Journal of the Operational Research Society (1999) 50, 765–773 0160-5682/99 $12.00

http://www.stockton-press.co.uk/jor

A semidefinite programming approach to the optimal control of a single server queueing system with imposed second moment constraints

PS Ansell[1], KD Glazebrook[1], I Mitrani[1] and J Niño-Mora[2]

[1]*University of Newcastle upon Tyne, UK and* [2]*Universitat Pompeu Fabra, Spain*

Classical analyses of the dynamic control of multi-class queueing systems frequently yield simple priority policies as optimal. However, such policies can often result in excessive queue lengths for the low priority jobs/customers. We propose a stochastic optimisation problem in the context of a two class $M/M/1$ system which seeks to mitigate this through the imposition of constraints on the second moments of queue lengths. We analyse the performance of two families of parametrised heuristic policies for this problem. To evaluate these policies we develop lower bounds on the optimum cost through the achievable region approach. A numerical study points to the strength of performance of threshold policies and to directions for future research.

Keywords: achievable region; Gittins index; multi-class queueing system; semidefinite program; stochastic optimisation

Introduction

Much of the literature on the optimal dynamic control of multi-class queueing systems has sought to elucidate service policies which minimise some measure of holding cost rate for the system. In a range of contexts, simple priority policies in which the server(s) chooses from among the jobs/customers available for service according to a fixed ordering of the classes to which they belong have been shown to be optimal. Recently, Shanthikumar and Yao[1] followed by Bertsimas and Niño-Mora[2] have given a unified account of many such classical results, including those of Cobham,[3] Klimov,[4] Gittins[5] and Whittle.[6]

In the following section we describe a simple two class system which is amenable to such an analysis. We point to an undesirable property of the claimed optimal policy, namely that of excessive queue lengths for the low priority class. Service for this class tends not only to be poor on average, but also unpredictable. To mitigate this effect we moderate the usual stochastic optimisation problem by imposing constraints on the second moments of the queue lengths. We continue by charting our progress toward solving this important but exceedingly difficult problem. We propose and analyse two families of parametrised heuristic policies. We go on to discuss how lower bounds on optimal cost may be developed from the perspective of the achievable region approach. In brief, this methodology approaches stochastic optimisation problems by (i) attempting to characterise the region of achievable performance of the system (in this case, the achievable first and second moments of the queue lengths for both customer classes) under all service policies; and (ii) reformulating the stochastic optimisation problem of interest as a mathematical program with the achievable region as feasible space. This approach goes back to Coffman and Mitrani.[7] Other notable contributions have been due to Federgruen and Groenevelt,[8] Shanthikumar and Yao,[1] Bertsimas and Niño-Mora[2] and Garbe and Glazebrook.[9] For the problem of interest to us, we are able to develop relaxations of the required achievable region which in turn yield lower bounds on the best cost achievable. These can be used as yardsticks for the evaluation of heuristics. The lower bounds are derived from a semidefinite program. Finally, we conclude with a numerical study. This points both to the high quality of performance available from threshold policies and also to directions for future research.

Correspondence: Prof KD Glazebrook, School of Mathematics and Statistics, University of Newcastle, Newcastle upon Tyne NE1 7RU, UK.

The model

We consider a model of an $M/M/1$ queue in steady state, with two customer/job types being served by a single server. Initially, we measure the performance of the system by the pair of expected queue lengths $[E(N_1), E(N_2)]$. Service policies are assumed to be non-anticipative and work conserving, and the server is allocated to customers in a preemptive fashion. There are no penalties imposed when the server switches and all switches are deemed instantaneous. Arrivals to queue k form independent Poisson streams with parameters $\lambda_k, k = 1, 2$, and have exponentially distributed service times with means $1/\mu_k, k = 1, 2$. To ensure finite queue lengths we require that the traffic intensity (load on the system), $\rho = \lambda_1/\mu_1 + \lambda_2/\mu_2$ is less than unity. Under these conditions, Coffman and Mitrani[7] showed that the set of achievable pairs $[E(N_1), E(N_2)]$ satisfy the following:

$$\frac{1}{\mu_1}E(N_1) + \frac{1}{\mu_2}E(N_2) = \frac{\rho_1\mu_1^{-1} + \rho_2\mu_2^{-1}}{1-\rho_1-\rho_2}, \tag{1}$$

$$E(N_1) \geqslant \frac{\rho_1}{1-\rho_1}, \tag{2}$$

$$E(N_2) \geqslant \frac{\rho_2}{1-\rho_2}. \tag{3}$$

Since the expected remaining service requirement of each class k customer is equal to $\mu_k^{-1}, k = 1, 2$, the expression on the 1.h.s. of (1) is seen to be the expected work-in-system in the steady state. Hence (1) is a statement that the expected work-in-system is invariant for the class of service policies considered. Inequality (2) is a consequence of the fact that that the expected class 1 work-in-system is minimised by the priority policy $(1, 2)$ which serves class 2 customers only when there are no class 1 customers in the system and which imposes this priority in a preemptive fashion. Inequality (3) follows similarly when the roles of the two classes are reversed. Coffman and Mitrani[7] developed this analysis further by showing that the set of achievable $[E(N_1), E(N_2)]$ is exactly the line segment determined by (1)–(3), namely

$$H = \left[(x_1, x_2);\ x_1 \geqslant \frac{\rho_1}{1-\rho_1},\ x_2 \geqslant \frac{\rho_2}{1-\rho_2},\ \frac{x_1}{\mu_1} + \frac{x_2}{\mu_2} = \frac{\rho_1\mu_1^{-1} + \rho_2\mu_2^{-1}}{1-\rho_1-\rho_2}\right],$$

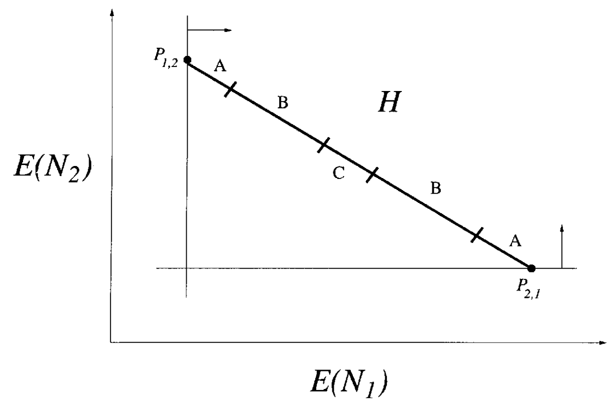

Figure 1 The performance region for $\{E(N_1), E(N_2)\}$ and the impact on it of second moment constraints.

as depicted in Figure 1. Note that the extreme points of H, namely $P_{1,2}$ and $P_{2,1}$ are simply the performances $[E(N_1), E(N_2)]$ obtained upon application of the priority policies $(1, 2)$ and $(2, 1)$ respectively.

Consider minimising a linear combination of average queue lengths

$$C = c_1E(N_1) + c_2E(N_2) \tag{4}$$

over all service policies. In (4) the c_k's can be thought of as holding rates. Referring to Figure 1, note that the cost function (4) is minimised at one of the extreme points. This restricts the search for the optimal service policy to a decision between the two static priority policies. More specifically, the customer class k, with the largest value of $c_k\mu_k$, is given highest priority (the so called $c\mu$-rule, see Gelenbe and Mitrani,[10] for example).

Notwithstanding this analysis, it can be argued that in many cases strict priority policies are practically unacceptable because of the large penalties which may be imposed on low priority jobs. In many of the simulation studies we have conducted, excessive queue lengths for the low priority class occur frequently. This results in particular in a very large second moment for the corresponding queue length and a consequentially large variance. The latter would indicate that in such cases not only is service poor on average, but unpredictably so. The following example is typical: consider a system with arrival rates $\boldsymbol{\lambda} = (0.5, 0.1)$ and service rates $\boldsymbol{\mu} = (1.4, 0.2)$. Table 1 contains the first and second moments for the queue lengths in this example. Note that the customers given top priority enjoy a high quality of service but the low priority jobs have to endure large first and second moments of their queue lengths.

A natural optimisation problem which reflects the concern expressed in the forgoing paragraph to mitigate excessive queue lengths for low priority jobs is one which seeks to minimise the linear objective (4) subject to second moment constraints of the form:

$$E(N_1^2) \leqslant v_1 \tag{5}$$

$$E(N_2^2) \leqslant v_2. \tag{6}$$

Plainly, other formulations are possible including those which directly constrain upper tail probabilities from the distributions of N_1 and N_2 and those which include second moments in the objective. However, as will emerge later, even the above problem with linear objective and second moment constraints presents a formidable technical

Table 1 First and second moments of queue lengths for the priority policies $(1, 2)$ and $(2, 1)$

	$E(N_1)$	$E(N_2)$	$E(N_1^2)$	$E(N_2^2)$
(1, 2)	0.56	3.78	1.17	32.42
(2, 1)	20.00	1.00	1170.00	3.00

challenge. Alternative approaches and formulations will be the subject of future work.

Extensive simulation studies indicate that constraining second moments as in (5) and (6) is subtly distinct from constraining first moments in some equivalent fashion. Roughly speaking, the imposition of (5) and (6) has the effect of reducing the achievable region H to some subset with the situation as depicted in Figure 1 as follows.

The performance pair $[E(N_1), E(N_2)]$ corresponding to specific policy u will

(1) NOT satisfy $E(N_1^2) \leqslant v_1$, $E(N_2^2) \leqslant v_2$ if the performance pair is in section A;
(2) POSSIBLY satisfy $E(N_1^2) \leqslant v_1$, $E(N_2^2) \leqslant v_2$ if the performance pair is in section B;
(3) CERTAINLY satisfy $E(N_1^2) \leqslant v_1$, $E(N_2^2) \leqslant v_2$ if the performance pair is in section C.

Therefore, the ultimate aim of this paper is to assist the search for practical service policies by:

(1) obtaining readily implementable and analytically tractable service policies which come close to minimising the linear cost function while satisfying constraints on second moments;
(2) characterising the complete set of achievable performance pairs

$$\tilde{H} = \{[E_u(N_1),\ E_u(N_2)]|\ \text{for those controls } u \text{ s.t}\ E_u(N_1^2) \leqslant v_1,\ E_u(N_2^2) \leqslant v_2\}$$

as closely as possible;
(3) optimising over a suitably chosen relaxation of $\tilde{H}$ to assess how close to optimal our proposed heuristic policies are.

We address the first of these problems by describing two classes of heuristic service policies. Both policy classes are characterised by a single parameter. This is an important feature of these policies as it allows the second moment constraints to be transformed into a parameter constraint, hence simplifying the consequential optimisation. Section 4 focuses attention on the second and third of the aims above. We determine sets of linear constraints that performances must satisfy and infer a relaxation of the achievable region. In section 5 we illustrate the results numerically. We are able to draw some conclusions about the quality of performance available from our heuristic policies.

Heuristic service policies

The problem of finding a suitable service policy can be approached most naturally by devising families of parametrised policies whose associated first moment performances span the entire set H and whose first and second moments are readily computable. Then second moment constraints (5) and (6) can be mapped to corresponding parameter constraints. If we choose the family of policies well, the restricted set of performance pairs will approximate $\tilde{H}$ and the selected policy will be close to optimal.

Randomised service policies

Consider a randomised service policy that depends on a single parameter α as follows: at the beginning of each busy period a random decision is made, independently of all other such decisions, which gives preemptive priority to type 1 with probability α, and to type 2 with probability $1-\alpha$. This policy operates for the whole of the current busy period. As α varies from 0 to 1, the policies in this family range from the priority policy $(2, 1)$ through to the priority policy $(1, 2)$. The corresponding set of performances $[E(N_1), E(N_2)]$ coincide with H. The first two moments of the number of type k jobs in the system are given by:

$$\begin{aligned} E_\alpha(N_k) &= \alpha E_{\{1,2\}}(N_k) + (1-\alpha)E_{\{2,1\}}(N_k) \\ E_\alpha(N_k^2) &= \alpha E_{\{1,2\}}(N_k^2) + (1-\alpha)E_{\{2,1\}}(N_k^2) \end{aligned} \tag{7}$$

where the subscripts $\{1,2\}$ and $\{2,1\}$ refer to application of the strict priority policies $(1,2)$ and $(2,1)$, respectively. There is an extensive literature on the analysis of strict priority policies (see Jaiswal,[11] for example).

Substitution of (7) into the constraint (5), leads to an inequality $\alpha \geqslant \alpha_1$ where $0 \leqslant \alpha_1 \leqslant 1$ if it is possible for a mixed policy to satisfy (5). In addition, substituting (7) into the constraint (6), leads to an inequality $\alpha \leqslant \alpha_2$, where $0 \leqslant \alpha_2 \leqslant 1$ if there are randomised policies that satisfy (6). If $\alpha_1 \leqslant \alpha_2$, then both constraints can be satisfied by a randomised policy and one of the two extreme policies (that is, with α_1 or α_2) is optimal within this class for our constrained problem.

Threshold policies

We also consider a family of policies of the threshold type, characterised by a parameter T. These can be regarded as polling policies since the customer class being served depends on the length of each queue. Denote the number of jobs in queue k at time t by $N_k(t)$. The service policy is preemptive in nature: the server is allowed to curtail service of the present customer in favour of a new arrival. The policy works as follows:

(1) if $N_1(t) > 0$ and $N_2(t) < T$ then a type 1 job is served;
(2) if $N_2(t) \geqslant T$, or if $0 < N_2(T) < T$ and $N_1(t) = 0$ then a type 2 job is served.

This family of policies has the property that as the value of T ranges from 1 to ∞ the corresponding policies range from the policy (2, 1) through to the policy (1, 2). We can define a dual family of threshold policies where:

(1) if $N_2(t) > 0$ and $N_1(t) < T$ then a type 2 job is being served;
(2) if $N_1(t) \geqslant T$, or if $0 < N_1(t) < T$ and $N_2(t) = 0$ then a type 1 job is being served.

A drawback of the threshold policies is that the parameter T is discrete. However, the range of performances achievable by the threshold policies is sufficiently well distributed throughout H for practical purposes. Intermediate performance values between those for $T = k$ and $T = k + 1$ can be obtained by randomization. We can evaluate the first two moments of any threshold policy. Plainly by inspection the optimal threshold policy can be obtained.

Solution methodologies for threshold policies

For both policy types we require to establish the joint steady state distribution

$$p_{i,j} = \lim_{t\to\infty} P[N_1(t) = i, N_2(t) = j], \qquad i, j = 0, 1, \ldots \tag{8}$$

The probabilities associated with the threshold policy with parameter T satisfy the following balance equations:

$$\begin{aligned}
&[\lambda_1 + \lambda_2 + \mu_1\delta(i > 0, j < T) + \mu_2\delta(j \geqslant T)\\
&\quad \mu_2\delta(i = 0, 0 < j < T)]p_{i,j}\\
&\quad = \lambda_1 p_{i-1,j} + \lambda_2 p_{i,j-1} + \mu_1\delta(j < T)p_{i+1,j}\\
&\quad + \mu_2\delta(j \geqslant T - 1)p_{i,j+1}\\
&\quad \mu_2\delta(i = 0, j < T - 1)p_{i,j+1}
\end{aligned} \tag{9}$$

where $p_{-1,j} = p_{i,-1} = 0$ and $\delta(B) = 1$ if B is true, 0 otherwise.

Before proceeding further, please note the following:

- the priority policy (2, 1) satisfies the balance equations with $T = 1$;
- to establish balance equations for the dual family of threshold policies we need only reverse the arrival and service parameters.

We employ two techniques in solving the balance equations (9). The first method involves the introduction of the generating functions:

$$g_j(x) = \sum_{i=0}^{\infty} p_{i,j}x^i, \; j = 0, 1, \ldots, T-1,$$

$$g(x, y) = \sum_{i=0}^{\infty}\sum_{j=T}^{\infty} p_{i,j}x^i y^{j-T}. \tag{10}$$

We can transform the balance equations for $j < T - 1$, into recurrence relations between the functions $g_j(x)$. This allows us to determine the functions, $g_0(x), g_1(x), \ldots, g_{T-2}(x)$, in terms of the constants $p_{0,0}, p_{0,1}, \ldots, p_{0,T-1}$. In addition we can transform the balance equations for $j = T - 1$, and for $j \geqslant T$. This allows us to establish $g_{T-1}(x)$ and $g(x, y)$. Having determined these generating functions (see Ansell *et al*[12]) we can use standard techniques to determine the first and second moments of the queue lengths, namely:

$$\begin{aligned}
E(N_1) &= \sum_{j=0}^{T-1} g_j'(1) + \frac{\partial}{\partial x}g(1, 1),\\
E(N_2) &= \sum_{j=1}^{T-1} jg_j(1) + \frac{\partial}{\partial y}g(1, 1) + Tg(1, 1),\\
E(N_1^2) &= \sum_{j=0}^{T-1} g_j''(1) + \frac{\partial^2}{\partial x^2}g(1, 1) + E(N_1),\\
E(N_2^2) &= \sum_{j=1}^{T-1} j^2 g_j(1) + \frac{\partial^2}{\partial y^2}g(1, 1)\\
&\quad + (2T + 1)\frac{\partial}{\partial y}g(1, 1) + T^2 g(1, 1).
\end{aligned}$$

Note that the derivatives at point (1, 1) involve determinacies of type 0/0, which are resolved *via* L'Hôpital's Rule.

This exact method for the evaluation of the queue length moments for the threshold policies is only practical to reasonable levels of accuracy for policies with a threshold value less than 10. For $T \geqslant 10$, the number of derivatives and limits required to be evaluated make the method impractical for use. To overcome this problem we use a numerical method called the Power Series Algorithm, see Blanc.[13] This method takes the recursively unsolvable balance equations (9) and transforms them into a recursively solvable set of equations via the introduction of an additional parameter. More specifically, we replace λ_i by $\lambda_i\rho$, where ρ is the traffic intensity. In the balance equations we can then express the probabilities $p_{i,j}$ in terms of the traffic intensity ρ, as follows:

$$p_{i,j} = \rho^{i+j}\sum_{k=0}^{\infty}\rho^k \hat{p}_{k,i,j}. \tag{11}$$

We now substitute from (11) into the balance equations (9) to obtain a recursive system of equations:

$$\begin{aligned}
&[\mu_1\delta(i > 0, j < T) + \mu_2\delta(j \geqslant T)\\
&\quad \mu_2\delta(i = 0, 0 < j < T)]\hat{p}_{k,i,j}\\
&\quad = -(\lambda_1 + \lambda_2)\delta(k > 0)\hat{p}_{k-1,i,j} + \lambda_1\hat{p}_{k,i-1,j}\\
&\quad + \lambda_2\hat{p}_{k,i,j-1} + \mu_1\delta(j < T, k > 0)\hat{p}_{k-1,i+1,j}\\
&\quad + \mu_2\delta(j \geqslant T - 1, k > 0)\hat{p}_{k-1,i,j+1}\\
&\quad + \mu_2\hat{p}_{k-1,i,j+1}\delta\,(i = 0, j < T - 1, k > 0)
\end{aligned} \tag{12}$$

To determine the coefficients we also need to use the normalising equation, which we express as

$$\sum_{k=0}^{\infty}\sum_{i=0}^{\infty}\sum_{j=0}^{\infty}\hat{p}_{k,i,j}\rho^{k+i+j}=1.$$

From this we obtain

$$\hat{p}_{000}=1 \tag{13}$$

and

$$\sum_{i+j+k=l}\hat{p}_{k,i,j}=0, l\geqslant 1. \tag{14}$$

We can now calculate the coefficients from (12)−(14). By setting ρ to 1 we return to the original formulation. An approximation is obtained by restricting the number of coefficients evaluated to those with a power of ρ less than a value K.

Features of the service policies

In this section, we illustrate the main characteristics of the above service policies *via* an example. Consider the system with arrival and service parameters $\boldsymbol{\lambda}=(1, 5)$ and $\boldsymbol{\mu}=(3, 12)$ respectively. Figure 2 illustrates how the first moment pairs $[E(N_1), E(N_2)]$ vary with α and T for the two classes of policies. Figure 3 shows the second moment pairs. In the latter, note that for the randomised policies the plot is linear in α in contrast with the convex nature of the plot for the threshold policies. The plots strongly suggest that if α and T are chosen such that the corresponding policies have identical first moments then the threshold policy will have more satisfactory (smaller) second moment performance. In this sense, a wider range of threshold policies will meet the constraints (5) and (6). Hence it is this class which offers more promise in solving the optimisation problems of interest.

Achievable performance region: an approximate characterisation

Thus far, we have focussed attention on performance analysis for specific families of service policies. Our aim here is to develop an approach to the assessment of how well these heuristic policies perform in relation to the best achievable for our optimisation problem which constrains

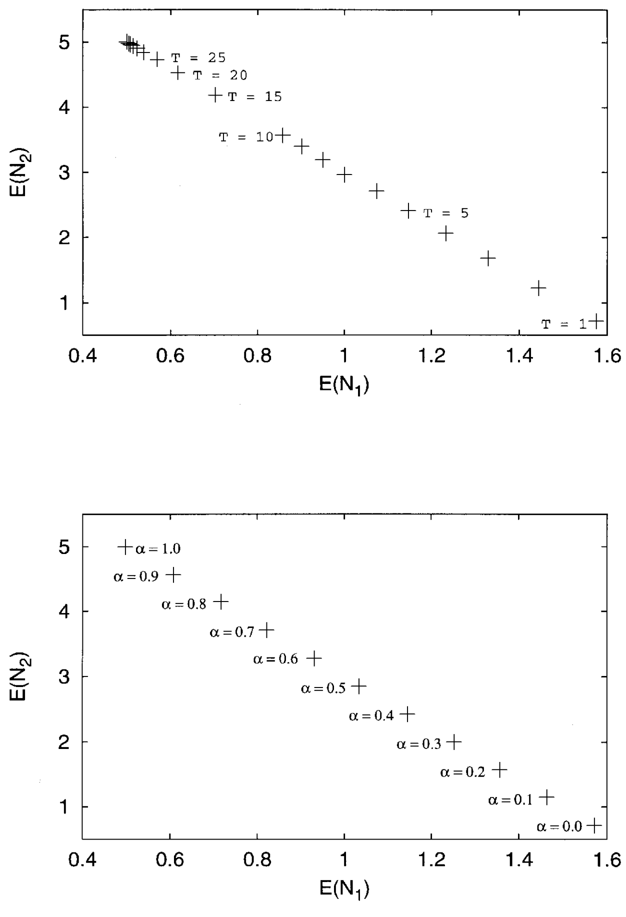

Figure 2 First moment pairs $\{E(N_1), E(N_2)\}$ for randomised and threshold service policies.

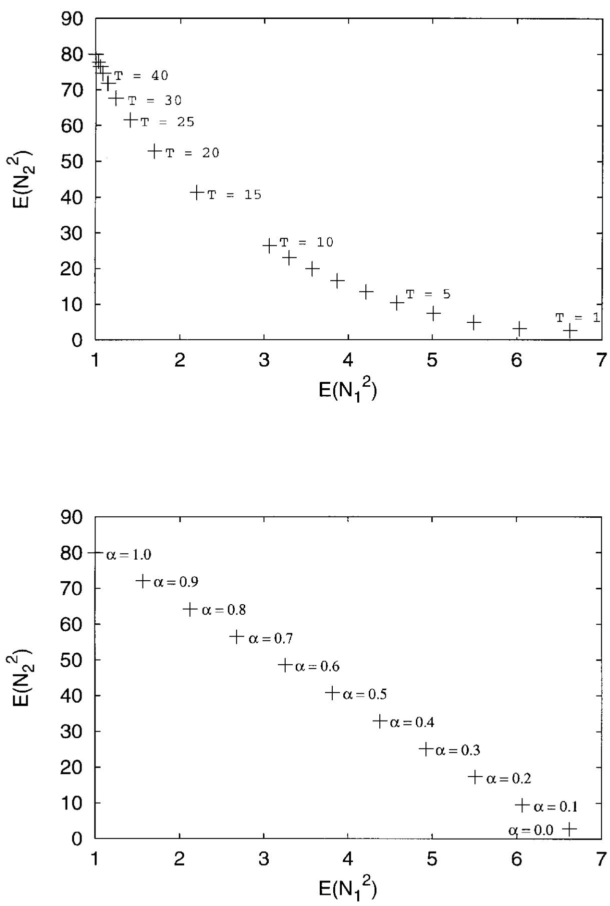

Figure 3 Second moment performance pairs $\{E(N_1^2), E(N_2^2)\}$ for randomised and threshold policies.

second moments. Our broad approach centres around the notion of an achievable region, though now our objective must be to characterise the set of achievable first and second moments of the queue lengths. In practice, we need to generate constraints on these quantities and most of this section will be devoted to doing that. These constraints will ultimately determine a relaxation of the required achievable region. Optimising over such a region will determine a lower bound on the optimal cost achievable under the second moment constraints.

We shall adopt a uniformisation which scales time such that

$$\lambda_1 + \lambda_2 + \mu_1 + \mu_2 = 1.$$

Let the sequence $\{\tau_k\}$ of random times be the times of successive events in a Poisson process of rate 1. These times are those at which state transitions in the system (caused by an arrival or a service completion) may happen. Service completions are disallowed when a customer of the requisite type is *not* being served; this is a standard construction. The function $\delta(A) = 1$ if event A occurs and zero otherwise. Moreover, $B_r(t)$ denotes that the server is busy with a type r customer at time t and $\bar{B}_r(t)$ is the complementary event. We require mixed moments evaluated at these times under the steady state distribution as follows:

$$\begin{aligned} n_r &= E[N_r(\tau_k)], \\ m_{rs} &= E[N_r(\tau_k)N_s(\tau_k)], \\ h_{rsw} &= E[\delta(B_r\{\tau_k\})N_s(\tau_k)N_w(\tau_k)], \\ i_{rs} &= E[\delta(B_r\{\tau_k\})N_s(\tau_k)] \end{aligned}$$

where $r, s, w = 1, 2$. Standard results indicate that such expectations evaluated at the epochs $\{\tau_k\}$ are equal to those taken under the steady state distribution at arbitrary time t. Note also that the quantities $m..$ and $i..$ reflect pairwise interactions among the customer classes while the $h\ldots$ reflect interactions among triples of classes.

We introduce (**n**, **m**, **h**, **i**) as the 18-vector with components (n_1, n_2, m_{11}, $\ldots$, i_{22}). Our goal is to describe as closely as we can the set of possible values of (**n**, **m**, **h**, **i**) as we range through the class of admissible service policies. To this end we develop equations and inequalities involving these quantities by the non-parametric bounding method of Bertsimas *et al.*[15] This method requires the introduction of the quantity

$$R(t) = f(1)N_1(t) + f(2)N_2(t)$$

where $N_r(t)$ denotes the number of type r jobs in the system at time t and the $f(r)$ are arbitrary constants.

We firstly explore pairwise interactions among the job classes by using the recursion:

$$\begin{aligned} E[R^2(\tau_{k+1})|\mathbf{N}(\tau_k)] = {} & \sum_{r=1}^{2} \lambda_r (R(\tau_k) + f(r))^2 \\ & + \sum_{r=1}^{2} \mu_r \delta(B_r\{\tau_k\})[(R(\tau_k) - f(r))^2] \\ & + \sum_{r=1}^{2} \mu_r \delta(\bar{B}_r\{\tau_k\}) R^2(\tau_k). \end{aligned}$$

Rearranging terms, taking expectations with respect to the invariant distribution, and equating coefficients of $[f(1)]^2, [f(1)f(2)]$ and $[f(2)]^2$, we obtain the following equations:

$$\mu_1 i_{11} - \lambda_1 n_1 = \lambda_1 \quad (15)$$

$$\mu_2 i_{22} - \lambda_2 n_2 = \lambda_2 \quad (16)$$

$$\mu_2 i_{21} + \mu_1 i_{12} - \lambda_2 n_1 - \lambda_1 n_2 = 0. \quad (17)$$

We also have the following trivial constraints:

$$\sum_{s=1}^{2} i_{1s} = n_1 \quad (18)$$

$$\sum_{r=1}^{2} i_{r2} = n_2 \quad (19)$$

$$i_{rs} \geqslant 0, \quad r = 1, 2 \quad s = 1, 2 \quad (20)$$

$$n_r \geqslant 0, \quad r = 1, 2. \quad (21)$$

We refer to the polyhedron of quantities satisfying (15)–(21) as P_2. It can be shown that (15)–(21) imply the exact achievable region for the first moment pairs described in (1)–(3) above and following. See Berstimas *et al.*[15]

The characterisation above only expresses pairwise interactions among the job classes. If we wish to incorporate second moments we require one which expresses higher order interactions. We capture these by examining a recursion for $E[R^3(\tau_{k+1})|\mathbf{N}(\tau_k)]$, which leads to a new set of linear constraints involving the variables (**n**, **m**, **h**, **i**). We have:

$$\begin{aligned} E[R^3(\tau_{k+1})|\mathbf{N}(\tau_k)] = {} & \sum_{r=1}^{2} \lambda_r (R(\tau_k) + f(r))^3 \\ & + \sum_{r=1}^{2} \mu_r \delta(B_r\{\tau_k\})[(R(\tau_k) - f(r))^3] \\ & + \sum_{r=1}^{2} \mu_r \delta(\bar{B}_r\{\tau_k\}) R^3(\tau_k). \end{aligned}$$

As before, expanding these cubics, rearranging terms, taking expectations and equating coefficients we obtain the following:

$$\lambda_1 n_1 + \lambda_1 m_{11} + \mu_1 i_{11} - \mu_1 h_{111} = 0 \tag{22}$$

$$\lambda_2 n_2 + \lambda_2 m_{22} + \mu_2 i_{22} - \mu_2 h_{222} = 0 \tag{23}$$

$$\lambda_1 m_{22} + \lambda_2 n_1 + 2\lambda_2 m_{12} + \mu_2 i_{21} - \mu_1 h_{122} - 2\mu_2 h_{221} = 0 \tag{24}$$

$$\lambda_2 m_{11} + \lambda_1 n_2 + 2\lambda_1 m_{12} + \mu_1 i_{12} - \mu_2 h_{211} - 2\mu_1 h_{112} = 0 \tag{25}$$

The following constraints are trivial:

$$\sum_{r=1}^{2} h_{r12} - m_{12} = 0 \tag{26}$$

$$\sum_{r=1}^{2} h_{r11} - m_{11} = 0 \tag{27}$$

$$\sum_{r=1}^{2} h_{r22} - m_{22} = 0 \tag{28}$$

$$n_r, m_{rs}, i_{rs}, h_{rst} \geqslant 0. \tag{29}$$

The polyhedron of quantities satisfying the (in) equalities (22)–(29) will be called P_3.

Since all of the equations and inequalities above are satisfied by the mixed moments derived from all service policies it follows that the set of all achievable ($\mathbf{n}$, $\mathbf{m}$, $\mathbf{h}$, $\mathbf{i}$) is contained in $P_2 \cap P_3$. However, our computational experience indicates that this relaxation of the achievable region we require is insufficiently tight to be of assistance. We therefore need additional constraints to strengthen the approximation. These are described in the following subsections.

Semidefinite constraints

It is standard that for any random vector ψ with the requisite moments existing, $E[\psi\psi'] - E[\psi][E(\psi)]'$ is a symmetric positive semidefinite matrix. Suitable application of this yields the conclusion that:

$$D_1 = \begin{pmatrix} 1 & n_1 & n_2 \\ n_1 & m_{11} & m_{12} \\ n_2 & m_{12} & m_{22} \end{pmatrix} \succeq 0, \tag{30}$$

$$D_2 = \begin{pmatrix} 1 & i_{11} & i_{12} \\ i_{11} & h_{111} & h_{112} \\ i_{12} & h_{112} & h_{122} \end{pmatrix} \succeq 0, \tag{31}$$

$$D_3 = \begin{pmatrix} 1 & i_{21} & i_{22} \\ i_{21} & h_{211} & h_{212} \\ i_{22} & h_{212} & h_{222} \end{pmatrix} \succeq 0 \tag{32}$$

where $A \succeq 0$ indicates that A is positive semidefinite.

Workload constraints

Standard work conservation arguments such as those in Gelenbe and Mitranl[10] yield the conclusion that all service policies have the same corresponding work-in-system in a stochastic sense. Hence, the steady state variance of work-in-system must be an invariant of the process. If we denote by X and Y a generic residual service time for class 1 and class 2 respectively then $X \sim \exp(\mu_1)$ and $Y \sim \exp(\mu_2)$. The steady state variance of work-in-system may then be written

$$\operatorname{var}\left(\sum_{i=1}^{N_1} X_i + \sum_{j=1}^{N_2} Y_j\right) = \frac{m_{11}}{\mu_1^2} + \frac{2m_{12}}{\mu_1\mu_2} + \frac{m_{22}}{\mu_2^2} + \frac{n_1}{\mu_2^2} + \frac{n_2}{\mu_2^2} - \left(\frac{n_1}{\mu_1} + \frac{n_2}{\mu_2}\right)^2,$$

by standard results. If we utilise (1) then we conclude that

$$\frac{m_{11}}{\mu_1^2} + \frac{2m_{12}}{\mu_1\mu_2} + \frac{m_{22}}{\mu_2^2} + \frac{n_1}{\mu_2^2} + \frac{n_2}{\mu_2^2} = K. \tag{33}$$

The constant K in (33) can be obtained easily from methods due to Takács.[16]

Lower bound on the optimum cost via semidefinite programming

It is a consequence of the above that the set of achievable ($\mathbf{n}$, $\mathbf{m}$, $\mathbf{h}$, $\mathbf{i}$) is contained in $P_2 \cap P_3$ and also satisfies (30)–(33). Hence the semidefinite program

$$\min\ c_1 n_1 + c_2 n_2 \tag{34}$$

subject to

$$[(\mathbf{n}, \mathbf{m}, \mathbf{h}, \mathbf{i})] \in P_2 \cap P_3, \tag{35}$$

$$D_1 \succeq 0, D_2 \succeq 0, D_3 \succeq 0, \tag{36}$$

$$\frac{m_{11}}{\mu_1^2} + \frac{2m_{12}}{\mu_1\mu_2} + \frac{m_{22}}{\mu_2^2} + \frac{n_1}{\mu_1^2} + \frac{n_2}{\mu_2^2} = K. \tag{37}$$

$$m_{11} \leqslant v_1, m_{22} \leqslant v_2 \tag{38}$$

will yield a lower bound on the cost C (see (4)) achievable by any service policy meeting the second moment constraints (5) and (6). This lower bound can then be used as a tool for evaluation of our heuristic policies. The solution of the semidefinite program (34)–(38) was obtained using the interior point algorithm and software package (SDPA) developed by Kojima.[17] See Vandenberghe and Boyd[18] for an extensive review of semidefinite programming techniques and algorithms.

Numerical example

Customers arrive at rates $\boldsymbol{\lambda} = (1, 5)$ and are served at rates $\boldsymbol{\mu} = (3, 12)$, respectively. Take the cost vector to be $\mathbf{c} = (10, 1)$. We have $c_1\mu_1 \geqslant c_2\mu_2$, and hence the priority policy (1,,2) is optimal for the unconstrained problem. Now consider a problem in which the second moment of type 2 jobs is constrained. We wish to compare the randomised policies with the threshold policies as solutions to our stochastic optimisation problem and to compare the performance of both with the bound on what is achievable obtained from the above semidefinite program. Table 2 contains the results obtained when thresholds and constraints are only imposed on the type 2 jobs. Note that the values of v_2 have been chosen so that the second moment constraint may be met with equality by a threshold policy. The calculations were repeated with **c** now taken to be (5, 1). This second example modifies the first in bringing the value of $c_1\mu_1$ much closer to $c_2\mu_2$. However, as will become clear the qualitative nature of most of the conclusions are substantially unchanged by this modification.

For each choice of v_2 and **c** Table 2 records the best cost achievable by a randomised policy (α^{COST}), the best by a threshold policy (T^{COST}) and the lower bound on achievable cost (SD^{COST}) determined from appropriate versions of the semidefinite program (34)–(38). The main features of Table 2 are transparent. For each value of v_2 and both choices of c we have $\alpha^{\mathrm{COST}} \geqslant T^{\mathrm{COST}} \geqslant SD^{\mathrm{COST}}$, indicating that threshold policies are uniformly (over v_2) better than randomised policies for the problems of interest. Indeed, closer inspection of the values shows that for all v_2 we have $\alpha^{\mathrm{COST}} - T^{\mathrm{COST}} \geqslant T^{\mathrm{COST}} - SD^{\mathrm{COST}}$, demonstrating that the degree of improvement available from threshold policies (as measured by distance from the lower bound) is very considerable. In the worst case T^{COST} is a little more that 20% greater that SD^{COST} (though usually is much closer to the lower bound than this). This occurs in the case $\mathbf{c} = (10,\ 1)$. The percentage cost suboptimalities are smaller in the $\mathbf{c} = (5,\ 1)$ case where T^{COST} is never more than 5% above SD^{COST}. Our view is that even these figures overstate the true degree of suboptimality of the threshold policy and point to the need for continuing work on developing improved lower bounds through tighter relaxations of the achievable region.

Table 2 Comparison between optimal randomised policy, optimal threshold policy (thresholds imposed on type 2 jobs) and semidefinite programming lower bound for the case $\lambda = (1,\ 5)$, $\mu = (3,\ 12)$; $\mathbf{c} = (10,\ 1)$ and (5, 1)

T	$E[N_2^2] \leqslant v_2$	$\alpha^{\mathrm{COST}}_{(10,1)}$	$T^{\mathrm{COST}}_{(10,1)}$	$SD^{\mathrm{COST}}_{(10,1)}$	$\alpha^{\mathrm{COST}}_{(5,1)}$	$T^{\mathrm{COST}}_{(5,1)}$	$SD^{\mathrm{COST}}_{(5,1)}$
1	1.735	16.424	16.424	16.424	8.569	8.569	8.569
2	2.986	16.326	15.660	15.401	8.554	8.445	8.400
3	4.983	16.162	14.981	14.277	8.527	8.331	8.216
4	7.501	15.959	14.392	13.392	8.492	8.232	8.067
5	10.368	15.719	13.882	12.667	8.453	8.147	7.947
6	13.459	15.486	13.439	11.989	8.411	8.074	7.834
7	16.679	15.201	13.048	11.367	8.367	8.008	7.726
8	19.958	14.932	12.705	10.792	8.322	7.950	7.629
9	23.240	14.662	12.405	10.262	8.277	7.900	7.541
10	26.487	14.395	12.141	10.000	8.233	7.856	7.500
15	41.332	13.176	11.211	10.000	8.029	7.701	7.500
20	53.029	12.215	10.905	10.000	7.869	7.615	7.500
25	61.616	11.510	10.401	10.000	7.752	7.565	7.500
30	67.672	11.013	10.233	10.000	7.669	7.538	7.500
35	71.833	10.671	10.138	10.000	7.612	7.522	7.500
40	74.641	10.440	10.086	10.000	7.573	7.516	7.500
45	76.509	10.286	10.048	10.000	7.548	7.508	7.500
50	77.741	10.186	10.031	10.000	7.531	7.506	7.500
∞	80.000	10.000	10.000	10.000	7.500	7.500	7.500

Conclusions

The paper has highlighted a weakness in performance of conventional priority policies for multi-class queueing systems in the form of a propensity to produce excessive queue lengths for low priority customer classes. We propose a stochastic optimisation problem which constrains second moments as a means to mitigating this effect. Two families of parametrised heuristic policies are analysed and assessed as solutions to the stochastic optimisation problem by means of the achievable region approach. Evidence of strong performance by a class of threshold policies are adduced. Further work will include

(i) consideration of formulations other than the current one which constrains second moments;
(ii) further analysis which seeks to develop improved bounds on the best cost achievable via the development of tighter relaxations of the achievable region of first and second moments;
(iii) study of more complex systems.

Acknowledgements—We would like to express our appreciation to the Engineering and Physical Sciences Research Council for supporting the work of the first author by means of a research studentship and the second author through the award of grants GR/K03043 and GR/M09308. We are grateful to the referees for their helpful comments.

References

1 Shanthikumar JG and Yao DD (1992). Multi-class queueing systems: polymatroidal structure and optimal scheduling control. *Opns Res* **40**: 293–299.
2 Bertsimas D and Niño-Mora J (1996). Conservation laws, extended polymatroids and multi-armed bandit problems: a polyhedral approach to indexable systems. *Math Opns Res* **21**: 257–306.
3 Cobham A (1954). Priority assignment in waiting line problems. *Opns Res* **2**: 70–76.
4 Klimov GP (1974). Time sharing systems I. *Theory Probab Appl* **19**: 532–551.
5 Gittins JC (1979). Bandit processes and dynamic allocation indicies. *J R Statist Soc B* **41**: 148–164.
6 Whittle P (1980). Multi-Armed Bandits and the Gittins Index. *J R Statist Soc B* **42**: 143–149.
7 Coffman E and Mitrani I (1980). A characterization of waiting time performance realizable by single server queues. *Opns Res* **28**: 810–821.
8 Federgruen A and Groenevelt H (1988). Characterisation and optimisation of achievable performance in queueing systems. *Opns Res* **36**: 733–741.

9 Garbe R and Glazebrook KD (1998). Submodular returns and greedy heuristics for queueing scheduling problems. *Opns Res* **46**: 336–346.
10 Gelenbe E and Mitrani I (1980). *Analysis and Synthesis of Computer Systems*. Academic Press: London.
11 Jaiswal NK (1968) *Priority Queues*. Academic Press: New York.
12 Ansell PS, Glazebrook KD and Mitrani I (1996). Server allocation subject to variance constraints. *Perform Eval* **27** and **28**: 147–158.
13 Blanc JPC (1993). Performance analysis and optimization with the Power-Series Algorithm. In: L. Donatiello and Nelson R (eds). *Performance Evaluation of Computer and Communications Systems*. Lecture Notes in Computer Science **729** Springer-Verlag, pp 53–80.
14 Ansell PS, Glazebrook KD and Mitrani I (1998). Threshold policies for a single server queueing network. Technical Report. University of Newcastle upon Tyne.
15 Bertsimas D, Paschalidis I and Tsitsiklis JN (1994). Optimisation of multi-class queueing networks: polyhedral and nonlinear characterizations of achievable performance. *Ann Appl Probab* **4**: 43–75.
16 Takács L (1968). Two queues attended by a single server. *Opns Res* **16**: 639–650.
17 Kojima M (1994). A primitive interior point algorithm for semidefinite programs in *Mathematica*, Research Report #293, Dept. of Mathematical and Computing Sciences, Tokyo Institute of Technology, Oh-Okayama, Meguro, Tokyo 152, Japan.
18 Vandenberghe L and Boyd S (1996). Semidefinite Programming. *SIAM Rev* **38**: 45–95.

Received October 1998;
accepted February 1999 after one revision