure 2: State transition diagram for $M/M_i/c$ systems.

the system. The upper bound is a consequence of the fact that the system is preferentially in the worst configuration, namely, when the jobs are in the slowest servers.

Thus, (3.2) can be formulated in two ways: when the system contains less than c jobs, μ_{eq_i} is variable and when there is c or more jobs in the system, μ_{eq_i} is a constant. Then, we define the quantities m_i and m_c in order to facilitate the development and visualization of the formulation as follows: if $i < c$,

$$m_i = \sum_{j=1}^{i} \mu_j, \tag{3.3}$$

and if $i \geq c$

$$m_i = m_c = \sum_{j=1}^{c} \mu_j. \tag{3.4}$$

When $i = 1$, m_i refers to μ_1, which actually is an *approximate* system death rate when it is in the state "1" (i.e., only 1 job is presently in the system). The system is in this state when there is only one job in the system and there is only one server processing the work (which we assume with high probability to be the slowest server).

For a better understanding of the influence of the μ parameters on the system, equilibrium equations for the diagram shown in Figure 2 can be written as

$$\begin{aligned} \mu_1 p_1 = p_0 \lambda \quad &\Longleftrightarrow \quad p_1 = \frac{\lambda}{\mu_1} p_0, \\ \lambda p_0 + (\mu_1 + \mu_2) p_2 = (\mu_1 + \lambda) p_1 \quad &\Longleftrightarrow \quad p_2 = \frac{\lambda^2}{\mu_1 (\mu_1 + \mu_2)} p_0, \\ &\vdots \end{aligned} \tag{3.5}$$

which leads to

$$p_c = \frac{\lambda^c}{\mu_1 (\mu_1 + \mu_2)(\mu_1 + \mu_2 + \mu_3) \cdots (m_c)} p_0. \tag{3.6}$$

In (3.6), μ_1 appears in all the terms of the denominator. It is repeated c times and has thus a great influence on the result. Similarly, the rate μ_2 has the second largest weight on the result and so on until μ_c, which only appears once in the last term of the denominator of (3.6).

We will show in the next section that these upper bounds can also be used as a good approximation of the performance measures of the system.

3.2. Computation of p_i and p_0

Developing the equilibrium equations, it can be shown that the probability p_i to find i jobs in the system is as follows: if $i < c$, then

$$p_i = p_0 \frac{\lambda^i}{\prod_{k=1}^{i}\left(\sum_{j=1}^{k}\mu_j\right)}, \tag{3.7}$$

and if $i \geq c$, one has

$$p_i = p_0 \prod_{j=1}^{c}\left(\frac{\lambda}{m_j}\right)\left[\prod_{j=c+1}^{i}\left(\frac{\lambda}{m_c}\right)\right]. \tag{3.8}$$

In order to compute the probability p_0 of the empty system, we isolate this term in (3.8). Using the fact that all probabilities p_i must obey the following relationship:

$$\sum_{i=0}^{\infty} p_i = 1, \tag{3.9}$$

an expression for p_0 can be found and

$$\begin{aligned} p_0^{-1} &= \sum_{i=0}^{c-1}\left[\frac{\lambda^i}{\prod_{j=1}^{i} m_j}\right] + \sum_{i=c}^{\infty}\left[\frac{\lambda^i}{\left(\prod_{j=1}^{c} m_j\right)(m_c)^{i-c}}\right] \\ &= \sum_{i=0}^{c-1}\left[\frac{\lambda^i}{\prod_{j=1}^{i} m_j}\right] + \frac{(m_c)^c}{\left(\prod_{j=1}^{c} m_j\right)}\sum_{i=c}^{\infty}\left(\frac{\lambda}{m_c}\right)^i. \end{aligned} \tag{3.10}$$

In order to keep the system stable (in other words, in order to prevent the queue from growing indefinitely), the system utilization

$$\rho = \frac{\lambda}{\prod_{j=1}^{c}\mu_j} = \frac{\lambda}{m_c} \tag{3.11}$$

must satisfy the condition $\rho < 1$. Thus, substituting ρ into (3.10) leads to the following expression for p_0:

$$p_0^{-1} = \sum_{i=0}^{c-1}\left[\frac{\lambda^i}{\prod_{j=1}^{i} m_j}\right] + \frac{(m_c)^c}{\left(\prod_{j=1}^{c} m_j\right)}\sum_{i=c}^{\infty}(\rho)^i. \tag{3.12}$$

As $|\rho| < 1$ and to keep the system stable, one has to satisfy

$$\sum_{i=c}^{\infty} (\rho)^i = (1-\rho)^{-1}(\rho^c). \tag{3.13}$$

Substituting (3.13) into (3.12), we obtain the following expression for p_0

$$p_0^{-1} = \sum_{i=0}^{c-1} \left[\frac{\lambda^i}{\prod_{j=1}^{i} m_j} \right] + \frac{(m_c)^c}{\left(\prod_{j=1}^{c} m_j \right)} \frac{1}{(1-(\lambda/m_c))} \frac{\lambda^c}{(m_c)^c}, \tag{3.14}$$

and finally

$$p_0^{-1} = \left(\sum_{i=0}^{c-1} \frac{\lambda^i}{\prod_{j=1}^{i} m_j} \right) + \frac{\lambda^c}{(1-\rho)\prod_{j=1}^{c} m_j}. \tag{3.15}$$

3.3. Computation of the Performance Measures

It is possible to obtain a formulation of the performance measures of the system from the probabilities p_i and p_0. The goal is to derive equations to measure (i) the average number in queue, L_q, and (ii) the average waiting time in queue, W_q. However, other performance measures could be chosen as well. In order to find the average number in queue, it is necessary to find its expectation, which is given by

$$L_q = \sum_{i=c}^{\infty} (i-c) p_i. \tag{3.16}$$

Substituting (3.8) into (3.16) leads to

$$\begin{aligned} L_q &= \sum_{i=c}^{\infty} (i-c) p_i \\ &= \sum_{i=c}^{\infty} (i-c) p_0 \frac{\lambda^i}{\left(\prod_{j=1}^{c} m_j \right) (m_c)^{i-c}} \\ &= p_0 \frac{(m_c)^c \rho^c}{\prod_{j=1}^{c} m_j} \sum_{i=c}^{\infty} (i-c)(\rho)^{i-c}. \end{aligned} \tag{3.17}$$

Using the change of indices $i = k + c$, one obtains

$$\begin{aligned} L_q &= p_0 \frac{(m_c)^c \rho^c}{\prod_{j=1}^{c} m_j} \sum_{k=0}^{\infty} (k)(\rho)^{\mathrm{k}} \\ &= p_0 \frac{(m_c)^c \rho^{c+1}}{\prod_{j=1}^{c} m_j} \sum_{k=0}^{\infty} (k)(\rho)^{k-1} \\ &= p_0 \frac{(m_c)^c \rho^{c+1}}{\prod_{j=1}^{c} m_j} \frac{d}{d\rho} \left[\sum_{k=0}^{\infty} (\rho)^k \right] \\ &= p_0 \frac{(m_c)^c \rho^{c+1}}{\left(\prod_{j=1}^{c} m_j \right)} (\rho - 1)^{-2}, \end{aligned} \tag{3.18}$$

therefore,

$$L_q = p_0 \frac{(m_c)^c \rho^{c+1}}{\left(\prod_{j=1}^{c} m_j \right) (\rho - 1)^2}. \tag{3.19}$$

The following equation for the average waiting time in queue, W_q, can be rearranged as

$$W_q = p_0 \frac{(m_c)^c \rho^{c+1}}{\left(\prod_{j=1}^{c} m_j \right) (\rho - 1)^2} \frac{1}{\lambda}, \tag{3.20}$$

according to Little's law, namely, $W_q = L_q/\lambda$. We remark that the approximations for the average number in queue, L_q, given by (3.19), and the average time in queue W_q, given by (3.20), are newly developed ones, for heterogeneous $M/M_i/c$ queueing systems.

4. Simulation Model

Although it is recognized that in some cases discrete event simulation techniques are less suitable because of the high computational capacities required, such techniques play a key role in queueing systems analysis [10]. In our work, they allow us to validate the developed upper bounds. Simulations are also used here to estimate the resulting errors when the selected performance measures are approximated by our proposed method or when they are approximated by a classical homogeneous $M/M/c$ queuing system. Our discrete event simulation model, which is available from the authors upon request, is coded for the well-known general purpose simulation system (GPSS) [30].

It is important to note that many simulation model limitations are encountered in our work. We experienced some difficulties in obtaining accurate results without too many computational efforts and during the generalization of the results. In addition, the required high accuracy led to some technical difficulties. The required level of accuracy dramatically increases the simulation running times as well as the number of replications

(large ρ coefficients require a high accuracy). In some cases, it is necessary to replicate the simulation many times in order to obtain appropriate mean standard errors (MSE) as low as 1% of the estimated average. Such low MSE are necessary in order to be able to compare the simulation results for different allocation strategies (i.e., FSF, RCS, and SSF). The desired accuracy is generally achieved after less than 200 replications, a simulation time of 700,000 timeunits, and a short burn-in (warm-up) period (further details on the selection of the warm-up period are given by Robinson [31]).

Finally, it is important to note that we have encountered some problems in the generalization of the results. Each model simulated gave results that were only valid for the specific system and combination of parameters to which they were related. Therefore, it was necessary to simulate many different configurations with slightly different parameters in order to get a complete understanding with the required accuracy of the general behavior of the $M/M_i/c$ systems. As a result, this additional number of configurations led to a considerable increase in the number of simulated cases.

5. Experimental Results

In this section, we present our experimental results in order to validate the quality of the proposed approximation and demonstrate that the proposed upper bounds can be used to get an estimate of the performance measures of $M/M_i/c$ queues. We will focus on the average queue waiting time, namely, W_q. However, we believe that the results will be insightful, because they will be immediately transposable to other performance measures, directly related to W_q by Little's law, such as the average number in queue L_q and the average number in the system L, as well as the average waiting time in the system, W. Our goal is to demonstrate that the upper bound given in (3.20) can be used to approximate W_q for heterogeneous Markovian multi-server queues, namely, $M/M_i/c$. In addition, we want to show that such an approximation is much more accurate than the traditional approximation given by homogeneous Markovian queues, namely, $M/M/c$.

To do so, it is necessary to thoroughly understand the proposed model. Thus, several types of queueing systems were created by changing for each one of them at least one of the following parameters:

(1) the number of servers c,

(2) the arrival rate of jobs, λ,

(3) the level of heterogeneity of the servers, given by the corresponding Gini Index.

The level of heterogeneity of the servers indicates how the overall processing capacity of the systems has been distributed among the servers. In this paper, we have chosen the Gini index of inequality to measure the differences between the processing capacities of the servers for a given system (see, for instance, Shalit [32] for details). This index ranges from 0 (completely homogeneous case) to 1 (completely heterogeneous case). For each system, W_q is calculated using both the proposed upper bound in (3.20) and the traditional homogeneous $M/M/c$ formula. In addition, we performed simulations using three different allocation strategies for each configuration (FSF, RCS, and SSF) to estimate three different simulated values for W_q. Different heterogeneities of the servers were simulated by varying the distributions of the overall processing capacity. Considering two servers for instance, they were initially treated as homogeneous, namely, by having a 50%–50% distribution of the total

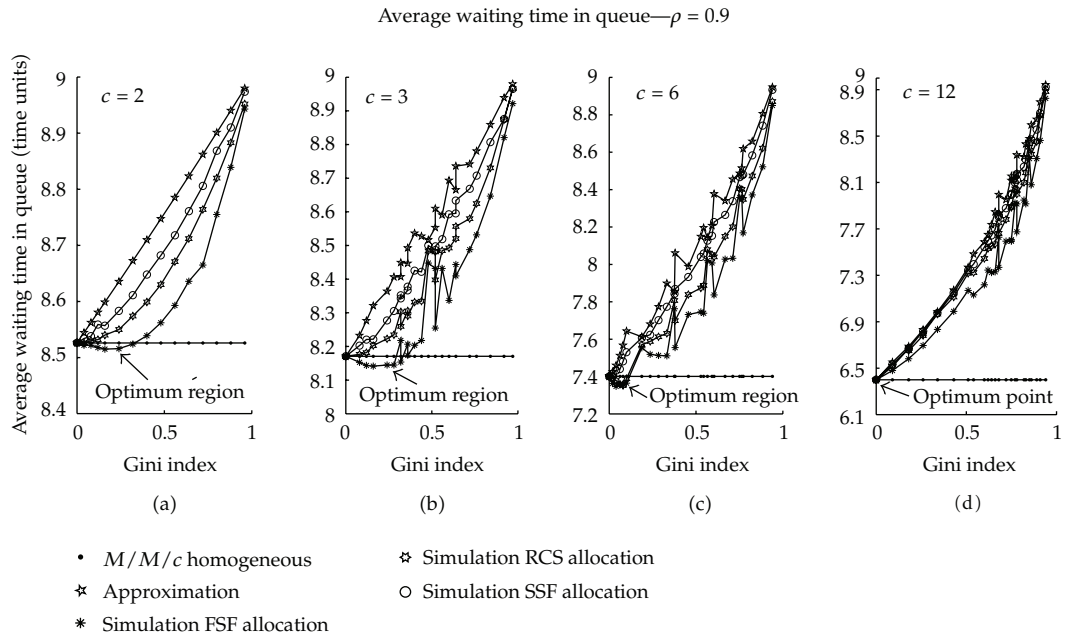

Figure 3: Average waiting times in queue for a heavily loaded system, $\rho = 0.9$.

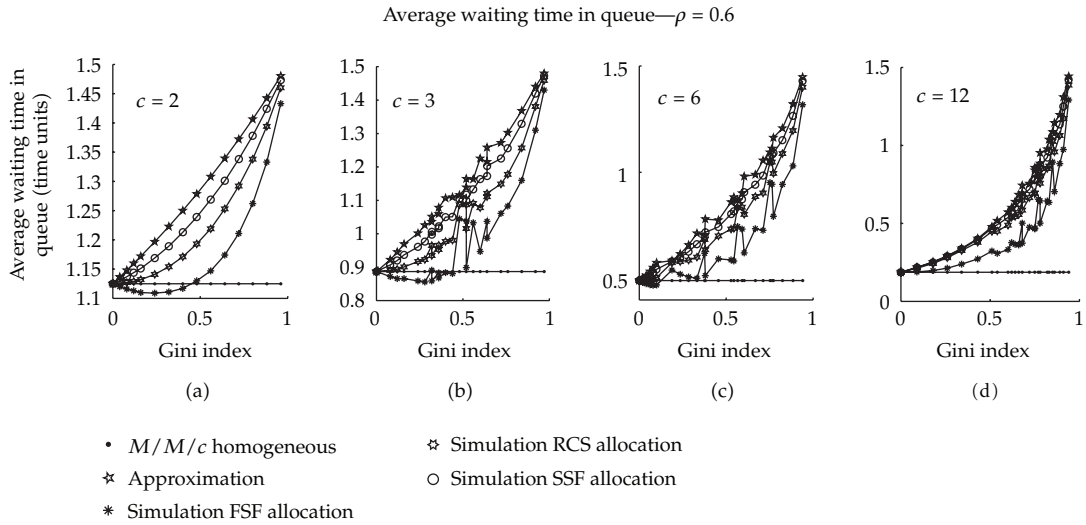

Figure 4: Average waiting times in queue for a moderately loaded system, $\rho = 0.6$.

processing rate μ. Afterwards, this ratio was changed to 52%–48%, 54%–46% and so on, until it reached 98%–2% (which is the most heterogeneous case considered in this work).

The average waiting times in queue W_q are plotted in Figures 3 and 4 for each configuration previously mentioned. In these figures, we show the values of W_q corresponding to the homogeneous $M/M/c$ and to our approximation method. They are compared with numerical results for three different allocation strategies, namely FSF, RCS, and SSF, and for

a number of servers equal to $c = 2, 3, 6$, and 12. Figure 3 corresponds to a heavily loaded system with $\rho = 0.9$, whereas Figure 4 corresponds to a moderately loaded system with $\rho = 0.6$.

Using the FSF allocation strategy, the systems can show a region for which the average waiting time in queue is optimum, namely, W_q is lower than the average waiting time in queue for a homogeneous system as indicated in Figures 3(a)–3(c). Thus, we have shown using numerical simulations that it is possible to have heterogeneous systems with better performance measures than homogeneous systems using the FSF allocation strategy, depending on the heterogeneity distribution among the servers. However, the simulations also revealed that the optimum region not only depends on the heterogeneity of the system itself but also on the number of servers. Each time, the number of servers was increased, the optimum region decreased until one single optimum point was found, which corresponded to the case where all the servers were homogeneous. Considering, for example, the case of $c = 12$ servers shown in Figure 3(d), we see that there exists no optimal region, even using the FSF allocation strategy. In other words, there is no condition for which the average waiting time in queue of a heterogeneous systems is better than that of a homogeneous one.

In order to better compare our approximation with the traditional homogeneous $M/M/c$ queue approximation, we used a normalized error given by

$$\text{error} = \frac{\|X_{\text{simulated}} - X_{\text{calculated}}\|}{X_{\text{simulated}}}. \tag{5.1}$$

Figures 5 and 6 represent the normalized errors for different allocation strategies and different numbers of servers for a heavily loaded system ($\rho = 0.9$) and for a moderately loaded system ($\rho = 0.6$), respectively. We observe that the normalized errors seem to be inversely proportional to ρ. In addition, considering the systems with only $c = 2$ servers initially for $\rho = 0.9$ (Figure 5(a)) and for $\rho = 0.6$ (Figure 6(a)), the maximum errors for the homogeneous $M/M/c$ queue (4.99% and 23.63% respectively; see Table 1) occurred using the SSF allocation strategy. This result is very different from that of our approximation, which predicts the maximum errors (2.27% and 14.08% resp.; see Table 1) using the FSF allocation strategy. The same differences are observed by analyzing the cases with $c = 3, 6$, and 12 heterogeneous servers (see Figures 5(b)–5(d) and Figures 6(b)–6(d)). These results are quite unexpected, and they might be useful to select the correct approximation to be used according to the available allocation strategy.

In Figures 5 and 6, it is possible to see the influence of the heterogeneities on the errors of the predictions from both methods (namely, the traditional homogeneous $M/M/c$ and the proposed upper bound). This constitutes another important characteristic of the approximation proposed. This new approximation shows a slower increase in the resulting errors when the number of servers is increased compared to the traditional homogeneous $M/M/c$ queue. In Figures 5 and 6, we observe that (i) the approximation proposed seems to give better predictions than the traditional homogeneous $M/M/c$ queue when the heterogeneity is high and that (ii) the Gini index for which the proposed approximation is better than the traditional homogeneous $M/M/c$ queue decreases when the number of servers is increased.

In other words, this seems to indicate that the proposed bounds give more rapidly better predictions than the traditional homogeneous $M/M/c$ queue as the number of servers increases. In fact, it is possible to see that for $c = 2$ and using the RCS allocation strategy

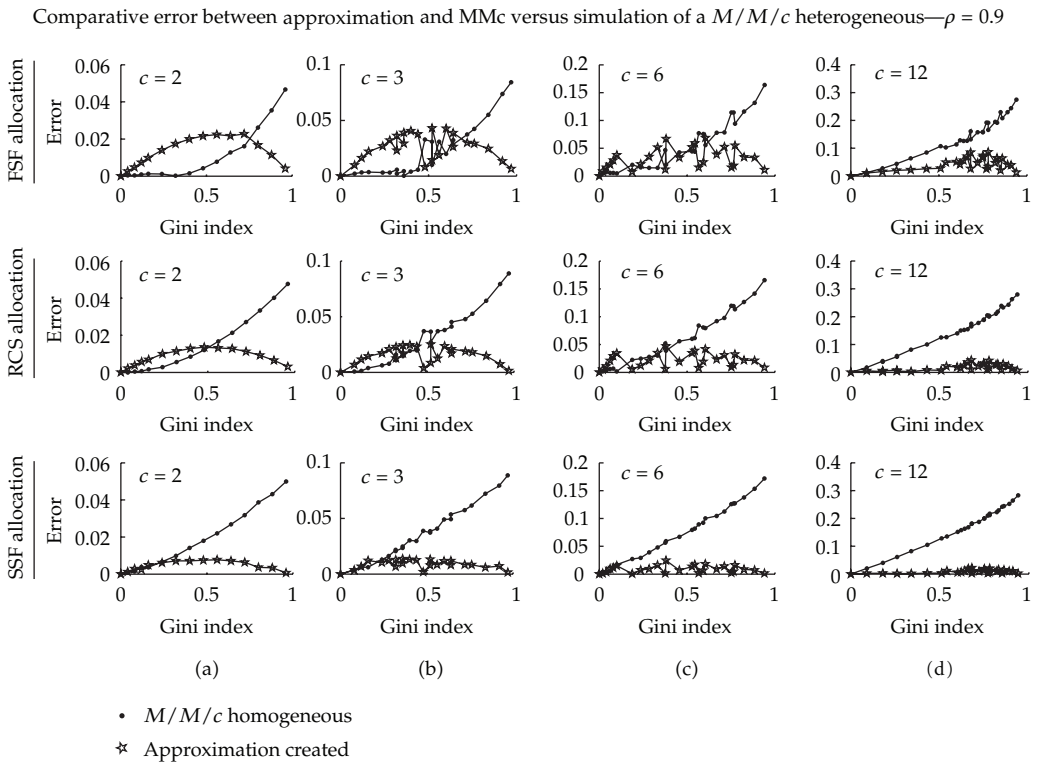

Figure 5: Errors in the estimation of the average waiting times in queue for the traditional homogeneous $M/M/c$ and for the proposed approximation for a heavily loaded system, $\rho = 0.9$.

in Figure 6(a), the proposed bound gives better results than the homogeneous $M/M/c$ queue for Gini indices larger than 0.54. On the other hand, for $c = 6$ and using the same RCS allocation strategy, we observe in Figure 6(c) that the proposed bound is more effective than the homogeneous $M/M/c$ queue for Gini indices only greater than 0.15. Therefore, we conclude that the proposed bounds give a better approximation of systems with heterogeneous servers when the number of servers is high and when there is a high degree of heterogeneity. However, we did not determine in this work for which values of the Gini index and for which number of servers the proposed bounds always give better predictions than the traditional homogeneous $M/M/c$ queue.

In Table 1, we list the maximum errors obtained for each model, when $\rho = 0.9$, 0.75, and 0.6. It is possible to observe the influence of the coefficient ρ on the approximations. As seen from Figures 5 and 6, the maximum errors are mostly for the Gini index around 0.5. This influence can be explained by the fact that only if one of the slow servers in the system is busy, it will be relevant for the upper bound of (3.20). However, the probability to find a busy slow server decreases when the utilization of the system decreases. That is, the number of jobs in the system decreases, and the variation of the states decreases (see Figure 2), as the traffic in the system decreases (low ρ). Therefore, when ρ becomes low, (3.20) is not a good approximation anymore. This observation is also true for the homogeneous $M/M/c$ queue. As the variation of the states increases in Figure 2, the approximation that all servers are equal becomes more unrealistic.

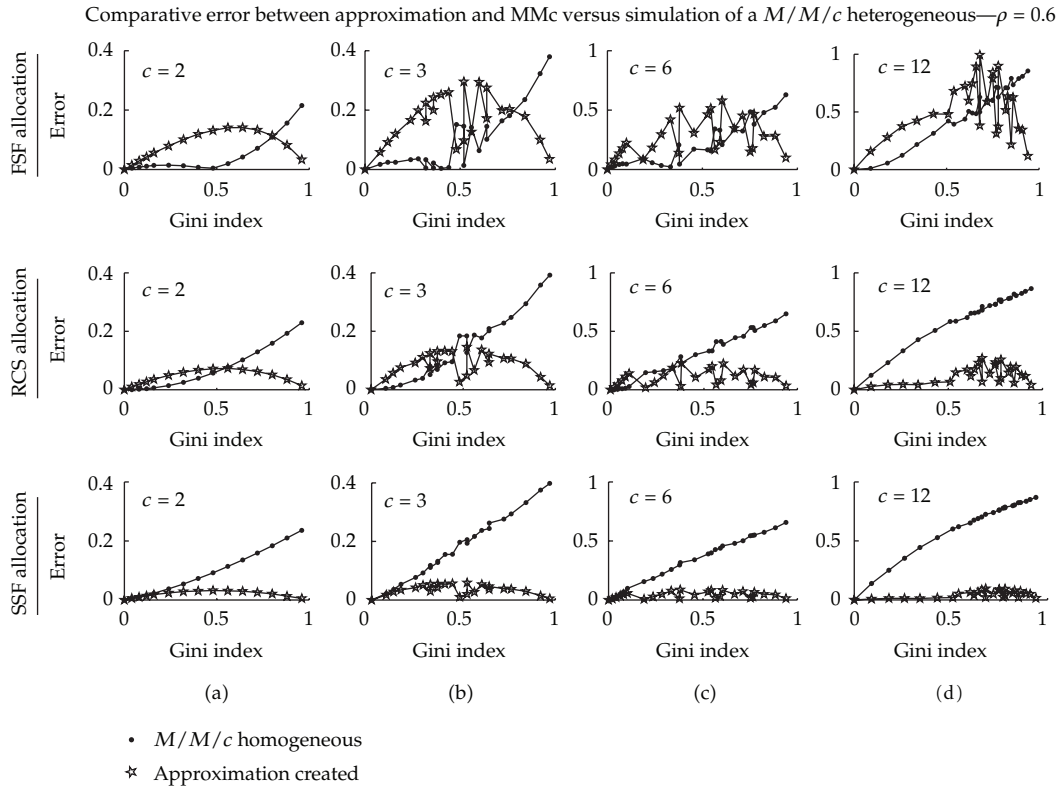

Figure 6: Errors in the estimation of the average waiting times in queue for the traditional homogeneous $M/M/c$ and for the proposed approximation for a moderately loaded system, $\rho = 0.6$.

In general, the errors increase when the number of servers increases. In Table 1, we observe that the maximum error always increases when the number of servers increases. That is a consequence of the fact that the systems with a greater number of servers have a higher number of possibilities for the job in service to be found than systems with a small number of servers. As a result, the higher the number of possibilities is, the higher the generated error in the approximation is as well as in the homogeneous $M/M/c$ queue. Another important point observed in Table 1 is that the maximum errors found for the proposed approximation is always lower than the maximum errors found for the homogeneous $M/M/c$ queue. This is an important result, as the proposed approximation is chosen to model a system with heterogeneous servers and the maximum possible error will be lower.

Finally, Table 2 summarizes the average error of each curve. The sum of the error obtained for each system is indicated, divided by the number of analyzed configurations. Three different values of ρ are shown, namely, 0.9, 0.75, and 0.6. The average errors can be related, for instance, to the curves shown in Figures 5 and 6. They were obtained by summing up the values inside a curve and by dividing this sum by the number of points on each curve. From this table, we can see the influence of the allocation policies on the results. For the allocation policies SSF and RCS, the average errors for the proposed upper bound are always lower than those of the homogeneous $M/M/c$ queue. However, for the allocation policy FSF, the point from which the proposed bound gives better predictions than the homogeneous $M/M/c$ queue depends on ρ and on the number of servers c.

Table 1: Maximum errors (in %) in the predictions of the average waiting times in queue for the proposed method and for the traditional homogeneous $M/M/c$ queue.

Number of servers			Allocation strategy		
c	ρ	Approximation method	FSF	RCS	SSF
2	0.90	Proposed bounds	**2.27**	1.36	0.77
		Homogeneous $M/M/c$	4.66	4.76	**4.99**
	0.75	Proposed bounds	6.91	3.93	1.93
		Homogeneous $M/M/c$	12.52	13.11	13.49
	0.60	Proposed bounds	**14.08**	7.31	3.18
		Homogeneous $M/M/c$	21.49	22.95	**23.63**
3	0.90	Proposed bounds	4.30	2.52	1.37
		Homogeneous $M/M/c$	8.42	8.88	8.87
	0.75	Proposed bounds	13.70	7.40	3.51
		Homogeneous $M/M/c$	22.62	23.33	23.59
	0.60	Proposed bounds	29.56	14.64	5.88
		Homogeneous $M/M/c$	37.97	39.21	39.79
6	0.90	Proposed bounds	6.89	4.70	2.45
		Homogeneous $M/M/c$	16.37	16.57	17.16
	0.75	Proposed bounds	23.58	11.82	5.43
		Homogeneous $M/M/c$	40.41	41.81	42.36
	0.60	Proposed bounds	62.75	22.64	8.64
		Homogeneous $M/M/c$	57.97	64.93	65.66
12	0.90	Proposed bounds	8.63	4.35	2.22
		Homogeneous $M/M/c$	27.48	27.96	28.28
	0.75	Proposed bounds	31.77	13.69	5.85
		Homogeneous $M/M/c$	62.09	63.17	63.53
	0.60	Proposed bounds	99.33	27.06	9.46
		Homogeneous $M/M/c$	85.53	86.54	86.87

6. Conclusions and Final Remarks

In many practical situations, queueing theory has been successfully applied [3, 8, 33]. Thus, the development and refinement of new analytical models is necessary to obtain better applications. The bounds developed in this paper for $M/M_i/c$ queues are a generalization of homogeneous $M/M/c$ queue formulas. We could develop worst case approximations for the invariant distribution of the number of jobs in a system, p_i, $i = 0, 1, \ldots$, for heterogeneous multi-server Markovian queues $M/M_i/c$, from which we derived tight upper bounds for useful performance measures, namely, the average number in queue L_q, and the average waiting time in queue W_q. From a comprehensive set of extensive computational experiments, we could validate the quality of the proposed bounds. The results presented here are certainly a step forward towards a better understanding of real-life heterogeneous multi-server queueing systems.

Future possible research in this field involves the development of tight lower bounds for the performance measures and extensions to general arrivals, batch arrivals, general service times, and finite queues. Also, it is important to consider that the lifetime of each server is finite in real life. The investigation of the effect of finite lifetimes in the performance of the server allocation strategies is another interesting topic for future research in the area.

Table 2: Average errors (in %) in the predictions of the average waiting times in queue for the proposed method and for the traditional homogeneous $M/M/c$ queue.

Number of servers			Allocation strategy		
c	ρ	Approximation method	FSF	RCS	SSF
2	0.90	Proposed bounds	1.31	0.82	0.46
		Homogeneous $M/M/c$	1.03	1.45	1.80
	0.75	Proposed bounds	4.02	2.33	1.16
		Homogeneous $M/M/c$	2.75	3.93	5.01
	0.60	Proposed bounds	8.10	4.28	1.91
		Homogeneous $M/M/c$	4.58	6.91	8.99
3	0.90	Proposed bounds	2.54	1.53	0.87
		Homogeneous $M/M/c$	2.15	2.96	3.59
	0.75	Proposed bounds	7.89	4.43	2.11
		Homogeneous $M/M/c$	5.62	8.07	10.09
	0.60	Proposed bounds	16.75	8.60	3.49
		Homogeneous $M/M/c$	9.38	14.00	17.98
6	0.90	Proposed bounds	3.15	2.12	0.88
		Homogeneous $M/M/c$	5.35	6.20	7.18
	0.75	Proposed bounds	10.74	5.17	2.49
		Homogeneous $M/M/c$	13.68	17.21	19.24
	0.60	Proposed bounds	26.19	10.27	4.22
		Homogeneous $M/M/c$	21.72	29.39	33.01
12	0.90	Proposed bounds	4.01	1.75	0.84
		Homogeneous $M/M/c$	13.43	15.27	16.00
	0.75	Proposed bounds	15.51	5.75	2.40
		Homogeneous $M/M/c$	33.26	38.56	40.32
	0.60	Proposed bounds	49.91	11.83	4.33
		Homogeneous $M/M/c$	48.30	60.44	62.51

Acknowledgments

This research has been partially funded by CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico; Grants nos. 201046/1994-6, 301809/1996-8, 307702/2004-9, 472066/2004-8, 304944/2007-6, 561259/2008-9, 553019/2009-0, 550207/2010-4, 501532/2010-2, 303388/2010-2), by CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior; Grant no BEX-0522/07-4), and by FAPEMIG (Grants no CEX-289/98, CEX-855/98, TEC-875/07, CEX-PPM-00401/08, and CEX-PPM-00390-10).

References

[1] A. M. Youssef and H. A. ElMaraghy, "Performance analysis of manufacturing systems composed of modular machines using the universal generating function," *Journal of Manufacturing Systems*, vol. 27, no. 2, pp. 55–69, 2008.

[2] J. Li, E. Enginarlar, and S. M. Meerkov, "Conservation of filtering in manufacturing systems with unreliable machines and finished goods buffers," *Mathematical Problems in Engineering*, vol. 2006, Article ID 27328, 12 pages, 2006.

[3] A. B. Hu and S. M. Meerkov, "Lean buffering in serial production lines with Bernoulli machines," *Mathematical Problems in Engineering*, Article ID 17105, 24 pages, 2006.

[4] I. Dimitriou and C. Langaris, "A repairable queueing model with two-phase service, start-up times and retrial customers," *Computers & Operations Research*, vol. 37, no. 7, pp. 1181–1190, 2010.
[5] T. van Woensel, L. Kerbache, H. Peremans, and N. Vandaele, "Vehicle routing with dynamic travel times: a queueing approach," *European Journal of Operational Research*, vol. 186, no. 3, pp. 990–1007, 2008.
[6] N. U. Ahmed and X. H. Ouyang, "Suboptimal RED feedback control for buffered TCP flow dynamics in computer network," *Mathematical Problems in Engineering*, Article ID 54683, 17 pages, 2007.
[7] J. Chen, C. Hu, and Z. Ji, "An improved ARED algorithm for congestion control of network transmission," *Mathematical Problems in Engineering*, vol. 2010, Article ID 329035, 14 pages, 2010.
[8] L. Tang, H. S. Xi, J. Zhu, and B. Q. Yin, "Modeling and optimization of $M/G/1$-type queueing networks: an efficient sensitivity analysis approach," *Mathematical Problems in Engineering*, Article ID 130319, 20 pages, 2010.
[9] T. van Woensel and F. R. B. Cruz, "A stochastic approach to traffic congestion costs," *Computers and Operations Research*, vol. 36, no. 6, pp. 1731–1739, 2009.
[10] Q. Wang, S. Lassalle, A. R. Mileham, and G. W. Owen, "Analysis of a linear walking worker line using a combination of computer simulation and mathematical modeling approaches," *Journal of Manufacturing Systems*, vol. 28, no. 2-3, pp. 64–70, 2009.
[11] H. Gumbel, "Waiting lines with heterogeneous servers," *Operations Research*, vol. 8, pp. 504–511, 1960.
[12] V. P. Singh, "Markovian queues with three heterogeneous servers," *AIIE Transactions*, vol. 3, no. 1, pp. 45–48, 1971.
[13] V. P. Singh, "Two-server Markovian queues with balking: heterogeneous vs. homogeneous servers," *Operations Research*, vol. 18, pp. 145–159, 1970.
[14] P. Le Gall, "The stationary $G/G/s$ queue with non-identical servers," *Journal of Applied Mathematics and Stochastic Analysis*, vol. 11, no. 2, pp. 163–178, 1998.
[15] P. Le Gall, "The stationary $G/G/s$ queue," *Journal of Applied Mathematics and Stochastic Analysis*, vol. 11, no. 1, pp. 59–71, 1998.
[16] K. W. Grassmann and Q. Y. Zhao, "Heterogeneous multiserver queues with general input," Tech. Rep., University of Winnipeg, Manitoba, Canada, 2004.
[17] O. J. Boxma, Q. Deng, and A. P. Zwart, "Waiting-time asymptotics for the $M/G/2$ queue with heterogeneous servers," *Queueing Systems*, vol. 40, no. 1, pp. 5–31, 2002.
[18] X. Chao and H. P. Luh, "A stochastic directional convexity result and its application in comparison of queues," *Queueing Systems*, vol. 48, no. 3-4, pp. 399–419, 2004.
[19] S. Y. Chiang, A. Hu, and S. M. Meerkov, "Lean buffering in serial production lines with nonidentical exponential machines," *IEEE Transactions on Automation Science and Engineering*, vol. 5, no. 2, pp. 298–306, 2008.
[20] S. Biller, S. P. Marin, S. M. Meerkov, and L. Zhang, "Closed bernoulli production lines: analysis, continuous improvement, and leanness," *IEEE Transactions on Automation Science and Engineering*, vol. 6, no. 1, pp. 168–180, 2009.
[21] M. Marmony, "Dynamic routing in large-scale service systems with heterogeneous servers," *Queueing Systems*, vol. 51, no. 3-4, pp. 287–329, 2005.
[22] A. van Harten and A. Sleptchenko, "On Markovian multi-class, multi-server queueing," *Queueing Systems*, vol. 43, no. 4, pp. 307–328, 2003.
[23] W. Lin and P. R. Kumar, "Optimal control of a queueing system with two heterogeneous servers," *IEEE Transactions on Automatic Control*, vol. 29, no. 8, pp. 696–703, 1984.
[24] G. Koole, "A simple proof of the optimality of a threshold policy in a two-server queueing system," *Systems & Control Letters*, vol. 26, no. 5, pp. 301–303, 1995.
[25] J. Walrand, "A note on 'optimal control of a queuing system with two heterogeneous servers'," *Systems & Control Letters*, vol. 4, no. 3, pp. 131–134, 1984.
[26] V. Rykov and D. Efrosinin, "Optimal control of queueing systems with heterogeneous servers," *Queueing Systems*, vol. 46, no. 3-4, pp. 389–407, 2004.
[27] S. Shenker and A. Weinrib, "Asymptotic analysis of large heterogeneous queueing systems," in *Proceedings of the ACMSIGMETRICS Conference on Measurement and Modeling of Computer Systems (SIGMETRICS '88)*, pp. 56–62, ACM, New York, NY, USA, 1988.
[28] S. Shenker and A. Weinrib, "The optimal control of heterogeneous queueing systems: a paradigm for load-sharing and routing," *IEEE Transactions on Computers*, vol. 38, no. 12, pp. 1724–1735, 1989.
[29] F. R. B. Cruz, T. van Woensel, J. MacGregor Smith, and K. Lieckens, "On the system optimum of traffic assignment in $M/G/c/c$ state-dependent queueing networks," *European Journal of Operational Research*, vol. 201, no. 1, pp. 183–193, 2010.

[30] T. J. Schriber, *An Introduction to Simulation Using GPSS/H*, John Wiley and Sons, New York, NY, USA, 1991.
[31] S. Robinson, "A statistical process control approach to selecting a warm-up period for a discrete-event simulation," *European Journal of Operational Research*, vol. 176, no. 1, pp. 332–346, 2007.
[32] H. Shalit, "Calculating the Gini index of inequality for individual data," *Oxford Bulletin of Economics and Statistics*, vol. 47, pp. 185–189, 1985.
[33] N. Al-Matar and J. H. Dshalalow, "Maintenance in single-server queues: a game-theoretic approach," *Mathematical Problems in Engineering*, Article ID 857871, 23 pages, 2009.