

Bayesian control of the number of servers in a $GI/M/c$ queueing system.

María Concepción Ausín,*

Departamento de Matemáticas, Universidade da Coruña,
15071 A Coruña, Spain.

R. E. Lillo, M. P. Wiper

Departamento de Estadística, Universidad Carlos III de Madrid,
28903 Getafe, Madrid, Spain.

Abstract

In this paper we consider the problem of designing a $GI/M/c$ queueing system. Given arrival and service data, our objective is to choose the optimal number of servers so as to minimize an expected cost function which depends on quantities, such as the number of customers in the queue. A semiparametric approach based on Erlang mixture distributions is used to model the general interarrival time distribution. Given the sample data, Bayesian Markov chain Monte Carlo methods are used to estimate the system parameters and the predictive distributions of the usual performance measures. We can then use these estimates to minimize the steady-state expected total cost rate as a function of the control parameter c . We provide a numerical example based on real data obtained from a bank in Madrid.

Keywords: Queueing systems, Bayesian control, birthdeath MCMC, optimal service channels.

1 Introduction

Optimal design and control of queues have been extensively investigated from an operational research point of view, see e.g. Kitaev and Rykov (1995). However, in this framework, the system parameters are typically assumed known. In practice, the system manager is faced with the problem of estimating the system parameters before solving the optimization problem. Furthermore, a common approach consist in selecting

¹Corresponding author. María Concepción Ausín, Departamento de Matemáticas, Facultad de Informática, Campus de Elviña, Universidad de A Coruña, 15071 A Coruña, Spain. Tel.: (+34) 981 167 000 (ext. 1318); fax: (+ 34) 981 167 160. E-mail address: mausin@udc.es

a queueing model and estimate the parameters without considering the uncertainty induced from this estimation in the system design. The Bayesian methodology offers a natural way to introduce the uncertainty resulting from the parameter estimation and model selection into a cost function depending on estimated performance measures.

Bayesian analysis of queueing systems is a fairly recent research area. Some recent references are Armero and Conesa (2000), Ausín et al. (2003, 2004). In these works, Bayesian inference and prediction is undertaken for different queueing models ranging from the $M/M/c$ system to more general queues. However, although most Bayesian analyses have considered the estimation of quantities of interest such as queue size, few studies have been devoted to the design and control problem. In one of the first works in Bayesian estimation for queues, Bagchi and Cunningham (1972) develop an optimal design procedure to find the optimum service rate and system capacity in a single server, Markovian queue. Also, Armero and Bayarri (1996) discusses some criteria for deciding the number of servers in a $M/M/c$ queue and Wiper (1998) also for the $Er/M/1$ model, but no systematic procedure for decision making is proposed. These works motivates the formulation of a closed expression based on a cost structure to address the decision problem on the number of servers.

On the other hand, most Bayesian analyses have considered queueing systems where the customers arrive according to a Poisson process. To the best of our knowledge, the only exception is Wiper (1998) where inference for the $Er/M/c$ model is considered. However, although the Erlang distribution may be used to fit interarrival (or service) time data with coefficient of variation less than one, it is inappropriate if the data have large coefficient of variation or are multimodal. Our objective in this paper is thus to consider Bayesian control for the general, $GI/M/c$ queueing system.

In § 2, we describe the $GI/M/c$ queueing model where we consider a semiparametric approximation to the general interarrival time distribution based on a mixture of Erlang distributions. Note that this family includes the Erlang, hyperexponential and exponential distributions, which are commonly used in the queueing literature, as special cases. It is also dense over the set of distributions on the positive reals.

The use of mixture distributions to model data is very common and the Bayesian approach provides an important tool for semiparametric density estimation, see, for example, Diebolt and Robert (1994). Markov Chain Monte Carlo methods (MCMC), see Robert (1996), have been developed for Bayesian analyses of mixture models. Recently, MCMC methods for exploring mixture models of unknown dimension have been proposed. Richardson and Green (1997) introduced the reversible jump technique to analyze normal mixtures. This type of algorithm was used by Ríos et al. (1998) for exponential mixtures and Wiper et al. (2001) for mixtures of gamma distributions. More recently, an alternative approach to reversible jump based on a birth-death process has been proposed by Stephens (2000). In § 3, we make use of the latter methodology to make inference for the system parameters. We define prior distributions and propose a

birth-and death MCMC algorithm to obtain a sample from the joint posterior distribution of the system parameters and the predictive interarrival time distribution.

In § 4 and § 5, we describe the estimation of various quantities of interest in the system and address the problem of optimizing the number of servers. Firstly, we estimate the traffic intensity and the probability that the equilibrium condition holds. Then, assuming a stable system, we estimate the predictive distributions of the system size and the waiting time in the queue, among other characteristics. Finally, we propose a steady state, average cost function which depends on the number of servers and some performance measures. The predictive cost and the performance measures are all estimated using the data generated from the MCMC algorithm.

In § 6, we illustrate the methodology with real data obtained from a bank in Madrid. Conclusions and a discussion of possible extensions are included in § 7.

2 Queueing model

Throughout, we will consider a multichannel queueing system with c servers, FIFO discipline and independence between interarrival and service times. Furthermore, service times are independent and exponentially distributed with unknown mean $1/\mu$. In order to model the general interarrival time distribution, we use a semiparametric model based on a mixture of Erlang distributions. Thus, customers are assumed to arrive individually with independent interarrival times distributed as a mixture of Erlang distributions. If T is a typical interarrival time, we have,

$$f(t | k, \mathbf{w}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = \sum_{r=1}^k w_r Er(t | \nu_r, \lambda_r), \quad 0 < t < \infty, \quad (1)$$

where k is the number of mixture components, $\mathbf{w} = (w_1, \dots, w_k)$, are weights and $Er(t | \nu_r, \lambda_r)$ represents the Erlang density function, which has been parameterized to have mean λ_r , for $r = 1, \dots, k$, that is,

$$Er(t | \nu_r, \lambda_r) = \frac{(\nu_r/\lambda_r)^{\nu_r}}{\Gamma(\nu_r)} t^{\nu_r-1} \exp(-\frac{\nu_r}{\lambda_r}t). \quad (2)$$

For fixed k , this model includes the usual Erlang, hyperexponential and exponential distributions as special cases and letting $k \rightarrow \infty$, essentially any distribution on the positive real line can be modeled as a mixture of Erlang distributions.

We wish to estimate the performance measures and a cost function for the system in equilibrium. The equilibrium condition for a $GI/G/c$ queue is that the traffic intensity, ρ , is less than the number of servers,

c , see, for example, Gross and Harris (1985). In the $GI/M/c$ model as outlined above, the traffic intensity is given by,

$$\rho = \left(\mu \sum_{r=1}^k w_r \lambda_r \right)^{-1}. \quad (3)$$

3 Bayesian inference.

In this section, we develop Bayesian inference techniques for the unknown arrival parameters, k , $\mathbf{w} = (w_1, \dots, w_k)$, $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_k)$, $\boldsymbol{\nu} = (\nu_1, \dots, \nu_k)$ and for the service parameter μ .

We consider throughout the simple experiment of observing n_s service times, $\mathbf{s} = \{s_1, \dots, s_{n_s}\}$, and n_a interarrival times, $\mathbf{t} = \{t_1, \dots, t_{n_a}\}$, which has been considered in a number of earlier articles; see e.g. Armero and Bayarri (1996). Given this experiment, the likelihood function separates into two parts, one concerning the arrival parameters, $(k, \mathbf{w}, \boldsymbol{\lambda}, \boldsymbol{\nu})$ and another concerning the service parameter, μ . Hence, assuming independent prior distributions for the arrival and service parameters, the corresponding posterior distributions will also be independent a posteriori.

3.1 Prior specification and updating

Here, we assign prior distributions for the system parameters. For the service rate, μ , we can assume a gamma prior distribution, $\mu \sim G(a, b)$, that is

$$f(\mu | a, b) = \frac{b^a}{\Gamma(a)} \mu^{a-1} e^{-b\mu} \quad \text{for } \mu > 0.$$

It is straightforward to show that, conditional on the service data, the posterior distribution is also gamma so that,

$$\mu | \mathbf{s} \sim G \left(a + n_s, b + \sum_{i=1}^{n_s} s_i \right). \quad (4)$$

In order to make inference for the interarrival distribution parameters, following Diebolt and Robert (1994), it is convenient to introduce a missing data formulation in which we define a set of independent and identically distributed (i.i.d.) latent variables, Z_1, \dots, Z_{n_a} , associated with the interarrival time variables, T_1, \dots, T_{n_a} , so that,

$$T_i | Z_i = r \sim Er(\nu_r, \mu_r), \quad P(Z_i = r | k, \mathbf{w}) = w_r,$$

for $r = 1, \dots, k$. With this approach, every interarrival data set, $\mathbf{t} = \{t_1, \dots, t_{n_a}\}$, is associated to a missing data set, $\mathbf{z} = \{z_1, \dots, z_{n_a}\}$, indicating the specific components of the mixture from which the observed interarrival times are assumed to arise.

Now, we can define a joint prior distribution on the mixture parameters, $(k, \mathbf{w}, \boldsymbol{\lambda}, \boldsymbol{\nu})$. Firstly, we assume a truncated Poisson prior distribution for the mixture size, k , taking values from 1 to k_{\max} ,

$$P(k) \propto \frac{\gamma^k}{k!}, \quad (5)$$

In practice we take $\gamma = 2$ and $k_{\max} = 10$ in order to penalize against overfitting the data with mixtures with a large number of components. We also define prior distributions for the remaining parameters conditional on k ,

$$\mathbf{w} \mid k \sim D(\phi_1, \dots, \phi_k), \quad \nu_r \mid k \sim GE(\vartheta), \quad \lambda_r \mid k \sim IG(\alpha, \beta),$$

for $r = 1, \dots, k$, where $D(\phi_1, \dots, \phi_k)$ denotes a Dirichlet distribution,

$$f(\mathbf{w} \mid k) = \frac{\Gamma(\phi_1 + \dots + \phi_k)}{\Gamma(\phi_1) \cdots \Gamma(\phi_k)} \prod_{r=1}^k w_r^{\phi_r - 1},$$

$GE(\vartheta)$ is a geometric distribution with mean $1/\vartheta$,

$$P(\nu_r) = (1 - \vartheta)^{\nu_r - 1} \vartheta, \quad \text{for } \nu_r = 1, 2, \dots,$$

and $IG(\alpha, \beta)$ denotes an inverted gamma distribution,

$$f(\lambda_r) = \frac{\beta^\alpha}{\Gamma(\alpha)} \lambda_r^{-(\alpha+1)} \exp\left(-\frac{\beta}{\lambda_r}\right).$$

Typically, in practice we set, for all $r = 1, \dots, k$; $\phi_r = 1$, which implies a uniform prior for \mathbf{w} and $\alpha = 1.1$, $\beta = 1$ and $\vartheta = 0.01$ giving fairly diffuse priors for λ_r and ν_r with finite means.

Conditional on k , and given the interarrival time data, the required posterior conditional distributions for the MCMC algorithm can be shown to be,

$$P(Z_i = r \mid \mathbf{t}, k, \mathbf{w}, \boldsymbol{\lambda}, \boldsymbol{\nu}) \propto w_r \frac{(\nu_r / \lambda_r)^{\nu_r}}{\Gamma(\nu_r)} t_i^{\nu_r - 1} \exp\left(-\frac{\nu_r}{\lambda_r} t_i\right), \quad \text{for } r = 1, \dots, k,$$

$$\mathbf{w} \mid \mathbf{t}, \mathbf{z}, k \sim D(\phi_1 + n_1, \dots, \phi_k + n_k),$$

$$\lambda_r \mid \mathbf{t}, \mathbf{z}, k \sim IG(\alpha + n_r \nu_r, \beta + S_r \nu_r),$$

and,

$$f(\nu_r \mid \mathbf{t}, \mathbf{z}, k, \mathbf{w}, \boldsymbol{\lambda}) \propto \frac{\nu_r^{n_r \nu_r}}{\Gamma(\nu_r)^{n_r}} \exp\left\{-\nu_r \left(-\log(1 - \vartheta) + \frac{S_r}{\lambda_r} + n_r \log \lambda_r - \log P_r\right)\right\}, \quad (6)$$

where $n_r = \#\{Z_i = r\}$, $S_r = \sum_{i:Z_i=r} t_i$ and $P_r = \prod_{i:Z_i=r} t_i$, for $r = 1, \dots, k$.

3.2 BDMCMC algorithm

In this subsection, we propose a Birth-Death MCMC (BDMCMC) algorithm to obtain a sample from the joint posterior distribution of the interarrival parameters, $k, \mathbf{w}, \boldsymbol{\lambda}$ and $\boldsymbol{\nu}$. The BDMCMC approach was introduced by Stephens(2000) for normal mixtures and is based on a birth-death process (BD) where the mixture size, k , changes so that births and deaths of the mixture components occur in continuous time. The stationary distribution of the BD process is the joint posterior of the mixture parameters. In order to improve mixing, the BD process can be combined with a standard MCMC method where k is kept fixed, as will be shown further on.

In the BD process, births of the mixture components occur at a constant rate which we might set equal to the parameter, γ , from the prior distribution of k in (5). A birth increases the number of components by one. The weight of the new component are generated from a beta distribution with parameters $(1, k)$ and the remaining parameters are sampled from the prior distribution. The death rate of every mixture component is a likelihood ratio of the model with and without this component, given by,

$$\delta_{r_0} = \prod_{i=1}^{n_a} \left(\frac{\sum_{\substack{r=1 \\ r \neq r_0}}^k \frac{w_r}{1-w_{r_0}} Er(t_i | \nu_r, \lambda_r)}{\sum_{r=1}^k w_r Er(t_i | \nu_r, \lambda_r)} \right), \quad \text{for } r_0 = 1, \dots, k.$$

Thus, death rates are very low if the corresponding component explains a lot of data and high if it does not. The total death rate, δ , of the process at any time is the sum of the individual death rates. A death decreases the number of mixture components by one. The birth and death processes are independent Poisson processes, thus, the time to next birth/death event is exponentially distributed with mean $1/(\delta + \gamma)$ and a birth or death occur with probabilities proportional to γ and δ , respectively.

Then, we define an algorithm, based on Stephens(2000), as follows:

1. Set initial values $k^{(0)}, \mathbf{w}^{(0)}, \lambda^{(0)}, \nu^{(0)}$.

Birth Death process.

2. Run the birth-death process for a fixed time t_0 .
 - 2.1. Start from $k^{(j)}, \mathbf{w}^{(j)}, \lambda^{(j)}, \nu^{(j)}$.
 - 2.2. Compute the death rates.
 - 2.3. Simulate the exponential time to next jump.
 - 2.4. Simulate the type of jump (birth or death).
 - 2.5. Modify the mixture components and

2.6. if the run time is less than t_0 go to 2.2.

MCMC algorithm conditional on k .

3. Update the allocation by sampling from $\mathbf{z}^{(j+1)} \sim \mathbf{z} | \mathbf{t}, k^{(j+1)}, \mathbf{w}^{(j)}, \lambda^{(j)}, \nu^{(j)}$.

4. Update the weights by sampling from $\mathbf{w}^{(j+1)} \sim \mathbf{w} | \mathbf{t}, \mathbf{z}^{(j+1)}, k^{(j)}$.

5. For $r = 1, \dots, k^{(j+1)}$,

5.1. Update the means by sampling from $\mu_r^{(j+1)} \sim \mu_r | \mathbf{t}, \mathbf{z}^{(j+1)}, k^{(j+1)}$.

5.2. Update ν_r using a Metropolis step.

6. $j = j + 1$. Go to 2.

Step 2 of the algorithm is the BD process described above. The BD process is run for a fixed time, t_0 , in each iteration of the algorithm. Following Stephens (2000), we have fixed in our examples $t_0 = 1$ because doubling t_0 is equivalent to doubling γ . As should be expected, we have found in practice that larger values of the birth rate, γ , produce better mixing but require more time in the computation of the algorithm.

Steps 3 to 5 are standard Gibbs sampling, see, for example, Gelfand and Smith (1990) whereby the model parameters are updated conditional on the mixture size, k . The only slightly complicated step is 5.2. where we introduce a Metropolis Hasting method, see Hastings(1970), to sample from the posterior distribution of ν . To do this, we generate candidate values for ν from a negative binomial proposal distribution. We have chosen this proposal distribution because, for large values of ν , the conditional distribution in (6) has a similar form to a negative binomial distribution. This part of the algorithm where the mixture size, k , is kept fixed is very similar to that used in Ausín et al. (2004).

This algorithm can be shown to produce a sample from the joint posterior parameter distribution; see e.g. Stephens (2000). Thus, given the MCMC output of size J , we can estimate the predictive density of the interarrival time distribution using,

$$f(t | \mathbf{s}, \mathbf{t}) = \frac{1}{J} \sum_{j=1}^J \sum_{r=1}^{k^{(j)}} w_r^{(j)} Er(t | \nu_r^{(j)}, \lambda_r^{(j)}). \quad (7)$$

For further details of the BDMCMC methodology in the context of Bayesian inference for a normal mixture model, see Stephens (2000) or Hurn et al. (2003). As commented before, this type of algorithms are an alternative to the reversible jump techniques introduced by Richardson and Green (1997). A reversible jump method was proposed in a previous work, to make inference on the general service time distribution for a M/G/1 system, see Ausín et al. (2004). In practice, we have found that, in general, both schemes perform similarly. However, the BDMCMC algorithm is somewhat easier to implement. Some illustrations comparing both approaches are shown in Section 6.

4 Estimation of performance measures in the system

Suppose now that we have obtained a Monte Carlo sample of size J from the posterior distribution of the arrival parameters, via the BDMCMC algorithm, and the service parameter μ via direct sampling of the gamma density $f(\mu | \mathbf{s})$ as in (4). Then we can estimate the probability of having a stationary distribution with,

$$P(\rho < c | \mathbf{s}, \mathbf{t}) \approx \frac{1}{J} \# \{ \rho^{(j)} < c \}, \quad (8)$$

where,

$$\rho^{(j)} = \left(\mu^{(j)} \sum_{r=1}^{k^{(j)}} w_r^{(j)} \lambda_r^{(j)} \right)^{-1}, \quad (9)$$

and $\{(k^{(1)}, \mathbf{w}^{(1)}, \boldsymbol{\lambda}^{(1)}, \boldsymbol{\nu}^{(1)}), \dots, (k^{(J)}, \mathbf{w}^{(J)}, \boldsymbol{\lambda}^{(J)}, \boldsymbol{\nu}^{(J)})\}$ is the sample obtained from the BDMCMC algorithm and $\{\mu^{(1)}, \dots, \mu^{(J)}\}$ is the sample generated from the posterior distribution of μ given by (4). If this probability is large, it may be reasonable to assume that the system is stable. Assuming equilibrium, we can estimate the traffic intensity, given in (3), as follows,

$$E[\rho | \mathbf{t}, \mathbf{s}, \rho < c] \approx \frac{1}{J_1} \sum_{j: \rho^{(j)} < c} \rho^{(j)}, \quad (10)$$

where $\rho^{(j)}$ is given in (9) and,

$$J_1 = \# \{ \rho^{(j)} < c \}, \quad (11)$$

is the size of the MCMC subsample where the equilibrium condition holds.

It is well known, see e.g. Gross and Harris (1985), that in queuing systems with non-Markovian interarrival process, the stationary distribution of the number of customers, N^* , found in the system by an arriving customer differs from the stationary distribution of the number of customers, N , found in the system at an arbitrary time instant. For our $GI/M/c$ model, given the system parameters, $\boldsymbol{\theta} = \{k, \mathbf{w}, \boldsymbol{\lambda}, \boldsymbol{\nu}, \mu\}$, we have that (see e.g. Allen, 1990),

$$P(N^* = n | \boldsymbol{\theta}) = \begin{cases} \sum_{m=n}^{c-1} (-1)^{m-n} \binom{m}{n} U_m & \text{for } n = 0, 1, \dots, c-2, \\ D\sigma^{n-c} & n \geq c-1, \end{cases} \quad (12)$$

where σ is the unique root in the interval $(0, 1)$ of the equation,

$$\sigma = f_A^*(c\mu(1-\sigma)), \quad (13)$$

and f_A^* is the Laplace transform of the interarrival time distribution,

$$f_A^*(s) = \sum_{r=1}^k w_r \left(\frac{\nu_r/\lambda_r}{s + \nu_r/\lambda_r} \right)^{\nu_r},$$

and,

$$\begin{aligned} U_m &= DC_m \sum_{p=m+1}^c \frac{\binom{c}{p}}{C_p(1-g_p)} \frac{c(1-g_p)-p}{c(1-\sigma)-p}, \quad \text{for } m = 0, 1, \dots, c-1. \\ D &= \left[\frac{1}{1-\sigma} + \sum_{p=1}^c \frac{\binom{c}{p}}{C_p(1-g_p)} \frac{c(1-g_p)-p}{c(1-\sigma)-p} \right]^{-1}, \\ C_p &= \begin{cases} 1 & \text{if } p = 0, \\ \prod_{m=1}^p \left(\frac{g_m}{1-g_m} \right) & \text{if } p = 1, 2, \dots, c, \end{cases} \\ g_p &= f_A^*(p\mu), \quad \text{for } p = 1, \dots, c, \end{aligned} \quad (14)$$

The distribution of N depends on the distribution of N^* and is given by,

$$P(N = n | \boldsymbol{\theta}) = \begin{cases} 1 - \frac{\rho}{c} - \rho \sum_{p=1}^{c-1} P(N^* = p-1 | \boldsymbol{\theta}) \left(\frac{1}{p} - \frac{1}{c} \right) & \text{for } n = 0, \\ \frac{\rho}{n} P(N^* = n-1 | \boldsymbol{\theta}) & \text{for } n = 1, \dots, c-1, \\ \frac{\rho}{c} P(N^* = n-1 | \boldsymbol{\theta}) & \text{for } n \geq c. \end{cases} \quad (15)$$

Assuming equilibrium, Monte Carlo approximations of the predictive stationary distributions of N^* and N , can be obtained. For example, we can approximate the predictive distribution of N by,

$$P(N = n | \mathbf{s}, \mathbf{t}, \rho < c) \approx \frac{1}{J_1} \sum_{j: \rho^{(j)} < c} P(N = n | \boldsymbol{\theta}^{(j)}) \quad (16)$$

where $\boldsymbol{\theta}^{(j)} = (k^{(j)}, \mathbf{w}^{(j)}, \boldsymbol{\lambda}^{(j)}, \boldsymbol{\nu}^{(j)}, \mu^{(j)})$ and J_1 is given in (11). Note that equation (13) has to be solved for every $\boldsymbol{\theta}^{(j)}$, but it is easy to approximate $\sigma^{(j)}$ by using the Newton-Raphson method or a similar procedure. Other quantities such as the stationary distribution of the number of busy servers can also be estimated although again, we must distinguish between the number of busy servers at arrival and arbitrary time instants, N_b^* and N_b . Observe that the number of busy servers is equal to the number of customers in the system if there are less customers than servers and equals c in the contrary case. Thus,

$$P(N_b = n | \mathbf{s}, \mathbf{t}, \rho < c) = \begin{cases} P(N = n | \mathbf{s}, \mathbf{t}, \rho < c) & \text{if } n < c, \\ P(N \geq c | \mathbf{s}, \mathbf{t}, \rho < c) & \text{if } n = c. \end{cases} \quad (17)$$

Conditioning on the number of busy servers, we can also calculate the distribution of the number of customers served per unit of time, N_s . Observe that in a $GI/G/c$ system with m busy servers, the variable N_s follows a Poisson distribution of rate μm . Then, given the system parameters, $\boldsymbol{\theta}$, we have that,

$$P(N_s | \boldsymbol{\theta}) = \sum_{m=1}^c \frac{(\mu m)^n e^{-\mu m}}{n!} P(N_b = m | \boldsymbol{\theta}).$$

And given the data, the predictive distribution can be approximated as before with,

$$P(N_s = n | \mathbf{s}, \mathbf{t}, \rho < c) \approx \frac{1}{J_1} \sum_{j: \rho^{(j)} < c} P(N_s = n | \boldsymbol{\theta}^{(j)}) \quad (18)$$

Other important quantities are the predictive distributions of the number of customers in the queue at arrival and arbitrary time instants, N_q^* and N_q . In the first case, we have,

$$P(N_q = n | \mathbf{s}, \mathbf{t}) = \begin{cases} P(N \leq c | \mathbf{s}, \mathbf{t}) & \text{if } n = 0, \\ P(N = c + n | \mathbf{s}, \mathbf{t}) & \text{if } n \geq 1. \end{cases} \quad (19)$$

Another measure which is of interest to arriving customers, is the waiting time in the queue, W . Given the system parameters, $\boldsymbol{\theta}$, this is exponentially distributed with a jump of height $P(W = 0)$ at the origin. The distribution function is given by,

$$F_W(x | \boldsymbol{\theta}) = 1 - P(W > 0) \exp\{-c\mu(1 - \sigma)x\}, \quad x \geq 0, \quad (20)$$

see Allen (1990), where,

$$P(W > 0 | \boldsymbol{\theta}) = \frac{D}{1 - \sigma}, \quad (21)$$

and where σ and D are given in (13) and (14), respectively. As above, we can use the following Monte Carlo approximation,

$$F_W(x | \mathbf{s}, \mathbf{t}) \approx \frac{1}{J_1} \sum_{j: \rho^{(j)} < 1} F_W(x | \boldsymbol{\theta}^{(j)}) \quad (22)$$

Wiper (1998) shows that, for any given $GI/M/1$ system, where independent, continuous priors on the arrival and service rates with positive density in $\rho = 1$ are considered, the moments of the predictive distributions of waiting time and queue size do not exist. It is straightforward to see that the moments for N^* , N , N_q , N_q^* and W do not either exist for the multiserver system, $GI/M/c$, with the same prior conditions. Thus, the distributions given in (16), (19) and in (22) do not have finite moments. It is possible however to evaluate the expectations of these predictive distributions if we assume $\rho < c - \varepsilon$ instead of $\rho < c$,

see Lehoczky (1990), but we have found in practical examples that this procedure is very sensible to the election of ε .

Observe, on the other hand, that the predictive distributions of the number of busy servers, N_b , and the number of served customers, N_s , given in (17) and (18), respectively, do have finite mean. This is straightforward to see as for any $GI/G/c$ system in equilibrium, the expected number of busy servers is,

$$E[N_b | \boldsymbol{\theta}] = \rho, \quad (23)$$

see e.g. Gross and Harris (1985), which in our queuing model is given by (3), and on average, each busy server attends μ clients per unit of time so that the number of served clients per unit of time is,

$$E[N_s | \boldsymbol{\theta}] = \mu E[N_b | \boldsymbol{\theta}] = \left(\sum_{r=1}^k w_r \lambda_r \right)^{-1}. \quad (24)$$

5 Cost functions and optimal control for the model.

In this section, we formulate cost functions in order to address the design problem for the $GI/M/c$ queueing model and determine the optimal number of servers in the system. We consider a classical, linear, cost structure evaluated in the stationary state. Each cost function will depend linearly on the expected values of the performance measures considered in the previous section, or equivalently, on their mean values per unit of time (u.t.). Thus, we are dealing with an infinite horizon problem where the objective function is the expected cost per u.t. evaluated in the stationary state.

Also, our aim is to construct cost functions which balance the designer's and the customers' interests. For that reason, we consider two different classes of costs in the queue: on the one hand, costs incurred from servers activities and, on the other hand, costs incurred from the wait of clients. The first group of costs includes the expenses coming from the number of busy and empty servers and the benefits obtained from the number of served clients which are all of them associated to the designers' interests. The second group of costs represents the customers' interests and are related with the number of clients and the period of time they spend waiting in the queue. We introduce the following notation to define the cost structure:

r_b = cost per u.t. associated to the number of busy servers.

r_e = cost per u.t. associated to the number of empty servers.

r_s = cost per u.t. associated to the number of served customers.

r_q = cost per u.t. associated to the number of customer waiting in the queue.

r_W = cost per u.t. associated to the waiting time in the queue.

These costs can take positive or negatives values depending on whether they correspond to profits or losses. As the problem of designing a queue is not generally a work for clients but for people supervising the system, we consider performance measures at arbitrary time instants and not at arrival time instants, both described in the previous section. Under this construction, the total cost per u.t. will be,

$$Cost = r_b L(N_b) + r_e \{c - L(N_b)\} + r_s L(N_s) + r_q L(N_q) + r_W L(W), \quad (25)$$

where N_b , N_s , N_q and W are the number of busy servers, the number of served customers, the number of customers and the waiting time in the queue, respectively, defined in the previous section. L is a loss function which represents the loss due to each performance measure. For example, $L(N_b)$ represents the loss due to the number of busy servers and $L(N_q)$ is the loss due to the number of people waiting for service. Many different loss formulations can be defined. For example, consider the loss associated with N_q , we can assume the following loss structure,

$$L_1(N_q) = \begin{cases} 0 & \text{if } N_q \leq n_q, \\ 1 & \text{if } N_q > n_q, \end{cases} \quad (26)$$

where a cost, r_q , is incurred per u.t. if the queue length exceeds a previously specified threshold, n_q . A more realistic alternative would be to consider a linear cost proportional to the number of waiting customers,

$$L_2(N_q) = \begin{cases} N_q & \text{if } N_q \leq n_q, \\ n'_q & \text{if } N_q > n_q, \end{cases} \quad (27)$$

where a cost, r_q , is incurred per u.t. per customer in the queue if the queue length does not exceed a threshold, n_q , and a constant cost, $r_q n'_q$, is incurred per u.t. when the threshold is exceeded, where n'_q is another previously fixed finite constant.

Similar L_1 and L_2 loss functions can be formulated for the other performance measures, N_b , N_s and W , for which a set of thresholds, n_b , n_s and w , respectively, have to be fixed. Note that the values of n_q and w must be finite by assumption because, as pointed out, the predictive distributions of N_q and W have no finite moments, and thus, an infinite value for a threshold will lead to an infinite value of the expected cost. On the contrary, we can assume infinite values for n_b and n_s as the predictive mean of n_b and n_s are finite.

Given that the system parameters, θ , verify the equilibrium condition and considering L_1 loss functions

as the given in (26), the expected cost per u.t. is a function of the number of server given by,

$$g(c | \boldsymbol{\theta}) = r_e c + (r_b - r_e) P(N_b > n_b | \boldsymbol{\theta}) + r_s P(N_s > n_s | \boldsymbol{\theta}) + r_q P(N_q > n_q | \boldsymbol{\theta}) + r_W P(W > w | \boldsymbol{\theta}). \quad (28)$$

Then, given the interarrival and service data, the predictive mean cost can be estimated using the tail probabilities of the predictive distributions given in (17), (18), (19) and (22).

However, in practice, the most common assumption is to consider a linear cost function whose mean value is the main interest, see e.g. (*referecia clásica relevante*). Then, in order to develop a parallel analysis to the classical approach, we consider L_2 loss functions for all performance measures, as the given in (27), with infinite thresholds whenever it is possible. Thus, using (23) and (24), the expected cost per u.t. is now given by,

$$g(c | \boldsymbol{\theta}) = r_e c + (r_b - r_e + r_s \mu) \rho + r_q E[L_2(N_q) | \boldsymbol{\theta}] + r_W E[L_2(W) | \boldsymbol{\theta}], \quad (29)$$

where the expected loss for N_q is given by,

$$E[L_2(N_q) | \boldsymbol{\theta}] = \sum_{n=0}^{n_q} n P(N = n + c | \boldsymbol{\theta}) + n'_q [1 - P(N \leq n_q + c | \boldsymbol{\theta})],$$

where the distribution of N can be derived from (15), and the expected loss for W can be shown to be,

$$E[L_2(W) | \boldsymbol{\theta}] = \frac{D [1 - e^{-c\mu(1-\sigma)w}]}{c\mu(1-\sigma)^2} + \frac{D [w' e^{-c\mu(1-\sigma)w'} - w e^{-c\mu(1-\sigma)w}]}{1-\sigma},$$

which is simplified to the first summand for the particular case where w and w' are chosen to be equal.

If the system parameters are not known, but we have a sample of interarrival and service times, $\{\mathbf{t}, \mathbf{s}\}$, we can estimate the mean cost per u.t. given the MCMC output in the usual way,

$$g(c | \mathbf{t}, \mathbf{s}, \rho < c) \approx \frac{1}{J_1} \sum_{j: \rho^{(j)} < c} g(c | \boldsymbol{\theta}^{(j)}), \quad (30)$$

For discrete functions, it is possible to find out how many minima there are considering a monotone optimal procedure, see Lillo and Martín (2000). This consists in finding a point, c_0 , of the objective function, $g(c)$, where $g(c_0 + 1) - g(c_0) > 0$, and such that, $g(c + 1) - g(c) > 0$ for every $c > c_0$. It can be shown that the expected cost functions (28) and (29) allow a monotone optimal procedure if $r_e > 0$. Observe in (28) that when the number of servers, c , increases, the probabilities that N_b and N_s are larger than n_b and n_s approaches to a constant value and the probabilities that N_q and W are larger than n_q and w approaches to zero. Then, $g(c)$ will be approximately linearly increasing for large c . The same argument can be used

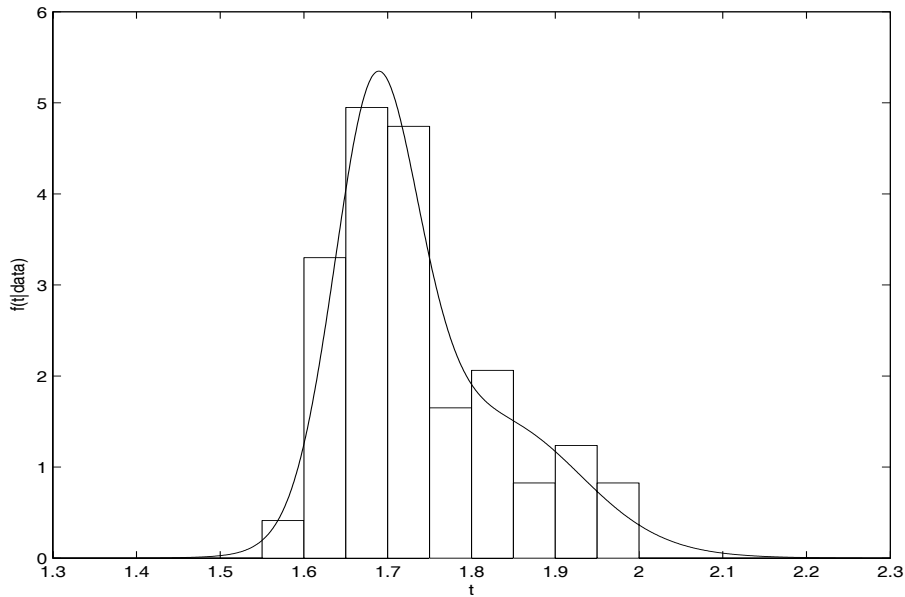


Figure 1: Histogram of interarrival time data and estimated interarrival time density.

for the expected cost function given in (29).

6 Bank data problem.

In this section, we consider the design of a multiserver real bank in Madrid. Interarrival and service times of 98 customers are recorded from 10:00 to 11:30 in the morning during three days. The mean service time is approximately 4.58 minutes. Using the proposed BDMCMC method for the service data, the posterior probability of having a single component is larger than 0.9. Thus, our Bayesian density estimation method predicts an exponential distribution for service time distribution. Then, we assume this model for the service time. We also use a non-informative prior in (4) by setting a and b equal to zero. Then, the posterior distribution of the service rate parameter, μ , is $G(98, 449.42)$.

Figure 1 shows the histogram of the 98 interarrival times. The estimated density function (7) using the Erlang mixture with the BDMCMC algorithm has been superimposed. None of times is larger than two minutes and the distribution seems to be bimodal. In fact, the posterior probability of having two Erlang mixture components is very high, $P(k = 2 | \mathbf{t}) \approx 0.9013$.

For comparison purposes, we also run the reversible jump algorithm (RJMCMC) proposed in Ausín et al. (2004) for these interarrival and service data. Note that this approach is a little harder to implement mainly because of the Jacobian calculation. As expected, we observe that the predictive densities obtained with both approaches are very close. However, the BDMCMC method seems to move faster through the

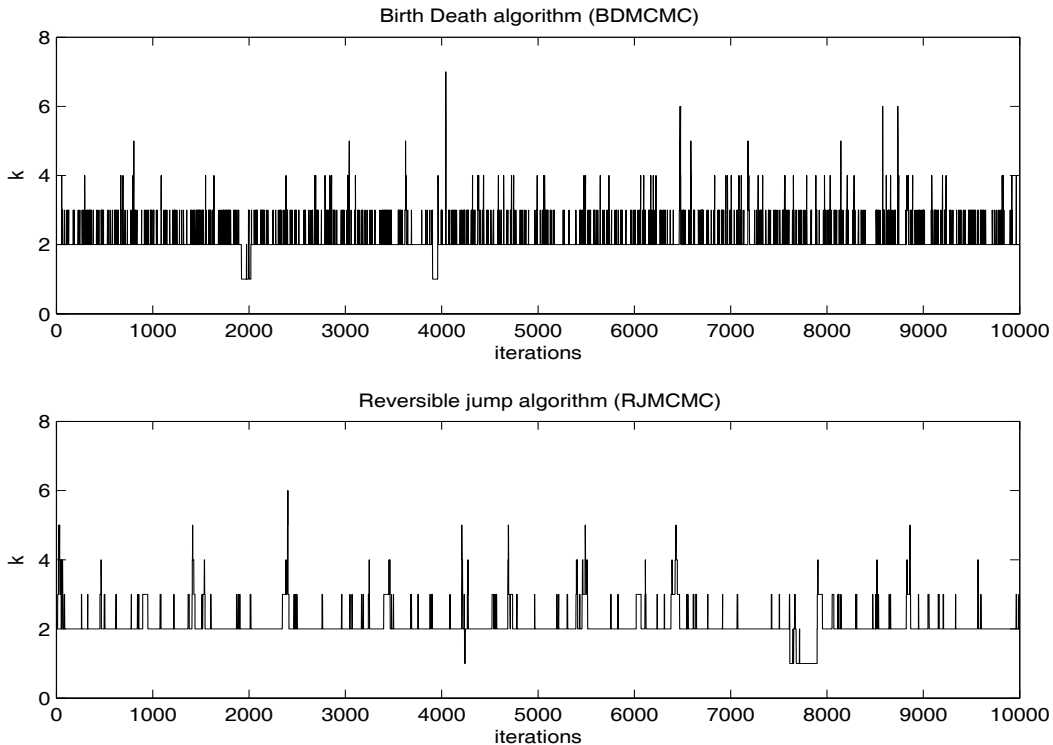


Figure 2: Changes in k against the first 10000 iterations for the interarrival time data with the BDMCMC method (top) and the RJMCMC method (bottom).

posterior distribution of k . This is illustrated in Figure 2, where the mixing of both chains for the interarrival time data is compared. This effect does not really imply a better convergence for the BDMCMC, but it is due to the continuous structure of the algorithm, see Cappé et al. (2003). In fact, we have observed a similar convergence performance for the posterior distribution of k with both approaches. As commented in Cappé (2003), we have also observed that the BDMCMC chain is more likely to visit the low probability regions as e.g. large values of k . Also, note that the BDMCMC is somewhat more costly in computational time because of the calculations of death rates for each mixture size. For further details in the choice of BDMCMC vs. RJMCMC see Ausín (2004).

Given the observed arrival and service data, we now estimate the posterior probability of having a stable system, see (8), for different values of c , which are shown in Table 1. Observe that at least, 3 servers are needed to assume that the ergodic condition, $\rho < c$, holds. However, 3 servers may not satisfy the optimal conditions resulting from the balance of costs in the system, as will be shown below.

Table 1 also shows the estimations for the traffic intensity for each c , see (10). Note that using (23), it can be shown that,

$$E[N_b | \mathbf{s}, \mathbf{t}, \rho < c] = E[\rho | \mathbf{s}, \mathbf{t}, \rho < c].$$

Then, when there are only 1 or 2 servers, all of them are almost always busy on average as the system is probably unstable. But, when there are 3 servers or more, the equilibrium condition holds with high probability and there are approximately 2.66 busy servers on average.

c	1	2	3	4	5	6
$P(\rho < c \mid \mathbf{s}, \mathbf{t})$.000228	.003328	.890256	.999916	1.00000	1.00000
$E[\rho \mid \mathbf{s}, \mathbf{t}, \rho < c]$	0.57401	1.75714	2.59582	2.65789	2.65801	2.65801

Table 1: Estimations of the posterior probabilities of having a stable system and the expected values for the traffic intensity for some values of c .

Figure 3 illustrates the estimated probabilities describing the number of customers in the system, N , see (16), at arbitrary time instants, for 3, 4 and 5 servers. Note that the probability of having 2 or 3 customers in the system are very similar for each number of servers. We have observed that this feature does not appear in the predictive distribution of N^* where we have identified the mode in 2 customers for any number of servers. This illustrates the fact that, although the mean number of busy servers at an arbitrary time instant is approximately 2.66, the mean number of busy servers found by an arrival customer is less than 2.66. It can also be seen that the distribution of N in a system with 3 servers has a long tail compared to other systems. Note that just by increasing the number of servers from 3 to 4 the probability of having an empty queue, $P(N_q = 0 \mid \mathbf{s}, \mathbf{t})$, grows from 0.44 to 0.87.

Figure 4 shows the distribution of the waiting time in the queue, W , see (22), for 3, 4 and 5 servers. Observe that, in a system with 3 servers, the probability of having to wait less than 10 minutes is fairly large, $P(W < 10 \mid \mathbf{s}, \mathbf{t}) \approx 0.8$. However, again, if the value of c is increased from 3 to 4 the probability of not having to wait, $P(W = 0 \mid \mathbf{s}, \mathbf{t})$, obtained from (22), grows from 0.37 to 0.82.

Now, we can address the optimization problem with the bank data. We wish to analyze the influence of the values of the different elements in the cost function. Thus, we formulate different cost functions defined from the minimum number of servers from which we have assumed equilibrium, that is, 3 servers. First, we examine the influence of the choice of costs. In practice, it is not easy to assign costs associated with the wait of customers, but, in general, the costs incurred from servers activities are known. Thus, we consider fixed costs for r_b, r_e and r_s , and different values for r_q and r_W . We assume a linear cost function defined with L_2 loss functions, see (27), and infinite thresholds for all variables with the exception of N_q and W whose thresholds must be finite and are assumed to be e.g. $n_q = n'_q = 5$ customers and $w = w' = 10$ minutes, respectively. The choice of thresholds and loss functions will be discussed later on.

Figure 5 shows the estimated cost functions obtained from (30) for different values of the costs per u.t. The first case (solid line) considers all the costs equal to the unity (except for $r_s = -1$ which is usually a benefit). The optimum value is 4 servers. Note that for $c \geq 5$, the waiting costs, r_q and r_W , have no

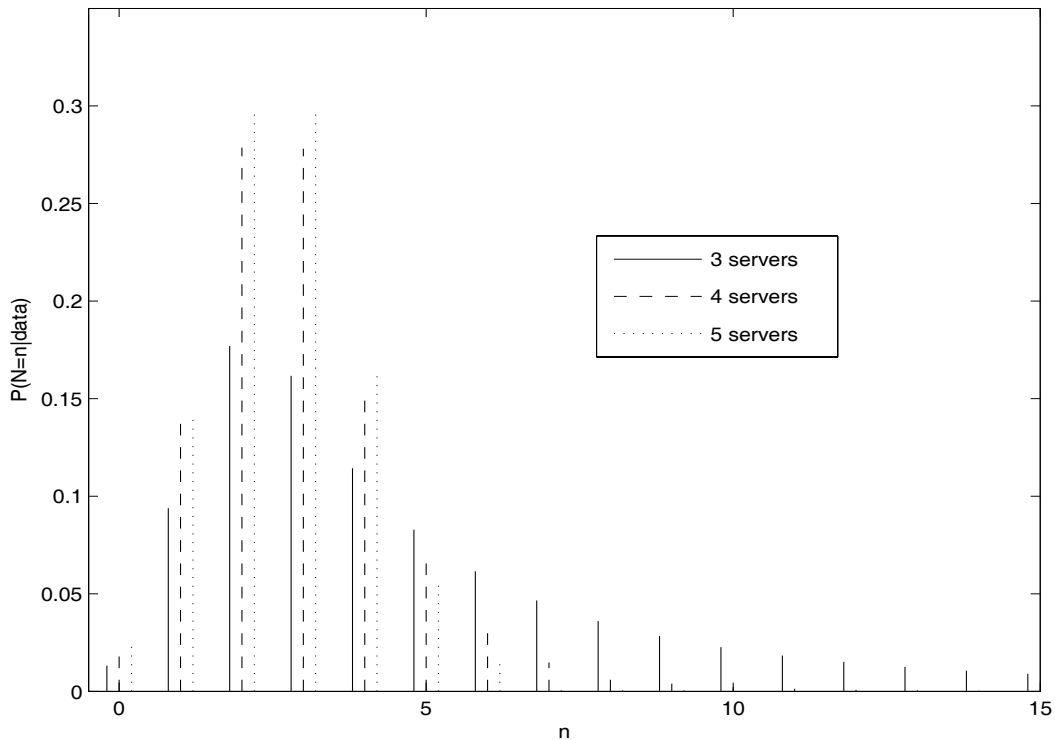


Figure 3: Predictive probabilities for the variable N , the number of customers in the system at arbitrary time instants for queues with 3, 4 and 5 servers.

effect and the cost function becomes linearly increasing as commented in Section 5. In the second case (dashed line), the costs of customer's interest, r_q and r_W , are increased in one unit. As expected, this affects increasing the expected cost for small c and the optimum value is now 5 servers. However, the expected cost is not affected for $c \geq 5$, which is approximately linear and similar to the first case. Conversely, in the third case (dotted line), when r_q and r_W are decreased, the cost function is decreased for small c and the optimum value is 3 servers. The same pattern with these three cases is observed for different values of the costs. For example, in case 4 (dash-dotted line), the cost function with two-unit costs is plotted. As in the first case, the optimum value is 3 servers although now the total cost function is much larger, as expected. If the waiting costs r_q and r_W were increased or decreased in this case, we would observe the same effects in this cost function as those observed in cases 2 and 3.

We now explore the influence of the selection of thresholds. In particular, we analyze the sensitivity to the choice of n_q and w because of their finiteness restriction. We firstly assume that to $n'_q = n_q$ and $w' = w$. Table 2 shows the estimated average cost obtained with unit costs and L_2 loss functions, as in the previous case 1, using different values for n_q and w . Note that the expected cost decreases as the values of the thresholds are decreased. The reason is that the predictive mean of $L_2(N_q)$ and $L_2(W)$ goes to infinity

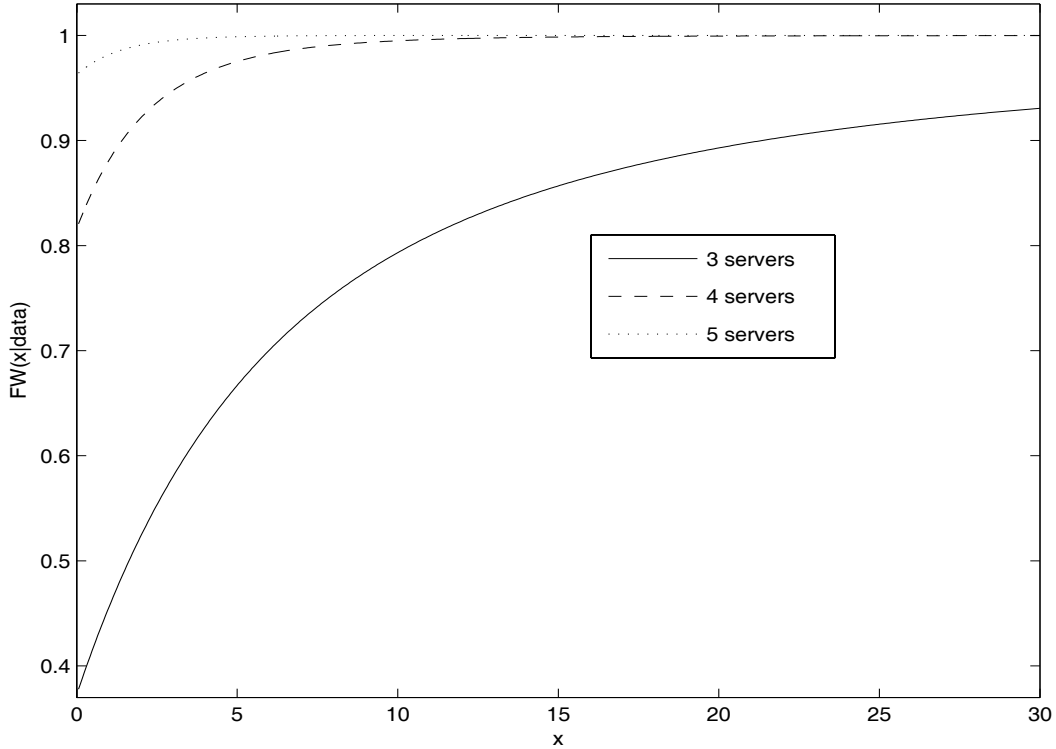


Figure 4: Predictive distribution functions of the variable W , the waiting time (in minutes) in the queue for systems with 3, 4 and 5 servers.

as these thresholds increase. However, although this effect is evident for $c = 3$, it is not readily noticeable for $c \geq 4$. In fact, the optimum value is 4 servers in all cases. The same optimum is obtained even assuming that n'_q and w' are much larger than n_q and w , respectively. As in this example, we have observed that, in general, the choice of the thresholds is quite robust to the resulting optimum number of servers.

c	$g(c \mid \mathbf{t}, \mathbf{s}, \rho < c)$					
	$n_q = 40$ $w = 100$	$n_q = 20$ $w = 30$	$n_q = 5$ $w = 10$	$n_q = 5$ $w = 5$	$n_q = 1$ $w = 1$	$n_q = 0$ $w = 0.25$
3	15.0486	11.9154	7.9219	6.6116	3.5655	2.5806
4	4.2067	4.1817	4.1260	4.0656	3.7052	3.4700
5	4.5355	4.5236	4.5143	4.5122	4.4751	4.4349
6	5.4228	5.4293	5.4343	5.4342	5.4325	5.4276
7	6.4330	6.4301	6.4280	6.4280	6.4272	6.4265
8	7.4263	7.4264	7.4264	7.4264	7.4264	7.4264

Table 2: Estimated average cost per u.t. for different values of n_q and w using L_2 loss functions for N_q and W . Optimal values are indicated in bold.

Finally, we design the same experiment but considering now loss functions of L_1 type, see (26), for N_q and W . Table 3 shows the estimated average cost for different values of n_q and w . Observe that, conversely to the previous case, the expected cost increases as the thresholds decrease. This should be expected as the

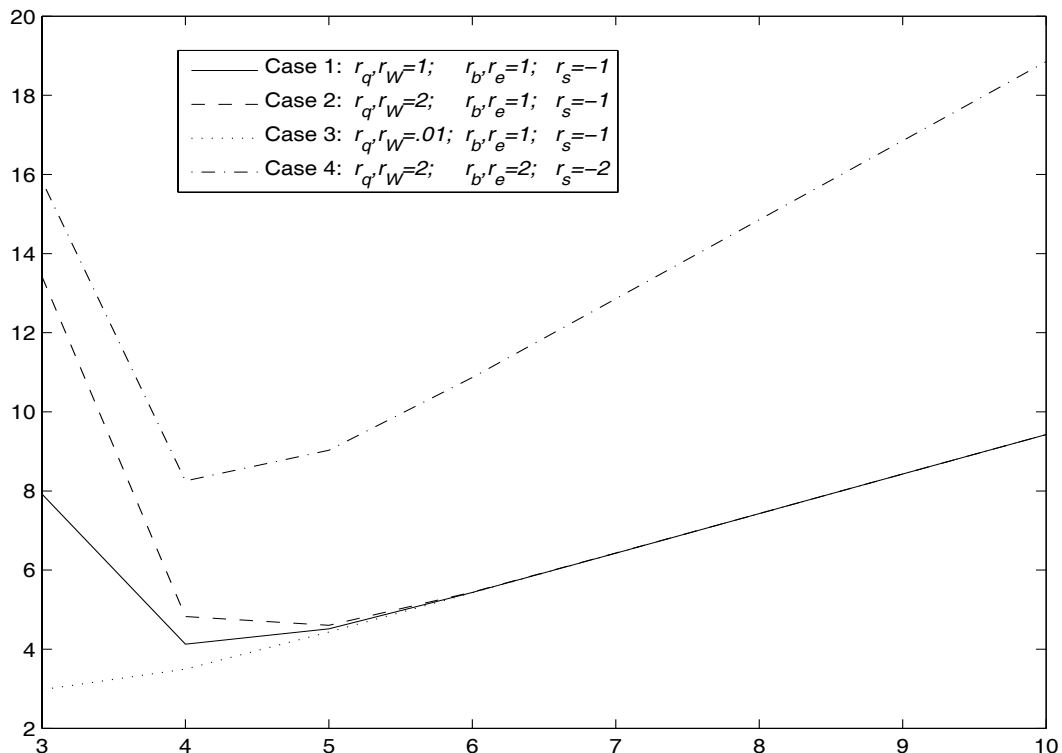


Figure 5: Estimated cost functions with different costs per u.t.

predictive means of $L_1(N_q)$ and $L_1(W)$ go to zero when n_q and w increase, respectively. However, again the optimum number of servers remains the same for all cases and is equal to 3 servers. Thus, we can conclude that, in general, there is larger sensitivity to the choice of loss functions than to the choice of thresholds.

c	$g(c \mathbf{t}, \mathbf{s}, \rho < c)$					
	$n_q = 40$ $w = 100$	$n_q = 20$ $w = 30$	$n_q = 5$ $w = 10$	$n_q = 5$ $w = 5$	$n_q = 1$ $w = 1$	$n_q = 0$ $w = 0.25$
3	2.4712	2.5518	2.8463	2.9726	3.4110	3.5861
4	3.4273	3.4276	3.4382	3.4579	3.6089	3.7211
5	4.4270	4.4270	4.4271	4.4284	4.4518	4.4797
6	5.4260	5.4260	5.4260	5.4261	5.4288	5.4333
7	6.4265	6.4265	6.4265	6.4265	6.4267	6.4273
8	7.4264	7.4264	7.4264	7.4264	7.4264	7.4264

Table 3: Estimated average cost per u.t. for different values of n_q and w using L_1 loss functions for N_q and W . Optimal values are indicated in bold.

7 Conclusions

In this paper, we have proposed a Bayesian approach for control of the number of servers c in a $GI/M/c$ system. We have developed a BDMCMC method based on mixtures of Erlang distributions to approximate

the general interarrival time distribution, performance measures have been predicted and incorporated into average cost functions to determine the optimal number of servers. This methodology have been illustrated with a real data set.

Our Bayesian approach can be extended to the $GI/G/c$ queue considering Erlang mixtures both for the interarrival and the service times. However, given the parameters in this case, there are not explicit expressions in the queueing literature for the stationary distributions which are straightforward to calculate. One possibility is to consider the phase type family of distributions (PH) introduced by Neuts (1981). Some known results of the $GI/PH/c$ model could be used as the Erlang mixture is a PH distribution. Similar ideas are implemented in Ausín et al. (2004) for the $M/PH/1$ queue.

A more general extension consists in the design of the $GI/G/c/K$ model, with $K \leq \infty$, where K is the system capacity. It is possible to extend the cost structure to queues with finite capacity by considering costs based on lost demand. An example for the particular case where the system capacity equals the number of servers can be found in Ausín et al. (2003).

Some modifications of our analysis could also be carried out. As commented before, the reversible jump technique introduced by Richardson and Green (1997) could be used as an alternative to the BDMCMC methodology. Also, we could have considered approximating the interarrival time with a mixture of gamma distributions which is a more flexible model; see Wiper et al. (2001). However, a disadvantage of this model is that the probability that a simpler model (exponential, Erlang or hyperexponential) cannot be easily calculated; see e.g. Ausín et al. (2004). Another disadvantage is that the gamma mixture is not PH which means that extension to more complex systems with this model is difficult.

Finally, there are also some alternatives to the cost structure defined. For example, costs per unit of time in the stationary state could be replaced by costs per busy cycle using the cycle criterion, see Lillo (2000).

Furthermore, an interesting extension of our cost analysis could be considering the entire posterior distribution of the cost function instead of using the expected value of the cost. Then, we could compare the different densities obtained for each number of servers by using e.g. the Kullback-Leibler distance. As a consequence, correlations between different components of costs could be considered.

Acknowledgements

We would like to thank two anonymous referees for their helpful comments (*Se agradece en este caso también al editor?*). We acknowledge the financial support partially provided by MEC project SEJ2004-03303 and by CAM project 06/HSE/0181/2004. The first author also acknowledges the financial support provided by Xunta de Galicia under the Isidro Parga Pondal Program.

References

- ALLEN, A.O. 1990. *Probability, Statistics and Queueing Theory with Computer Science Applications*. Academic Press, Boston.
- ARMERO, C., AND M.J. BAYARRI. 1996. Bayesian questions and answers in queues. J.M. Bernardo, J.O. Berger, A.P. Dawid, A.F.M. Smith, eds. *Bayesian Statistics 5*. Oxford University Press, Oxford, pp. 613-618.
- ARMERO, C., AND D. CONESA. 2000. Prediction in Markovian bulk arrival queues. *Queueing Systems*. 34, 327-350.
- AUSÍN, M.C. 2004. Análisis Bayesiano de colas. *PhD Thesis*. Universidad Carlos III de Madrid, Getafe, Madrid, Spain.
- AUSÍN, M.C., R.E. LILLO, F. RUGGERI AND M.P. WIPER. 2003. Bayesian Modeling of Hospital Bed Occupancy Times Using a Mixed Generalized Erlang Distribution. J.M. Bernardo, J.O. Berger, A.P. Dawid, M. West, eds. *Bayesian Statistics 7*. Oxford University Press, Oxford, pp. 443-452.
- AUSÍN, M.C., M.P. WIPER AND R.E. LILLO. 2004. Bayesian Estimation for the M/G/1 queue using a phase type approximation. *Journal of Statistical Planning and Inference*. 118, 83-101.
- BAGCHI, T.P., AND A.A. CUNNINGHAM. 1972. Bayesian approach to the design of queueing systems. *INFOR*. 10, 36-46.
- CAPPÉ, O., C.P. ROBERT AND T. RYDÉN. 2003. Reversible jump MCMC converging to birth-and-death MCMC and more general continuous time samplers. *Journal of the Royal Statistical Society, Series B*, 65, 679-700.
- DIEBOLT, J., AND C.P. ROBERT. 1994. Estimation of finite mixture distributions through Bayesian sampling. *Journal of the Royal Statistical Society, Series B*. 56, 363-375.
- GELFAND, A.E., AND A.F.M. SMITH. 1990. Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association*. 85, 398-409.
- GROSS, D., AND C.M. HARRIS. 1985. *Fundamentals of Queueing Theory*. John Wiley & Sons, New York.
- HASTINGS, W.K. 1970. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*. 57, 97-109.

- HURN, M., A. JUSTEL AND C.P. ROBERT. 2003. Estimating mixtures of regressions. *Journal of Computational and Graphical Statistics*. 12, 1-25.
- KITAEV, M.Y., AND V.V. RYKOV. 1995. *Controlled queueing systems*. CRC Press, Boca Raton, FL.
- LEHOCZKY, J. 1990. Statistical methods. D.P. Heyman and M.J. Sobel, eds. *Stochastic Models*. Elsevier/North-Holland, New York and Amsterdam, pp. 255-293.
- LILLO, R.E. 2000. On the optimal control of $M/G/1$ systems under the cycle criterion. *Systems & Control Letters*. 41, 29-39.
- LILLO, R.E., AND M. MARTIN. 2000. Characterization and computation of optimal policies for an $M/G/1$ priority queue. *The Belgian Journal of Operations Research, Statistics and Computer Science*. 38, 45-57.
- NEUTS, M.F. 1981. *Matrix Geometric Solutions in Stochastic Models*. Johns Hopkins University Press, Baltimore.
- RICHARDSON, S., AND P.J. GREEN. 1997. On Bayesian analysis of mixtures with an unknown number of components. *Journal of the Royal Statistical Society, Series B*. 59, 731-792.
- RÍOS, D., M.P. WIPER AND F. RUGGERI. 1998. Bayesian analysis of $M/Er/1$ and $M/H_k/1$ queues. *Queueing Systems*. 30, 289-308.
- ROBERT, C.P. 1996. Mixtures of distributions : inference and estimation. W.R. Gilks, S. Richardson, D.J. Spiegelhalter, eds. *Markov Chain Monte Carlo in Practice*. Chapman and Hall, London, pp. 441-464.
- STEPHENS, M. Bayesian analysis of mixture models with an unknown number of components — an alternative to reversible jump methods. *The Annals of Statistics*. 28, 40-74.
- WIPER, M.P. 1998. Bayesian analysis of $Er/M/1$ and $Er/M/c$ queues. *Journal of Statistical Planning and Inference*. 69, 65-79.
- WIPER, M.P., D. RÍOS AND F. RUGGERI. 2001. Mixtures of gamma distributions with applications. *Journal of Computational and Graphical Statistics*. 10, 440-454.