The Achievable Region Approach to the Optimal Control of Stochastic Systems*

Marcus Dacre[1], Kevin Glazebrook[1] and José Niño-Mora[2]

[1] *Department of Statistics, Newcastle University,*
Newcastle-upon-Tyne NE1 7RU, UK.
E-mail: kevin.glazebrook@newcastle.ac.uk

[2] *Department of Economics and Business, Universitat Pompeu Fabra,*
E-08005 Barcelona, Spain.
E-mail: jose.nino-mora@econ.upf.es

June 1998

*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 for supporting the work of the second author through the award of grants GR/K03043 and GR/M09308. We would also like to thank colleagues in the Department of Statistics, Newcastle University for their constructive comments on earlier drafts of the paper. The work of the third author was initiated during his stay at the Center for Operations Research and Econometrics (CORE) of the Université catholique de Louvain, Belgium, where it was supported by EC individual Marie Curie Postdoctoral Fellowship no. ERBFMBICT961480. Further research support is acknowledged from Universitat Pompeu Fabra.

Abstract

The achievable region approach seeks solutions to stochastic optimisation problems by: (i) characterising the space of all possible performances (the achievable region) of the system of interest, and (ii) optimising the overall system-wide performance objective over this space. This is radically different from conventional formulations based on dynamic programming. The approach is explained with reference to a simple two-class queueing system. Powerful new methodologies due to the authors and co-workers are deployed to analyse a general multi-class queueing system with parallel servers and then to develop an approach to optimal load distribution across a network of interconnected stations. Finally, the approach is used for the first time to analyse a class of intensity control problems.

Keywords: Achievable region, Gittins index, linear programming, load balancing, multi-class queueing systems, performance space, stochastic optimisation, threshold policy.

JEL: C60, C61.

1 Introduction

The last decade has seen a substantial research focus on the modelling, analysis and optimisation of complex stochastic service systems, motivated in large measure by applications in areas such as computer and telecommunication networks. Optimisation issues which broadly focus on making the best use of limited resources are recognised as of increasing importance. However, stochastic optimisation in the context of systems of any complexity is technically very difficult.

For the most part, the optimal dynamic control of queueing and other stochastic systems has been approached via dynamic programming (DP) formulations. Within such formulations, a variety of special arguments (of which the simplest and most effective have been interchange arguments) have been adduced to obtain structural results concerning optimal controls. A good summary of how things stood in the mid-to-late 80's can be found in chapters 8 and 9 of Walrand (1988). It would not be unfair to claim that a consensus view of this enterprise is that, with the exception of one or two notable successes (including the discovery and development of the Gittins index - see, for example, Gittins (1979,1989), Glazebrook (1982), Weber (1992), Weiss (1988) and Whittle (1980)) there was relatively little to show for a great deal of effort and that a pressing need existed for new approaches.

Many of the most important recent developments in the control, for example, of multi-class queueing networks have sought to optimise some associated/limiting process, whether a diffusion process (Brownian system model) in heavy traffic (see, for example, Harrison and Nguyen (1993) and Harrison and Wein (1989)) or a fluid model (see, for example, Atkins and Chen (1995) and Maglaras (1997)). These are powerful methodologies and have rightly been very influential. However, since the main focus of optimisation is an approximating/limiting process there can be formidable challenges in the subsequent derivation of controls for the queueing system of original interest and in the evaluation of such controls. See Harrison (1996) and Maglaras (1997).

The paper concerns a different approach - namely, the so-called achievable region or mathematical programming approach. It is possible that this could ultimately turn out to be more limited in its range of applications than those cited above (although the current pace of development throughout the field makes a final judgement impossible). However, it does have the considerable advantage of staying in firm contact with the original stochastic system of interest throughout. Hence when analyses via this methodology are available, they typically make clear and strong statements about the control policies identified.

The achievable region approach seeks solutions to stochastic optimisation problems by: (i) characterising the space of all possible performances (the achievable region) of the stochastic system, and (ii) optimising the overall system-wide performance objective over this space. The performance space

in (i) is often a polyhedron of special structure, yielding in (ii) a mathematical program (usually a linear program (LP)) for which efficient algorithms exist. The earliest contributions in this vein were due to Gelenbe and Mitrani (1980), followed by Federgruen and Groenevelt (1988). Contributions by Shanthikumar and Yao (1992) and Bertsimas and Niño-Mora (1996) took the approach decisively further forward, the latter giving an account of Gittins indexation from this perspective.

Our goal is, firstly, to bring the achievable region approach to the attention of a wider audience than it has enjoyed hitherto. To this end, many of the ideas alluded to in the previous paragraph are presented in Section 2 in a way which we trust will be widely accessible. In addition a range of powerful new methodologies with which the authors and co-workers have been associated are described and illustrated by the discussions in Sections 3-5 of a range of important stochastic optimisation problems. This material is new and should convey something of the power and scope of the achievable region approach. Given a familiarity with the content of Section 2, the later sections are self-contained with Section 3 the most demanding technically. Section 3 discusses the status of index policies for a general multi-class queueing system with servers working in parallel. We consider in Section 4 an approach to distributing the workload across a network of interconnected stations when each station is assumed to schedule its own offered load optimally. The problem of controlling input and output rates for a simple queueing system is discussed in Section 5. The paper concludes in Section 6 with proposals for future work.

2 The achievable region approach

For definiteness, we shall develop the core ideas underlying the achievable region approach in the context of multi-class queueing systems. Let $E = \{1, 2, \ldots, N\}$ denote a set of *customer classes*. *Customers* in the system require service which is provided by a collection of *servers*. A *control* u is a rule for determining how servers should be assigned to waiting customers. The set of *admissible controls* is denoted $\mathcal{U}$. Although admissibility will be defined in context, it will invariably be required that controls should be *non-anticipative* (decisions are made on the basis of the history of the process only) and *non-idling* (servers should never be idle when there is work to be done). With each control u is associated a *system performance vector* $\boldsymbol{x}^u = (x_1^u, x_2^u, \ldots, x_N^u)$ with x_i^u denoting the *class i performance*, $i \in E$. Throughout the paper, x_i^u will be the expectation of some quantity related to class i. A standard choice for x_i^u, denoted $E_u(N_i)$, is the long-term average number of class i customers in the system under control u. The *performance space* is the set of all possible performances, denoted $X = \{\boldsymbol{x}^u, u \in \mathcal{U}\}$. There is a cost $c(\boldsymbol{x}^u)$ associated with operating the system under control u which depends upon the control only through its associated performance. The stochastic optimisation problem of interest is expressed as

$$Z^{OPT} = \inf_{u \in \mathcal{U}} c(\boldsymbol{x}^u) \tag{1}$$

The prime goal is the identification of a control u^{OPT} attaining the infimum in (1). If X is known, an alternative computation of Z^{OPT} is via the minimisation

$$Z^{OPT} = \inf_{\boldsymbol{x} \in X} c(\boldsymbol{x}) \tag{2}$$

In all of the cases we shall consider we shall have $c(\boldsymbol{x}) = \boldsymbol{c}^T \boldsymbol{x}$ for some cost vector $\boldsymbol{c}$ and X a convex polyhedron, yielding in (2) a linear program (LP). Solution of (2) will yield $\boldsymbol{x}^{OPT}$, the optimising performance. The question then arises of whether a control u^{OPT} can be found which realises $\boldsymbol{x}^{OPT}$.

The *achievable region approach* seeks solutions to stochastic optimisation problems as in (1) by (i) identification of the performance space X, (ii) solution of a mathematical programming problem as in (2) with feasible space X and (iii) identification of controls yielding the optimum performance. This agenda can be fully realised in the case of *indexable systems*. To give the reader some idea of how the approach might proceed, we outline very briefly the case of a two class M/M/1 queueing system, first analysed in this manner by Coffman and Mitrani (1980).

Customers of class k arrive at a single server according to independent Poisson streams of rate λ_k with service requirements (independent of each other and of the arrival streams) which are exponentially distributed with mean μ_k^{-1}, $k = 1, 2$. The rate at which work arrives in the system is $\lambda_1/\mu_1 + \lambda_2/\mu_2$ which is assumed to be less than 1 (the available service rate) to guarantee stability, i.e. that the time-average number of customers in the system is finite. Controls for the system must be non-anticipative and non-idling and priorities between customer classes may be imposed preemptively (i.e. a customer whose requirements have not yet been fully met may be removed from service to make way for another customer of higher priority). The goal is to choose a control u to minimise a long-term holding cost rate, i.e.

$$\inf_{u \in \mathcal{U}} \{c_1 E_u(N_1) + c_2 E_u(N_2)\} \tag{3}$$

In (3) c_k is a cost rate, N_k is the number of class k customers in the system and E_u denotes an expectation taken under the steady state distribution when control u is applied. The achievable region approach solves the stochastic optimisation problem in (3) by proceeding through the above steps (i)-(iii) as follows:

(i) *Identification of the performance space X*

By a fairly simple standard argument it follows that in the steady state, the expected work (i.e. uncompleted processing) in the system is control invariant. In this particular case the constant concerned is easily identified and we have

$$\frac{E_u(N_1)}{\mu_1} + \frac{E_u(N_2)}{\mu_2} = \frac{(\rho_1\mu_1^{-1} + \rho_2\mu_2^{-1})}{(1-\rho_1-\rho_2)}, \quad u \in \mathcal{U}, \tag{4}$$

where the key quantity $\rho_k = \lambda_k/\mu_k$, $k = 1, 2$, has an interpretation as the rate at which class k work enters the system. In addition to (4), we note that the amount of class k work in the system can be minimised by giving class k customers (preemptive) priority over non-k customers. This yields

$$\frac{E_u(N_1)}{\mu_1} \geq \frac{\rho_1\mu_1^{-1}}{(1-\rho_1)}, \quad u \in \mathcal{U}, \tag{5}$$

$$\frac{E_u(N_2)}{\mu_2} \geq \frac{\rho_2\mu_2^{-1}}{(1-\rho_2)}, \quad u \in \mathcal{U}, \tag{6}$$

with the right hand sides of (5) and (6) attained when the system is controlled by the priority policies $1 \to 2$ and $2 \to 1$ respectively. Motivated by (4)-(6), we take $x_k^u = E_u(N_k)/\mu_k$ as the class k performance associated with control u, $k = 1, 2$. From (4)-(6), it immediately follows that performance space $X = \{(x_1^u, x_2^u),\ u \in \mathcal{U}\}$ is contained within the line segment P given by

$$P = \left\{(x_1, x_2);\ x_1 \geq \frac{\rho_1\mu_1^{-1}}{(1-\rho_1)},\ x_2 \geq \frac{\rho_2\mu_2^{-1}}{(1-\rho_2)},\ x_1 + x_2 = \frac{(\rho_1\mu_1^{-1} + \rho_2\mu_2^{-1})}{(1-\rho_1-\rho_2)}\right\} \tag{7}$$

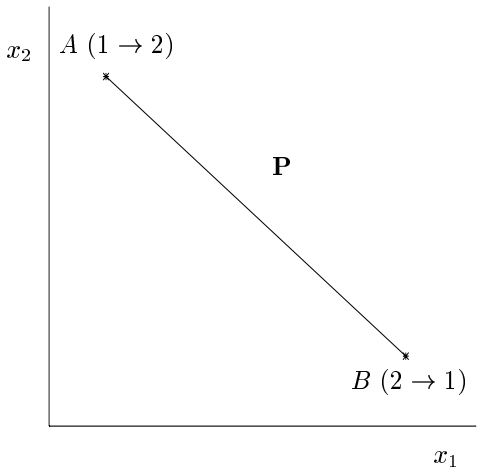

Figure 1: The line segment P

See Figure 1. To show that $P \subseteq X$, observe that the end-points of P, labelled A and B, may be identified as the performances associated with the priority policies $1 \to 2$ and $2 \to 1$ respectively. This follows from the remarks after (6). Any point of P is a convex combination of A and B and hence is easily seen to be the performance of a suitable randomisation of the policies $1 \to 2$ and $2 \to 1$. Hence all points of P are performances, as required. We conclude that $X = P$.

(ii) *Solution of an LP with feasible space X*

For reasons that will soon become clear, consider the LP

$$Z^{OPT} = \inf_{\boldsymbol{x} \in P} \{c_1\mu_1 x_1 + c_2\mu_2 x_2\}. \tag{8}$$

It is trivial to show that the minimum is attained at end-point A when $c_1\mu_1 \geq c_2\mu_2$ and at end-point B otherwise.

(iii) *Solution of the stochastic optimisation problem of interest*

Our objective is to identify a control u^{OPT} to solve (3), rewritten as

$$Z^{OPT} = \inf_{\boldsymbol{u} \in \mathcal{U}} \{c_1\mu_1 x_1^u + c_2\mu_2 x_2^u\}. \tag{9}$$

Since $X = P$, the quantities in (8) and (9) are equal. However from (ii) above, the $\boldsymbol{x}^{OPT}$ which solves the LP in (8) is known and we can readily identify a u^{OPT} giving rise to this performance. When $c_1\mu_1 \geq c_2\mu_2$, $\boldsymbol{x}^{OPT} = A$ and a control achieving this is $1 \to 2$. When $c_2\mu_2 \geq c_1\mu_1$, $\boldsymbol{x}^{OPT} = B$ and this is achieved by $2 \to 1$. We thus conclude that the control solving (9) is the so-called *$c\mu$-rule* which gives priority to the customer class with the larger $c_k\mu_k$-value. Hence the optimal control favours options which drive down the holding cost rate most rapidly.

The analysis for so-called indexable systems generalises that above as follows: equation (4) is replaced by a *generalised work conservation law* for the entire set E of customer classes, given by

$$\sum_{j \in E} V_j^E x_j^u = b(E), \quad u \in \mathcal{U}, \tag{10}$$

with V_j^E, $j \in E$ a suitably chosen set of positive constants. To generalise (5) and (6) suitably, we have to consider an arbitrary subset S of customer classes. We then have

$$\sum_{j \in E} V_j^S x_j^u \geq b(S), \quad u \in \mathcal{U} \tag{11}$$

for suitably chosen positive constants V_j^S, $j \in S$, with the right hand side of (11) attained by any control which gives customers in S priority over those not in S. This latter requirement is expressed by

$$\sum_{j \in E} V_j^S x_j^\pi = b(S) \quad \text{for } \pi : S \to S^c \tag{12}$$

Note that (11) and (12) need to hold for all proper subsets of E. Bertsimas and Niño-Mora (1996) referred to (10)-(12) as *generalised conservation laws* (GCL). They were able to show that when

performances are positive-valued and X is convex, a GCL system has performance space given by the convex polyhedron

$$P = \left\{ \boldsymbol{x} \in (\Re^+)^N;\ \sum_{j \in E} V_j^S x_j \geq b(S),\ S \subset E, \text{ and } \sum_{j \in E} V_j^E x_j = b(E) \right\}. \tag{13}$$

We suppose that the stochastic optimisation problem of interest can be expressed as

$$Z^{OPT} = \inf_{u \in \mathcal{U}} \left\{ \sum_{j \in E} c_j x_j^u \right\} = \min_{\boldsymbol{x} \in P} \left\{ \sum_{j \in E} c_j x_j \right\}. \tag{14}$$

Now, the LP on the r.h.s. of (14) can be shown to be solved by the performance $\boldsymbol{x} = \boldsymbol{x}^{\pi_G}$ of a *Gittins index priority policy* and hence by an argument similar to that used in our simple example, such a control must solve the stochastic optimisation problem. The policy π_G operates by giving each customer class k an index G_k and then implementing priorities among E according to these indices, with the maximal index class being accorded highest priority. The indices are obtained from the so-called adaptive greedy algorithm $AG(V, \boldsymbol{c})$ whose inputs are the matrix $V = \{V_j^S,\ j \in S,\ S \subseteq E\}$ and the cost vector $\boldsymbol{c}$. In this way, Gittins index policies can be shown to be optimal for discounted and undiscounted branching bandits. These single server models include many classical ones, including the discounted multi-armed bandit of Gittins (1979, 1989), the multi-class queue with Bernoulli feedback of Tcha and Pliska (1977) and Klimov networks (1974).

Recent contributions by the authors and co-workers have sought to develop these ideas in a number of directions, of which we shall mention just two, both of which are relevant to the later sections of this paper. Firstly, it has been demonstrated by Glazebrook and Garbe (1998) and Glazebrook and Niño-Mora (1997) that many systems of interest may come close to satisfying the key requirements in (10) - (12) above, but fail to do so exactly. In this event, Gittins index policies may reasonably be expected to perform well, if not optimally. In fact, the primal-dual structure of LP may be exploited to yield performance guarantees for such policies. In Section 3 this methodology is exploited to develop an analysis of a general multi-class queueing system serviced by M servers in parallel. In the single server case $M = 1$, (10) - (12) are satisfied exactly and Gittins index policies are optimal for a linear objective. When $M > 1$, we can develop measures of how close we come to achieving this (in Theorem 1) which in turn leads (in Corollary 2) to performance guarantees for such policies.

A second avenue of recent development has concerned work aimed at developing our understanding of how the optimal return Z^{OPT} depends upon the mix of customer classes requiring service. Garbe and Glazebrook (1998) elucidate system properties which yield laws of diminishing returns (increasing marginal costs) as more demands are placed upon the system. Such a notion may be expressed mathematically by the requirement that the optimal return is a supermodular function of

the set of customer classes which is allowed access to the service. This work is exploited in Section 4 to develop an approach to distributing the load across a network of interconnected stations, when the work offered at each station is itself to be scheduled optimally.

We finally pause to note that the achievable region approach has recently found application outside the scope of indexable systems. Bertsimas (1995) discusses polling systems, multi-class queueing networks and loss systems. Niño-Mora (1998) has begun a study of intensity control problems from this perspective. Some early conclusions are presented in Section 5.

3 A general multi-class queue on parallel servers

We consider here the optimal control of an M-server queueing system. In the single server case $M = 1$, we demonstrate that the system satisfies GCL (10) - (12) and in consequence Gittins index policies are optimal for a linear objective. The analysis of this case will serve to show the reader how GCL may be established in practice. Note that this index result is not new. See Bertsimas et al. (1995) for an account. What is new here is our analysis of the notoriously difficult parallel server problem with $M > 1$. Here we do not have exact GCL but we come close. As a consequence, Gittins index policies come close to optimality. Following the work of Glazebrook and Garbe (1998), the achievable region approach furnishes us with performance guarantees for index policies, from which their asymptotic optimality in a heavy traffic limit may be inferred.

M servers are available to process the requirements of customers from classes in $E = \{1, 2, \ldots, N\}$. An assignment of available customers to servers is made at each integer time point. Should a class i customer be assigned to server m at time t (which occurrence is registered by assigning the indicator function $I_i^m(t)$ the value 1; it is 0 otherwise) then at time $t+1$ the class i customer disappears to be replaced by $\boldsymbol{n}_i^{tm} \equiv \{n_{11}^{tm}, n_{12}^{tm}, \ldots, n_{iN}^{tm}\}$ customers of classes $1, 2, \ldots, N$ respectively. For a given $i \in E$, the vectors $\boldsymbol{n}_i^{tm}$ are i.i.d. as (t, m) varies and for simplicity t (and sometimes also m) will be dropped from the notation when no confusion arises. As we shall see, this modelling approach enables us to incorporate state transitions for existing customers as well as new arrivals into the system. To complete the system description, note that an idle server is deemed to be serving a class 0 customer and we suppose that there are always M such customers present in the system, one for each server. This additional class is needed to ensure that the model allows new arrivals to enter an empty system. We extend the notation $\boldsymbol{n}_i^{tm}$ to include the case $i = 0$, but note that $n_{00} = 1$ and $n_{i0} = 0$ for all $i \neq 0$.

If $N_i(t)$ denotes the number of class i customers present in the system at decision epoch t, then

the evolution of the system between t and $t+1$ is described by

$$N_i(t+1) = N_i(t) + \sum_{m=1}^{M} \sum_{j=0}^{N} I_j^m(t) \left(n_{ji}^m - \delta_{ij} \right), \quad i \in E \tag{15}$$

$$N_0(t+1) = N_0(t) = M.$$

In (15), δ_{ij} is the Kronecker delta. The set of admissible controls $\mathcal{U}$ available are (i) non-anticipative, (ii) non-idling (which here means that E has priority over 0) and (iii) server-symmetric (scheduling systems do not use server label information). Please note that this third requirement is not strictly needed. It has been included as a vehicle for simplifying the discussion at certain key points. We can guarantee the *stability* of this system under all $u \in \mathcal{U}$ (the time-average number of customers in the system is finite) by requiring that the $N \times N$ matrix $\boldsymbol{I} - \boldsymbol{n}$ be positive definite. Here $\boldsymbol{I}$ is the identity and $\boldsymbol{n}$ has (i,j)-th entry equal to $E(n_{ij})$. See Bertsimas and Niño-Mora (1996). We shall assume that admissible controls result in a discrete time stochastic process $\{N(t)\}_{t=-\infty}^{\infty}$ with unique stationary distribution, all of whose second moments are finite. Write

$$\rho_i^u = E_u \left\{ I_i^m(t) \right\}, \quad i \in E \cup \{0\} \tag{16}$$

for the probability that control u assigns server m to a class i customer at decision epoch t, where the expectation in (16) is with respect to the stationary distribution. That this expectation does not depend upon t (by stationarity) and m (by server symmetry) is clear. However it turns out that it is also independent of the control u. To see this, apply E_u to both sides of (15) and use

$$E_u \left\{ N_i(t+1) \right\} = E_u \left\{ N_i(t) \right\}, \quad i \in E$$

to infer that $\boldsymbol{\rho}^u$ satisfies the system of equations

$$\sum_{i=0}^{M} \rho_i^u E(n_{ij}) = \rho_j^u, \quad j \in E; \qquad \sum_{i=0}^{M} \rho_i^u = 1. \tag{17}$$

This has a unique solution when $\boldsymbol{I} - \boldsymbol{n}$ is non-singular. We shall write ρ without the superscript in what follows. One particular focus of the analysis will concern the quality of the control policies in *heavy traffic*. In discussing a sequence of systems, we are said to approach the heavy traffic limit if the value ρ_0 (the steady state probability that a server is idle) approaches 0.

Example

Consider a discrete time version of a multi-class M/G/*parallel* queueing system with M servers and customer feedback as follows: customers belonging to one of L classes arrive for service according to independent Poisson streams with λ_l the rate for class l, $1 \leq l \leq L$. Service times T_l for class l customers are *i.i.d.* discrete random variables whose distribution has finite support $\{1, 2, \ldots, R_l\}$.

F_l denotes the distribution function of T_l. Once a class l customer has completed service, (s)he is fed back to the system as a class k customer with probability p_{lk}, or leaves the system, with probability $p_{l0} = 1 - \sum_{k=1}^{L} p_{lk}$. The scheduling regime gives to each customer chosen for service a single unit of processing before the position is reviewed again.

It is straightforward to cast this example into the general framework above. We require classes labelled $\{(l,r),\ 0 \le r \le R_l - 1,\ 1 \le l \le L\}$ with (l,r) representing those class l customers present in the system who have already received r units of processing. A newly arrived class l customer (either from outside or via feedback) belongs to $(l,0)$. When a unit of processing is allocated to a class (l,r) customer, there are two possibilities: either there is a failure to complete service and the customer is now in class $(l,r+1)$ or service is completed and the customer leaves the system or feeds back as a $(k,0)$ customer for some k. This, together with consideration of external arrivals yields the following choices of the components of the matrix $\boldsymbol{n}$:

$$\begin{aligned} E\{n_{(l,r),(l,r+1)}\} &= \frac{1 - F_l(r+1)}{1 - F_l(r)}, \quad 0 \le r \le R_l - 1, \quad 1 \le l \le L, \\ E\{n_{(l,r),(k,0)}\} &= \frac{\lambda_k}{M} + \frac{\{F_l(r+1) - F_l(r)\}p_{lk}}{1 - F_l(r)}, \\ &\qquad 0 \le r \le R_l - 1, \quad 1 \le l \le L, \quad 1 \le k \le L. \\ E\{n_{0,(k,0)}\} &= \frac{\lambda_k}{M}, \quad 1 \le k \le L. \end{aligned} \tag{18}$$

Now introduce the system parameters α_l, $1 \le l \le L$, obtained as the solution to the linear system

$$\alpha_l = \lambda_l + \sum_{k=1}^{L} \lambda_k p_{kl}, \quad 1 \le l \le L. \tag{19}$$

The quantity α_l is easily seen to be the *total arrival rate* for class l customers, aggregating the external arrival rate (λ_l) with an internal rate (second term on r.h.s. of (19)) obtained via feedback. Substituting from (18) and (19) into the equations (17) we obtain solutions in the form

$$\rho_{(l,r)} = \alpha_l \frac{\{1 - F_l(r)\}}{M}, \quad 0 \le r \le R_l - 1, \quad 1 \le l \le L.$$

Hence we deduce that

$$\rho_0 = 1 - \sum_{l=1}^{L} \sum_{r=0}^{R_l - 1} \rho_{(l,r)} = 1 - \sum_{l=1}^{L} \alpha_l E(T_l)/M \to 0 \tag{20}$$

in the heavy traffic limit. Note that $\sum_l \alpha_l E(T_l)$ measures the rate at which work is created. Hence (20) asserts that the heavy traffic limit is attained as this approaches M, the total service rate available.

Returning to our general system, our stochastic optimisation problem is expressed as

$$Z^{OPT} = \inf_{u \in \mathcal{U}} \left\{ \sum_{i \in E} c_i x_i^u \right\} \tag{21}$$

where $c_i > 0$, $i \in E$, and

$$x_i^u = E_u\{N_i(t)\}, \quad i \in E, \tag{22}$$

the expectation being taken under the stationary distribution. We would like to generate linear (in)equalities of the form (10) and (11). To this end, we deploy the potential function approach of Bertsimas et al. (1995) who consider, for each subset S of E, a quantity $\{R^S(t)\}^2$, where

$$R^S(t) = \sum_{i \in S} V_i^S N_i(t), \tag{23}$$

for suitably chosen V_i^S, $i \in S$. Note from (22), that $E_u\{R^S(t)\}$ is precisely the quantity on the l.h.s. of (11). We choose the positive constants V_i^S, $i \in S$, as solutions of the linear system

$$V_i^S = 1 + \sum_{j \in S} E(n_{ij}) V_j^S, \quad i \in S. \tag{24}$$

Observe that V_i^S may be thought of as the mean amount of S-work required by a class i customer - i.e., beginning from a situation in which only a single class i customer is present, this is the mean amount of processing required until there are no S-customers present. Hence $R^S(t)$ is the total amount of S-work in the system at t. Using (15) and (23) we infer that

$$R^S(t+1) = R^S(t) + \sum_{i \in S} \sum_{m=1}^{M} \sum_{j=0}^{N} V_i^S I_j^m(t)(n_{ji}^m - \delta_{ij}) \tag{25}$$

If we square both sides of (25), take E_u and enforce the stationarity condition

$$E_u\left[\{R^S(t+1)\}^2\right] = E_u\left[\{R^S(t)\}^2\right]$$

we infer the condition

$$\begin{aligned} 2E_u &\left\{ R^S(t) \sum_{m=1}^{M} \sum_{j=0}^{N} I_j^m(t) \sum_{i \in S} V_i^S (n_{ji}^m - \delta_{ij}) \right\} \\ &+ E_u \left[\left\{ \sum_{m=1}^{M} \sum_{j=0}^{N} I_j^m(t) \sum_{i \in S} V_i^S (n_{ji}^m - \delta_{ij}) \right\}^2 \right] = 0 \end{aligned} \tag{26}$$

Considerable simplification of (26) is possible which exploits the server-symmetry of u and the fact that $I_j^m(t) I_k^m(t) = 0$ whenever $j \neq k$. Straightforward algebra yields the following:

$$\sum_{i \in S} V_i^S x_i^u \quad = \quad E_u\{R^S(t)\}$$

$$
\begin{aligned}
&= E_u\left\{R^S(t)\sum_{i\notin S}V_i^S I_i^1(t)\right\} + b(S) \\
&+ \frac{1}{2}(M-1)E_u\left[\left\{\sum_{i\notin S}V_i^S I_i^1(t)\right\}\left\{\sum_{j\notin S}V_j^S I_j^2(t)\right\}\right], \qquad (27)
\end{aligned}
$$

where $b(S)$ is a control-invariant constant given by

$$
b(S) = \frac{1}{2}\sum_{j=0}^{N}\rho_j E\left[\left\{\sum_{i\in S}V_i^S(n_{ji}-\delta_{ij})\right\}^2\right] + \frac{1}{2}(M-1)\left\{1-2\left(\sum_{i\notin S}\rho_i V_i^S\right)\right\}. \qquad (28)
$$

Note that in both (27) and (28) the constants V_i^S, $i \notin S$ are obtained from a suitable extension of (24). It is from equation (27) that we are able to develop suitable forms of the (in)equalities (10) and (11) for this system. The requirement in (12) which we need for a full GCL/Gittins index analysis as described in Section 2 is satisfied in the single server case $M = 1$. When $M > 1$ we come close to having (12) in a sense which is made precise in the following result. Before stating it, please note that (10) may be regarded as a particular case of (12), namely for $S = E$. We shall also require the notation $b^+ = \max(b, 0)$ for set functions b.

Theorem 1 (Exact and approximate GCL for the system)

(i) For all values of M and all controls $u \in U$

$$
\sum_{i\in S}V_i^S x_i^u \geq b^+(S), \quad S \subseteq E; \qquad (29)
$$

(ii) M=1*: In the single server case $M = 1$ the system satisfies GCL, i.e. in addition to (i) we have*

$$
\sum_{i\in S}V_i^S x_i^\pi = b^+(S) = b(S) \quad \text{for } \pi : S \to S^c, \quad S \subseteq E;
$$

(iii) M $\geq$ 2*: When there is more than one server, controls π which give S priority over S^c come within a finite constant of achieving the bound on the right hand side of (29). In particular*

$$
\sum_{i\in S}V_i^S x_i^\pi \leq b^+(S) + \frac{1}{2}(M-1)(3+\hat{n})V^S, \quad S \subseteq E \qquad (30)
$$

where

$$
\hat{n} = \max_{i\in E}\sum_{j\in E}E(n_{ij}) \quad \text{and} \quad V^S = \max_{i\in S}V_i^S.
$$

Outline Proof

(i) The l.h.s. of (29) must be non-negative as must the first and third terms on the r.h.s. of (27). Part (i) is then an immediate consequence of (27).

(ii) If $M = 1$ and π gives S priority over S^c then, under π

$$
R^S(t) > 0 \Longrightarrow N_i(t) > 0 \quad \text{for some } i \in S \Longrightarrow I_i^1(t) = 0 \quad i \notin S.
$$

Hence the first term on the r.h.s. of (27) is zero. Since the third term is zero trivially, the result follows.

(iii) In the case $M \geq 2$ we are required to bound the first and third terms in (27) from above for policies π which give S priority over S^c. Taking the first term, we can assert that, since under π

$$\sum_{i \in S} N_i(t) \geq M \Longrightarrow I_i^1(t) = 0, \quad i \notin S$$

it follows that

$$\begin{aligned} E_\pi \left\{ R^S(t) \sum_{j \notin S} V_j^S I_j^1(t) \right\} &\leq (M-1)V^S \sum_{j \notin S} V_j^S E\{I_j^1(t)\} \\ &= (M-1)V^S \sum_{j \notin S} \rho_j V_j^S = (M-1)V^S \end{aligned} \tag{31}$$

In (31), note that $\sum_{j \notin S} \rho_j V_j^S = 1$ may be established *either* algebraically (from (17) and (24)) *or* by use of probabilistic arguments. We now consider the third term in (27). A Cauchy-Schwarz inequality yields

$$\begin{aligned} E_\pi \left[\left\{ \sum_{i \in S} V_i^S I_i^1(t) \right\} \left\{ \sum_{i \notin S} V_j^S I_j^2(t) \right\} \right] &\leq E_\pi \left[\left\{ \sum_{i \notin S} V_i^S I_i^1(t) \right\}^2 \right] \\ &= \sum_{j \notin S} \rho_j (V_j^S)^2 \leq (1+\hat{n}) V^S \end{aligned} \tag{32}$$

The last inequality in (32) follows simply from (24). The result is now a straightforward consequence of (27), (31) and (32). □

We see from Theorem 1 (i),(ii) and the material in Section 2 that in the single server case $M = 1$, the requirements described in (10)-(12) are met (i.e. GCL are satisfied) and the stochastic optimisation problem (21) is solved by a Gittins index policy. The indices concerned are derived from the adaptive greedy algorithm AG$(V, \boldsymbol{c})$.

In the parallel server case with $M \geq 2$ we proceed as follows: from Theorem 1 (iii), the set function Φ given by

$$\Phi(S) = \frac{1}{2}(M-1)(3+\hat{n})V^S, \quad S \subseteq E, \tag{33}$$

is a natural measure of how close we come to satisfying the GCL requirement in (12). Glazebrook and Garbe (1998) utilise the primal-dual structure of LP to develop a performance guarantee for the Gittins index policy derived from AG$(V, \boldsymbol{c})$ in terms of the measure Φ. Numerical and analytical evidence to date suggests that the tightest such guarantees available perform very well in bounding the level of suboptimality of the index policy π_G. We shall give a somewhat simplified account here which will be sufficient for our purposes. Note that the bounds we shall describe are by no means the tightest available from the methodology.

Application of AG$(V, \boldsymbol{c})$ yields the indices G_i, $i \in E$. The customer classes are then renumbered such that $G_N \geq G_{N-1} \geq \ldots \geq G_1$. Hence, the index policy π_G implements priorities among the customer classes in decreasing numerical order. We identify $S(j) = \{j, j+1, \ldots, N\}$ as the set of cardinality $N - j + 1$ of classes with highest index. Note that π_G prefers $S(j)$ to $\{S(j)\}^c$ for all j. Our goal here is to develop a bound for $Z^{\pi_G} - Z^{OPT}$ where Z^{π_G} is the expected cost associated with the Gittins index policy. From the theory cited above we deduce that

$$Z^{\pi_G} - Z^{OPT} \leq \sum_{j=1}^{N} \Phi\{S(j)\}(G_j - G_{j-1}) \tag{34}$$

where Φ is the above measure and G_0 is taken to be zero.

It is not difficult now to establish Corollary 2 by substituting from (33) into (34) and utilising the form of the algorithm AG$(V, \boldsymbol{c})$ which produces the indices.

Corollary 2 (Performance guarantee for Gittins index policy when $M \geq 2$)

$$Z^{\pi_G} - Z^{OPT} \leq \tfrac{1}{2} N(M-1)(3+\hat{n})\Big(\max_{i \in E} c_i\Big)$$

One remarkable thing about the claim in Corollary 2 that the index policy comes within a constant of optimality is that the optimum cost Z^{OPT} becomes infinite (under reasonable conditions) as the heavy traffic limit is approached. Hence π_G is asymptotically optimal in a sense made precise below. Such a result is not unexpected. Index policies are optimal in the single server case since they always make choices which drive down the rate at which costs are incurred as rapidly as possible. The parallel server case is complicated by the issue of how effectively controls utilise the full service capacity. (Attempts to tackle these issues directly have met with little success. See Weiss (1992,1995) for an authoritative discussion in the context of much simpler models than those cited here.) However, and to hugely oversimplify the issues concerned, in the heavy traffic limit server utilisation disappears as an issue and the system looks increasingly like one serviced by a single server working at M times the speed.

In order to establish the asymptotic optimality of π_G we can infer from inequality (29) with $S = E$ that

$$Z^{OPT} = \sum_{i \in E} c_i x_i^{OPT} \geq \min_{j \in E} \left(c_j / V_j^E\right) \sum_{i \in E} V_i^E x_i^{OPT} \geq b(E) \min_{j \in E} \left(c_j / V_j^E\right) \tag{35}$$

with the set function b given by (28). It will be enough to elucidate conditions which guarantee that the r.h.s. of (35) diverges to infinity in the heavy traffic limit $\rho_0 \to 0$. One way of achieving this is as follows: suppose that the vectors $\boldsymbol{n}_i$ record two types of changes to the composition of the system, namely

(1) external arrivals into customer classes within some designated subset $A \subseteq E$; and

(2) internal transfers via feedback or some other transition mechanism.

Plainly, our example above of an M/G/*parallel* system with feedback may be thought of in these terms. Hence when $i \in E \cup \{0\}$ we write

$$n_{ij} = \begin{cases} A_j + \tilde{n}_{ij}, & j \in A, \\ \tilde{n}_{ij}, & \text{otherwise.} \end{cases} \tag{36}$$

In (36), A_j denotes external arrivals to j (assumed independent of all other A_i and all of the $\tilde{n}_{kl}$) and $\tilde{n}_{ij}$ internal transfers from i to j. We assume that $E(A_j) = \lambda_j/M$, where λ_j is an overall class j arrival rate for the system. We shall approach the heavy traffic limit by increasing the λ_j appropriately while (a) keeping the $E(\tilde{n}_{ij})$ fixed and (b) keeping the var(A_j) bounded away from zero. Note that (b) is required to avoid certain pathologies which occur in deterministic cases. Note also that all this is quite natural in the M/G/*parallel* case.

Utilising (36) within an expanded version of (24) which includes the "idleness" class 0, we can solve for $\boldsymbol{V}^E = [V_j^E,\ j \in E \cup \{0\}]$, obtaining

$$\boldsymbol{V}^E = \hat{V}(\boldsymbol{I} - \tilde{\boldsymbol{n}})^{-1}\boldsymbol{e}, \tag{37}$$

where in (37)

$$\hat{V} = 1 + \left(\sum_{j \in A} \lambda_j V_j^E / M \right),$$

$\boldsymbol{e}$ is a vector with all entries equal to one and $\tilde{\boldsymbol{n}}$ is a matrix whose (i,j)-th entry is $E(\tilde{n}_{ij})$. Note that $\boldsymbol{I} - \tilde{\boldsymbol{n}}$ is guaranteed non-singular by earlier assumptions.

Recall from the proof of Theorem 1 the identity $\sum_{j \notin S} \rho_j V_j^S = 1$. In the case $S = E$ this yields $\rho_0 V_0^E = 1$. Hence in the heavy traffic limit $\rho_0 \to 0$ and $V_0^E \to \infty$. However in (37), since we have assumed that $\boldsymbol{I} - \tilde{\boldsymbol{n}}$ remains fixed as we take the limit, it must follow that $\hat{V} \to \infty$ and hence that $V_j^E \to \infty$, $j \in E$. We can now assert the asymptotic optimality of the Gittins index policy π_G.

Theorem 3 (Heavy traffic optimality of Gittins index policy when $M \geq 2$)
In the above heavy traffic limit

$$\frac{Z^{\pi_G} - Z^{OPT}}{Z^{OPT}} \to 0$$

Proof

We utilise (28) to obtain $b(E)$. By standard results and the fact that $\rho_0 V_0^E = 1$, we deduce that

$$\begin{aligned} 2b(E) &\geq \sum_{j=0}^{N} \rho_j \text{var} \left(\sum_{i \in E} V_i^E n_{ji} \right) - (M-1) \\ &\geq \sum_{j=0}^{N} \rho_j \text{var} \left(\sum_{i \in A} V_i^E A_i \right) - (M-1) \qquad (38) \\ &= \sum_{i \in A} (V_i^E)^2 \text{var}(A_i) - (M-1). \qquad (39) \end{aligned}$$

To obtain (38), we use (36) and the independence assumptions following. From (35), (37) and (39) we conclude that

$$Z^{OPT} \geq O(\hat{V}) \to \infty$$

in the heavy traffic limit. The result now follows from Corollary 2. □

4 Load balancing in distributed systems

A common architecture for multiprocessor systems is a distributed one consisting of a network of (relatively) autonomous servers. The issue of the efficient allocation of resources in such contexts is both important and complex. See Gelenbe and Pekergin (1993). One fundamental question concerns the distribution of work across the network or, as we shall call it, load balancing. The theoretical literature has, in the main, concentrated on very simple models. For these, simple round robin policies and Bernoulli routing with equal probabilities have frequently been proposed as optimal load balancing regimes when little information is available to the controller. See, for example, Liu and Towsley (1994). When full information on queue lengths is available, join the shortest queue has been shown to be optimal for a variety of models. See Weber (1978).

In a contribution which represented a significant advance, Ross and Yao (1991) were able to show that considerable savings could be made if optimal scheduling of the work offered at each station of the network could be incorporated into the load balancing problem. Their work made use of the achievable region approach, but predated many of the most significant advances outlined in Section 2. The authors of the current paper and co-workers plan a much more extensive study and we report here some of the early findings.

We shall consider a communication network interconnecting multiple stations, with two types of jobs generated at each station: those which are dedicated (D) to that station and must be processed there, and those which are generic (G) and could be processed anywhere in the network. There may be several classes of D and G jobs, arriving in independent Poisson streams. We seek to split the G traffic among the individual stations in an optimal fashion given that each station schedules its offered work (both D and G) optimally. On the basis of a realistic appraisal of the communication/processing overhead generated thereby our policies for scheduling at each station will be dynamic (i.e. decisions will be made on the basis of the evolving state of each station) while the load balancing component of the problem will be static (i.e. the vector of generic arrival rates will be split once for all between the stations). At this point we introduce two simple examples to assist the reader.

Examples

It may seem plausible to conjecture that when the stations in the network are identical (in all relevant respects) then an optimal load balancing regime will split the G jobs equally among them. The following simple examples will caution the reader against drawing such conclusions too easily. In both examples the network comprises two identical single-server stations. In each case there are two G job classes and no D jobs. The objective in both examples is the minimisation of $c_1E(N_1) + c_2E(N_2)$ where $E(N_i)$ is the expected number of class i jobs in the system and the expectation is taken under the steady-state distribution of the corresponding stochastic process with c_i a holding cost rate, $i = 1, 2$.

Example 1

Here we shall suppose that generic job class 1 has zero holding costs ($c_1 = 0$) but a high arrival rate to the network ($\lambda_1 = 0.9$), while for job class 2 we have positive holding costs ($c_2 = 1$) and a low arrival rate ($\lambda_2 = 0.1$). The processing time of all jobs is exactly 1 and at each station scheduling is non-preemptive. Plainly at both stations the optimal scheduling regime prefers class 2 to class 1 and must impose that priority in a non-preemptive fashion.

Since $c_1 = 0$, our objective is to split the load in order to minimise $E(N_2)$. An even split of arriving jobs between the stations will result (frequently) in situations where an arriving class 2 job finds the machine busy with a class 1 job and is thus delayed while its processing is completed. An alternative regime in which all class 1 jobs go to one machine and all class 2 jobs to the other will result in less frequent delays to the latter because λ_2 is small. Simple calculations show that the "one job class per machine" regime yields a 16.43% saving in expected cost over an even distribution of work.

Example 2

Plainly the non-preemptive nature of the scheduling regime plays a significant role in Example 1. Consider now a situation in which scheduling priorities are imposed preemptively. We shall suppose that all processing times are exponentially distributed with mean 1 for class 1 and mean 0.1 for class 2. The usual full range of independence assumptions are made. We also take $c_1 = 1, \lambda_1 = 0.5$ and $c_2 = 0.1, \lambda_2 = 10$. Direct calculations serve to show that a (near-optimal) splitting of the load in which all class 1 traffic is directed to station 1, while 85% of class 2 traffic goes to station 2 offers a 17.25% saving in expected cost over an even distribution of work. Please note that Example 2 elaborates Example 1 in that processing times for generic jobs are not identically distributed. Subsequent theory serves to show that this is a required feature for an even split solution to be suboptimal with exponential processing times and priorities imposed preemptively.

We shall suppose that our load balancing problem may be expressed as

$$\min_{\mathbf{\Lambda}} \sum_m Z_m^{OPT}(\boldsymbol{\lambda}_m) \quad \text{subject to} \quad \mathbf{\Lambda} \boldsymbol{e} = \boldsymbol{\lambda} \tag{40}$$

In (40), λ_{gm} is the offered load (i.e. arrival rate) of class g jobs at station m, where $g \in G$, $\boldsymbol{\lambda}_m$ is the vector of generic loads at station m and $\mathbf{\Lambda} = \{\lambda_{gm}\}$ is the generic load matrix. Vector $\boldsymbol{\lambda}$ summarises the total generic load for the network and $\boldsymbol{e}$ is an M-vector of 1's, where M is the number of stations. $Z_m^{OPT}(\boldsymbol{\lambda}_m)$ is the minimised cost at station m when $\boldsymbol{\lambda}_m$ is the generic load offered there. This minimised cost is achieved when the offered work is scheduled optimally.

Plainly, an ability to compute and/or characterise the returns Z_m^{OPT} as functions of the generic load vectors $\boldsymbol{\lambda}_m$ will contribute to achieving optimal or near optimal solutions to (40). As we shall now see, we can make considerable progress when each station satisfies GCL. We drop the station suffix m as we carry the discussion forward regarding the individual stations in the network.

Happily it is one of the features of the GCL/indexable systems described in Section 2 that computations of expected cost for a given (priority) policy can be performed with ease, as can the computation of minimised cost Z^{OPT}. See Bertsimas and Niño-Mora (1996). In addition, Z^{OPT} can often be characterised in a way which will ultimately assist with (40) as follows: consider a GCL system with linear costs and an associated universal set E of *potential* customer classes. For specified $S \subseteq E$, let $Z^{OPT}(S)$ be the minimised cost for the reduced system in which only customer classes within S are allowed access to service. Garbe and Glazebrook (1998) showed that the achievable region approach yields the conclusion that, subject to some additional structural requirements, Z^{OPT} is an *increasing, supermodular* function, namely

$$Z^{OPT}(S) \leq Z^{OPT}(T), \quad S \subseteq T, \quad \text{(increasing)}$$

$$\begin{aligned} Z^{OPT}(S \cup \{j\}) - Z^{OPT}(S) \quad &\leq \quad Z^{OPT}(T \cup \{j\}) - Z^{OPT}(T), \\ &\qquad S \subseteq T \quad \text{and} \quad j \notin T \quad \text{(supermodular)}. \end{aligned} \tag{41}$$

Supermodularity states, in this context, that allowing an additional class of customers access to a more congested system increases the optimum cost by more than allowing the same additional class access to a less congested system. This seems a natural property for Z^{OPT}.

We shall want to draw on this result in our discussion of load balancing. However, rather than develop the theory through general model structures which have the properties required to establish (41), for clarity we shall conduct the discussion in terms of a specific GCL model for each station which meets the requirements. Directions in which the material can be generalised are sketched at the end of the section.

Model for local scheduling at each station

We shall suppose that each station is a *Klimov network* (see Klimov (1974)) as follows (note that we continue to drop the station suffix m): customers who are members of classes within $D \cup G$ are assumed to arrive at the station in a set of independent Poisson streams. Use λ_g to denote an arrival rate for generic class $g \in G$ and $\boldsymbol{\lambda}$ the corresponding vector of generic arrival rates. All customers have exponential service times with mean denoted μ_g^{-1} for class $g \in G$. Upon completion of service, a class i customer may be routed to receive further service as a class j customer with probability p_{ij}, or it may leave the station with probability $p_{i0} = 1 - \sum_{j \in D \cup G} p_{ij}$. All the customer arrival processes, service times and routing events are mutually independent. The routing probability matrix $\boldsymbol{P} = (p_{ij},\ i, j \in D \cup G)$ is such that $\boldsymbol{I} - \boldsymbol{P}$ is invertible, thus guaranteeing that a customer entering the system will leave it with probability one. We also require that

$$p_{ij} = 0 \text{ for } i \in D,\ j \in G \text{ and } i \in G,\ j \in D \cup G$$

Hence D-customers can feed back as D-customers in a quite general way but G-customers are "simple" in that they have no feedback mechanism. In this way the dedicated traffic at a station can be quite general in its structure. For example, the framework proposed allows dedicated customers to have a state which evolves in continuous time as a (finite state) Markov process, through to completion. In contrast, the generic traffic is simple in structure, as would seem natural. Admissible scheduling controls at the station are non-anticipative, non-idling and preemptive.

Each customer class $i \in D \cup G$ has an associated holding cost rate c_i and so the minimised cost Z^{OPT} for the station is given by

$$Z^{OPT}(\boldsymbol{\lambda}) = \inf_{u \in \mathcal{U}} \left\{ \sum_{i \in D \cup G} c_i x_i^u \right\}$$

with $x_i^u = E_u(N_i)$, the long-term average number of class i customers at the station.

For the efficient solution of (40), we would ideally like each Z^{OPT} to be an increasing, convex function of the offered generic load $\boldsymbol{\lambda}$. However in higher dimensions, full convexity is a very strong property and in general we must settle for the weaker form described in Definition 1 in which convexity is available in certain directions (NE-SW) only in load space.

Definition 1

A function $f : (\Re^+)^n \to \Re$ *is* North-East (NE) convex *if, for all* $\alpha \in [0,1]$ *and all* $\boldsymbol{\lambda}'$, $\boldsymbol{\lambda}''$ *such that* $\boldsymbol{\lambda}' - \boldsymbol{\lambda}'' \in (\Re^+)^n \cup (\Re^-)^n$,

$$\alpha f(\boldsymbol{\lambda}') + (1-\alpha) f(\boldsymbol{\lambda}'') \geq f\{\alpha \boldsymbol{\lambda}' + (1-\alpha)\boldsymbol{\lambda}''\}$$

There is a simple proof of Theorem 4 which begins with the supermodularity property in (41) and infers from that properties of $Z^{OPT}(\boldsymbol{\lambda})$. The argument essentially secures increased generic arrival

rates through the introduction of new generic job classes with appropriate service characteristics. We omit the details.

Theorem 4

For our Klimov network model, the minimised cost $Z^{OPT} : (\Re^+)^{|G|} \to \Re^+$ *is increasing and NE convex.*

Please note that NE convexity certainly includes convexity in each co-ordinate direction (for fixed values elsewhere). Further, in the one-dimensional case it coincides with full convexity. Corollary 5 follows.

Corollary 5

If $|G| = 1$*, the minimised cost* $Z^{OPT} : \Re^+ \to \Re^+$ *is increasing, convex.*

Hence we have full convexity for the case of a single generic class. This can be readily extended in two directions: the first concerns situations in which the controller of the distributed system cannot distinguish between generic jobs. In this case, the solution to (40) will be of the form $\{\alpha_m,\ 1 \leq m \leq M\}$ where α_m is the proportion of all generic traffic passed to station m. Let $\boldsymbol{\lambda}$ now stand, as in (40), for the generic load for the network. The optimisation goal becomes the minimisation of

$$\sum_m Z_m^{OPT}(\alpha_m \boldsymbol{\lambda}) \text{ subject to } \alpha_m \geq 0,\ 1 \leq m \leq M,\ \sum_m \alpha_m = 1. \tag{42}$$

To solve (42), our interest is in $Z^{OPT}(\alpha\boldsymbol{\lambda})$ as a function of α for fixed $\boldsymbol{\lambda}$, where $\alpha \in [0,1]$. The following is an immediate consequence of Theorem 4.

Corollary 6

For fixed $\boldsymbol{\lambda}$*, the minimised cost* $Z^{OPT}(\cdot\boldsymbol{\lambda}) : \alpha \to Z^{OPT}(\alpha\boldsymbol{\lambda})$ *is increasing, convex.*

Another direction in which Theorem 4 can be extended is to cover those situations for which $|G| > 1$, but where all generic processing requirements are i.i.d. with $\mu_g = \mu$, $g \in G$.

Theorem 7

If $\mu_g = \mu$*,* $g \in G$*, the minimised cost* $Z^{OPT} : (\Re^+)^{|G|} \to \Re^+$ *is increasing, convex.*

Outline Proof

Write $N = |D \cup G|$. As in Section 3, the customer classes at our single station are renumbered such that $G_N \geq G_{N-1} \geq \ldots \geq G_1$ and again we write $S(j) = \{j, j+1, \ldots, N\}$. In the notation established in (10)-(14) in Section 2, it will assist to express the dependence of the base function b on the generic arrival rate $\boldsymbol{\lambda}$. Hence we write $b(S, \boldsymbol{\lambda})$, $S \subseteq E$. Recall that Z^{OPT} is the value of the

LP in (14) and hence also of its dual. The latter was shown by Bertsimas and Niño-Mora (1996) to be expressible as

$$Z^{OPT}(\boldsymbol{\lambda}) = \sum_{j=1}^{N} b\{S(j), \boldsymbol{\lambda}\}(G_j - G_{j-1}) \tag{43}$$

where in (43), $G_0 = 0$. Note also that it is straightforward to show for our Klimov network model that the indices G_j do not depend on $\boldsymbol{\lambda}$.

From the generalised conservation laws in (11) and (12) we may write

$$\inf_{u \in \mathcal{U}} \left\{ \sum_{i \in S(j)} V_i^{S(j)} x_i^u \right\} = b\{S(j), \boldsymbol{\lambda}\} \tag{44}$$

It can be shown that, since the generic classes in $G \cap S(j)$ have i.i.d. processing requirements then they must all have the same associated value of $V_g^{S(j)}$. Hence, they may be regarded corporately as a single customer class with arrival rate $\sum_{g \in G \cap S(j)} \lambda_g$ so far as the stochastic optimisation problem on the l.h.s. of (44) is concerned. It then follows from Corollary 5 that we may write $b\{S(j), \boldsymbol{\lambda}\} \equiv b_j \left(\sum_{g \in G \cap S(j)} \lambda_g \right)$ where $b_j : \Re^+ \to \Re^+$ is increasing, convex. Theorem 7 now follows from (43), the $\boldsymbol{\lambda}$-independence of the indices and from basic properties of convex functions. □

We have established a range of scenarios in which the minimised cost at each station is increasing, convex (Corollaries 5, 6 and Theorem 7, with more to come) and a greater range for which convexity is available for certain directions in load space (including co-ordinatewise). We now proceed to consider briefly the implications for the load balancing problem (40). We begin by consideration of the special case in which all stations are identical, i.e. the minimised cost for station m, $Z_m^{OPT}(\cdot) \equiv Z^{OPT}(\cdot)$, $1 \leq m \leq M$.

Theorem 8

When stations are identical, it is optimal to split the generic load evenly among stations for all loads $\boldsymbol{\lambda}$ *if and only if* $Z^{OPT}(\cdot)$ *is convex.*

Proof

If $Z^{OPT}(\cdot)$ is convex and $\boldsymbol{\lambda}_m$ is the generic load for station m as in (40), then

$$\sum_{m=1}^{M} Z^{OPT}(\boldsymbol{\lambda}_m) \geq M Z^{OPT} \left(\sum_{m=1}^{M} \frac{1}{M} \boldsymbol{\lambda}_m \right) = M Z^{OPT} \left(\frac{1}{M} \boldsymbol{\lambda} \right), \tag{45}$$

by convexity. However, the final term in (45) is plainly the cost corresponding to an even load distribution. For the converse, see Dacre and Glazebrook (1998). □

It is possible to supplement Theorem 8 via the development of performance guarantees for an even split of the generic load when full convexity for $Z^{OPT}(\cdot)$ is not available. For example, if we

take one of the simplest cases of interest, namely of two identical stations each having $|D| = 0$ and $|G| = 2$, it can be shown that an even load distribution yields a cost which is within a fraction

$$\frac{1}{2}\frac{|\mu_1 - \mu_2|}{(\mu_1 + \mu_2)} \tag{46}$$

of the optimal cost for the network. See Dacre and Glazebrook (1998). Note that the expression in (46) is 0 when $\mu_1 = \mu_2$, indicating that an even distribution is optimal in the i.i.d. case. This is in agreement with Theorems 7 and 8.

Please note that, following Theorem 4 and extensive numerical investigation, there are many systems for which the optimum cost, while not fully convex, comes close to being so. When the $Z_m^{OPT}(\cdot)$ are indeed all convex, an efficient iterative procedure is available for the load balancing problem in (40) which solves a sequence of LP's determined via sub-gradients of the objective. Our numerical study has shown that, in practice, this approach yields acceptable solutions even in the absence of full convexity. In Figures 2 and 3 below we illustrate the performance of (a) an even load distribution and (b) this LP-based heuristic for (40) for the simple case above, namely of two identical stations with $|D| = 0$ and $|G| = 2$. The figures are based on a grid of $(60)^2$ points with both $\log_e(c_2/c_1)$ and $\log_e(\mu_2/\mu_1)$ taken to be in the range $-3(0.1)3$. The values of μ_1 and c_1 are both set equal to 1, although the results presented are the same for any assigned values of these constants. At each grid point is presented a summary of the performance for the chosen load balancing regimes over 120 problems - each one corresponding to a choice of generic arrival rate $\boldsymbol{\lambda}$. In Figure 2, the chosen performance measure is the maximum percentage level of suboptimality of the load balancing regime over the 120 problems while in Figure 3 we report the percentage of solutions which were within 0.01% of optimality. In interpreting the results, note the following:

(i) By Theorem 8, we should expect the performance of the even load distribution to give an indication of the extent of non-convexity of Z^{OPT}.

(ii) Following (43), it is possible to express Z^{OPT} in the form

$$\{|c_1\mu_1 - c_2\mu_2| \times \text{convex}(\boldsymbol{\lambda})\} + \{(\min c_i\mu_i) \times \text{non-convex}(\boldsymbol{\lambda})\}$$

Hence we should expect the degree of non-convexity of Z^{OPT} to be related to the absolute size of $c_1\mu_1 - c_2\mu_2$ and to be at its most pronounced when $\log_e(c_2/c_1) \cong -\log_e(\mu_2/\mu_1)$;

(iii) By Theorem 7, $\log_e(\mu_2/\mu_1) = 0$ is a convex case for which the even load distribution will be optimal.

The results presented in Figures 2-3 are wholly consistent with (i)-(iii). The even load distribution heuristic is optimal when $\log_e(\mu_2/\mu_1) = 0$ and has its weakest performance around the

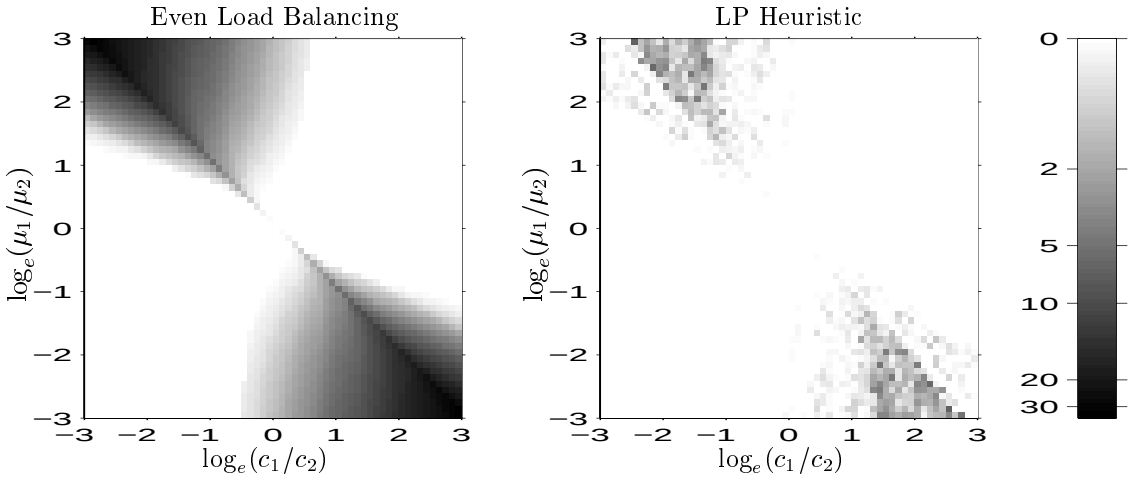

Figure 2: Maximum Suboptimality (Percentage of Optimum)

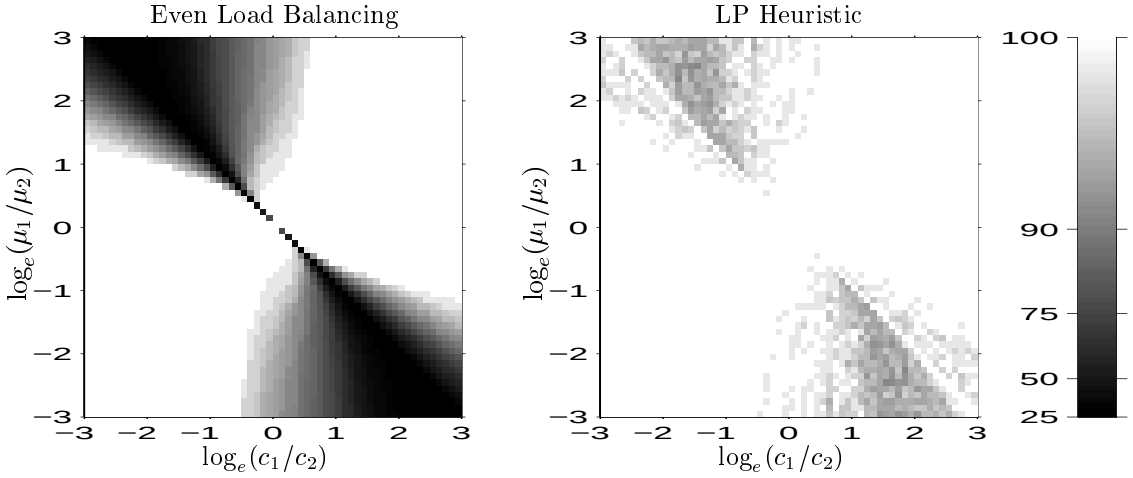

Figure 3: Percentage of Solutions Optimal

line $\log_e(c_2/c_1) = -\log_e(\mu_2/\mu_1)$. The LP-based heuristic offers a significant improvement when non-convexity is a serious issue and achieves a high level of performance almost uniformly.

In addition to the special role of even load distributions for identical stations, simple algorithms for load balancing are available when $|G| = 1$ and the $Z_m^{OPT}(\cdot)$ are fully convex, but where stations are not identical. The latter raises many important modelling possibilities, including those in which the dedicated traffic has a different stochastic character at different stations and also where the processing time distributions of generic jobs are station-dependent. The algorithms concerned are all based on procedures which match gradients and are variants of those proposed by Tantawi and Towsley (1984,1985). We omit the details.

We conclude by supposing that we have such an algorithm for a network in which Corollary 5 applies at each station - namely, there is a single generic class and the minimised cost is increasing convex. We shall show how to develop from that a load balancing algorithm for the more general situation in Theorem 7 in which generic job classes are distinguished only by their holding cost rates. Suppose, then, that we have such an algorithm for the situation in which $|G| = 1$ and, further, that at each station D-jobs always have preemptive priority over G-jobs. Processing requirements for G-jobs may vary from station to station but holding costs do not. With this set-up, balancing the generic load can have no impact upon the total costs incurred by dedicated jobs across the network. We write $\nu_m(\lambda)$ for the optimal generic load at station m when λ is the total generic load for the network. See Dacre and Glazebrook (1998) for a proof of the following:

Lemma 9

There is a solution to the above simple load balancing problem for which $\nu_m : \mathcal{R}^+ \to \mathcal{R}^+$ is increasing for each m.

We now move on to consider a more general model, as above, but where the $|G|$ job classes have processing requirements which are i.i.d. at each station and have holding cost rates $c_i, i \in G$, which apply across the network. It is straightforward to establish that the optimal scheduling of generic jobs at each station is according to priorities determined by the c_g, $g \in G$, with the largest c_g having the highest priority. Renumber the generic classes such that

$$c_{|G|} \geq c_{|G|-1} \geq \ldots \geq c_1$$

Recall that the total generic loads for the network are λ_g, $g \in G$.

Theorem 10

There is a solution to the above load balancing problem for which the optimal class g load at station m is $\nu_m\left(\sum_{j=g}^{|G|} \lambda_j\right) - \nu_m\left(\sum_{j=g+1}^{|G|} \lambda_j\right)$ for all g, m where ν_m is as in Lemma 9.

Proof

Let $\pi_{mg}(\boldsymbol{\lambda})$ be the class g load allocated to station m by a general solution π to our load balancing problem. If we write $Z(\pi, \boldsymbol{\lambda})$ for the total network cost for the generic jobs under this solution then it is not difficult to see that we have the decomposition

$$Z(\pi, \boldsymbol{\lambda}) = \sum_{g=1}^{|G|} Z_g(\pi, \boldsymbol{\lambda}). \tag{47}$$

In (47), $Z_g(\pi, \boldsymbol{\lambda})$ is the generic cost associated with an equivalent $|G| = 1$ network in which the station m load is $\sum_{j=g}^{|G|} \pi_{mj}(\boldsymbol{\lambda})$ and the common holding cost rate is $c_g - c_{g-1}$. We take $c_0 = 0$. But

$$\sum_{m=1}^{M} \sum_{j=g}^{|G|} \pi_{mj}(\boldsymbol{\lambda}) = \sum_{j=g}^{|G|} \lambda_j$$

is the total load for this network, and so by Lemma 9 it is optimised by allocating generic load $\nu_m(\sum_{j=g}^{|G|}\lambda_j)$ to each station m. However,

$$\nu_m\left(\sum_{j=g}^{|G|}\lambda_j\right)=\sum_{j=g}^{|G|}\nu_{mj}(\boldsymbol{\lambda})\quad\text{for all } g,m$$

where $\nu\equiv\{\nu_{mj}\}$ is the load balancing solution proposed in the theorem. Note that Lemma 9 guarantees the admissibility of ν. From the above we conclude that

$$Z_g(\pi,\boldsymbol{\lambda})\geq Z_g(\nu,\boldsymbol{\lambda})\text{ for all } g$$

and so, from (47),

$$Z(\pi,\boldsymbol{\lambda})\geq Z(\nu,\boldsymbol{\lambda}),$$

as required. □

Extensions

(1) The general conditions which guarantee that an indexable system has an increasing supermodular value function Z^{OPT} is that it be reducible and decomposable. See Garbe and Glazebrook (1998) for more details and examples of systems which meet these requirements.

(2) The above discussion via the Klimov network model supposes that the D-customers and the G-customers at a station are dealt with on the same basis through a linear objective involving all job classes. Hence, prioritising between these two customer types (and the natural proposal is to give dedicated customers a higher priority) is via appropriate choice of the c_i, $i\in D\cup G$. Another obvious approach is to *impose* the requirement that D-customers must always be given priority over G-customers as is done in the concluding discussion leading to Theorem 10. The latter then have the status of "background" jobs which are allowed access to service capacity which is surplus to the primary goal of serving the D-customers. This latter proposal can easily be accommodated through Garbe and Glazebrook's (1998a) achievable region account of stochastic scheduling with imposed priorities.

(3) Another way of asserting the primacy of the D-customers at each station is to impose delay constraints of the form $x_d^u\leq t_d$, $d\in D$. Among the controls which meet the delay constraints the goal would be to choose one to minimise $\sum_{g\in G}c_g x_g^u$. This is the approach of Ross and Yao (1991) who take as their station model a multi-class M/G/1 queue with priorities imposed non-preemptively. In this case we have convexity of the optimal returns for the case of a single generic class.

5 Threshold policies for intensity control

A fundamental question which can be asked of any queueing system concerns how much work we need to hold in the system in order to achieve a given level of throughput (i.e., the rate of job flow through the system). The intuition is that letting the work in process (WIP) grow beyond a certain level will do little to increase throughput. However, achieving a given throughput can only be done at the expense of a large enough WIP. A class of policies used frequently in practice is the class of *threshold policies* which control the system by setting a WIP cap: when the work in process reaches this cap the arrivals process is shut off. The following basic questions arise: what is the minimum WIP level required to attain a target throughput level? When are threshold policies optimal for maximising a linear throughput-WIP objective?

Such questions have conventionally been explored by dynamic programming methods. Niño-Mora (1998) is developing a unifying achievable region approach to such issues and this section contains an introduction to the key ideas based on an application to a queueing intensity control model due to Chen and Yao (1990). We give an indication of how the ideas generalise at the end of the section.

The model is a queueing system which consists of a facility servicing a single customer class. $N(t)$ denotes the number of customers in the system at time $t \geq 0$. We control the process $\{N(t),\ t \geq 0\}$ by means of a policy which sets the current *stochastic intensities* (or rates) $\lambda(t)$ and $\mu(t)$ of the arrival and departure process, respectively. The sequences $\{\bar{\lambda}_k,\ k = 0, 1, \ldots\}$ and $\{\bar{\mu}_k,\ k = 0, 1, \ldots\}$ of input and output capacity limits impose bounds on the arrival/departure intensities when k customers are in the system. A policy will be *admissible* if it is non-anticipative (i.e., it is adapted to the system's history), stable (i.e., the process $\{N(t),\ t \geq 0\}$ is ergodic) and satisfies the input and output capacity constraints, expressed as

$$N(t) = k \Longrightarrow \lambda(t) \leq \bar{\lambda}_k,\ \mu(t) \leq \bar{\mu}_k, \quad t \geq 0,\ k = 0, 1, 2, \ldots$$

We denote by $\mathcal{U}$ the class of admissible policies. Of special interest is the class of *threshold policies*: for each integer $b \geq 0$, the *b-threshold policy* sets the input intensity at full capacity if $N(t) < b$, and to 0 otherwise. Output intensity is always set at full capacity.

The achievable region approach requires us to develop a notion of performance, which here must measure both throughput (μ^u for policy u) and WIP (N^u). We consider a time-average criterion and define

$$\mu^u = \lim_{T \to \infty} \frac{1}{T} \int_0^T E_u\{\mu(t)\}\ dt \tag{48}$$

and

$$N^u = \lim_{T\to\infty} \frac{1}{T} \int_0^T E_u\{N(t)\}\, dt \tag{49}$$

We focus primarily on the following economic structure: a unit reward is received at each service completion time. In addition, each customer in the system (whether waiting or in service) incurs holding costs at a rate $c > 0$ per unit time. Our goal is to choose an admissible control to maximise the long-term net rate of return, i.e.

$$Z^{OPT}(c) = \sup_{u\in\mathcal{U}} Z^u(c) = \sup_{u\in\mathcal{U}}\{\mu^u - cN^u\} \tag{50}$$

In order to make progress we need to make the following plausible assumptions about the input and output capacity limits. In Assumption 1, the terms increasing and decreasing are used in the non-strict sense.

Assumption 1

(a) The sequence $\{\bar{\lambda}_k,\ k \geq 0\}$ of input capacity limits is decreasing;

(b) The sequence $\{\bar{\mu}_k,\ k \geq 0\}$ of output capacity limits is increasing concave, with $\bar{\mu}_{k+1} - \bar{\mu}_k \to 0$, $k \to \infty$.

Consider now this system evolving under a b-threshold policy, defined above. We denote the associated performance measures μ^b and N^b, and the corresponding objective $Z^b(c) = \mu^b - cN^b$, $b \geq 0$. We also write c^b for the *critical cost parameter*, given by

$$c^b = (\mu^b - \mu^{b-1})/(N^b - N^{b-1}), \quad b \geq 1, \tag{51}$$

with $c^0 = 0$. Under a b-threshold policy the system evolves as a birth-death process on states $0, \ldots, b$ with state-dependent birth intensities $\bar{\lambda}_i$, $0 \leq i \leq b-1$, (and 0 otherwise) and death intensities $\bar{\mu}_i$, $1 \leq i \leq b$. The stationary distribution of this process is well known to be given by

$$\pi_i^b = K_b \prod_{j=0}^{i-1} (\bar{\lambda}_j/\bar{\mu}_{j+1}), \quad 0 \leq i \leq b, \tag{52}$$

where an empty product is unity and K_b is the required normalising constant. We have

$$\mu^b = \sum_{i=1}^{b} \bar{\mu}_i \pi_i^b \quad \text{and} \quad N^b = \sum_{i=1}^{b} i\pi_i^b, \quad b \geq 1. \tag{53}$$

An expression for the critical cost parameter c^b is easily recovered from (52) and (53).

It is straightforward to demonstrate that the following properties of the quantities introduced above flow from Assumption 1. See Niño-Mora (1998) for details.

Lemma 11

(i) The sequences $\{\mu^b,\ b \geq 0\}$ and $\{N^b,\ b \geq 0\}$ are both (strictly) increasing;

(ii) The sequence $\{c^b,\ b \geq 1\}$ *is positive and (strictly) decreasing with limit zero;*
(iii) $\mu^u - c^b N^u \leq Z^b(c^b), \quad u \in \mathcal{U}, \quad b \geq 1.$

Note that Lemma 11(iii) is an assertion of the optimality of the b-threshold policy for the critical cost parameter c^b, $b \geq 1$. The achievable region analysis of the stochastic optimisation problem in (50) for any $c > 0$ now flows naturally. Many of the issues raised in the introductory paragraph to this section are resolved as a by-product of the analysis.

We introduce the *performance space*

$$X = \{(\mu^u, N^u),\ u \in \mathcal{U}\}$$

Following Lemma 11, a natural candidate for X is the *threshold polygon* P given by

$$P = \{\boldsymbol{x} \in (\Re^+)^2; \quad x_1 - c^b x_2 \leq Z^b(c^b), \quad \text{for} \quad b \geq 1\} \tag{54}$$

which is depicted in Figure 4. It is easy to show that the extreme points on the lower boundary of P are (μ^b, N^b), $b \geq 1$, namely, the performances of the b-threshold policies. The corresponding LP of interest is given by

$$Z^{LP}(c) = \max_{\boldsymbol{x} \in P}\{x_1 - cx_2\} \tag{55}$$

In our main result we shall require the *critical threshold function* $b^*(\cdot)$ given by

$$b^*(c) = \min\{b \geq 0;\ c^{b+1} \leq c\}$$

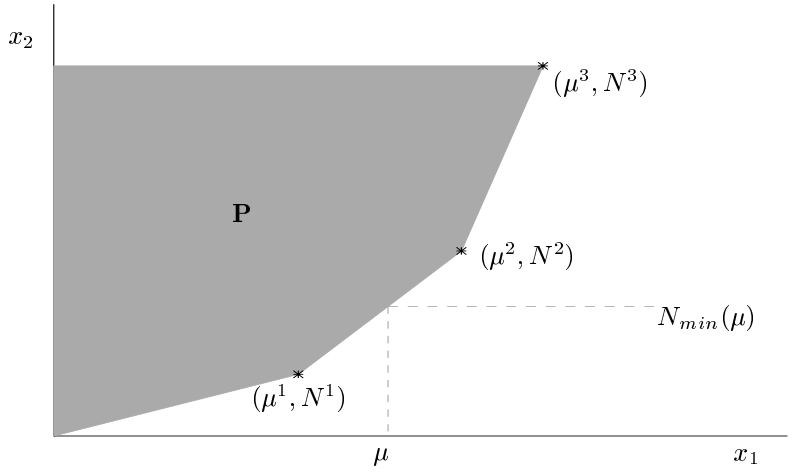

Figure 4: The threshold polygon P

Theorem 12 (Threshold optimality via the achievable region)
(i) $Z^{LP}(c) = \mu^{b^*(c)} - cN^{b^*(c)},\ c > 0;$

(ii) $X \subseteq P$;

(iii) $Z^{LP}(c) = Z^{OPT}(c),\ c > 0$;

(iv) The stochastic optimisation problem in (50) is solved by the $b^*(c)$*-threshold policy,* $c > 0$;

(v) $X = \bar{P}$*, the closure of* P.

Outline Proof

(i) follows by considering the dual LP of (55) through a standard complementary slackness argument which makes use of the properties described in Lemma 11(i),(ii);

(ii) is an immediate consequence of Lemma 11(iii);

(iii) It follows from (ii) that $Z^{OPT}(c) \leq Z^{LP}(c)$. However, from (i), $Z^{LP}(c)$ is achieved by the performance $(\mu^{b^*(c)}, N^{b^*(c)})$ of the $b^*(c)$-threshold policy. This yields $Z^{LP}(c) \leq Z^{OPT}(c)$ and (iii) follows;

(iv) is an immediate consequence of (i) and (iii);

(v) Plainly, from (ii) we have that $X \subseteq \bar{P}$. To secure the reverse inclusion, the reader is referred to Figure 4 for assistance. Observe that any point on the lower boundary of P is the performance of a policy which randomises between (at most) two threshold policies. Hence the lower boundary of P is contained in X. Note also that all points $(0, N)$ are in X where N is a non-negative integer. To see this, consider a policy which guarantees that the system enters the state in which N customers are present in finite time and which then freezes the system by closing down both the input and the output. Plainly there is such a policy and its performance is $(0, N)$.

By appealing further to randomisations, we infer that the convex hull of the lower boundary of P together with $\{(0, N), N \geq 0\}$ is contained in X. We deduce that $\bar{P} \subseteq X$ and (v) follows. □

We finally broach the issue raised above of the minimum WIP level, $N_{min}(\mu)$ required to achieve a target throughput level μ. From Theorem 12 we can write

$$\begin{aligned} N_{min}(\mu) &= \min\{N^u;\ \mu^u = \mu,\ u \in \mathcal{U}\} \\ &= \min\{N;\ (\mu, N) \in X\} \\ &= \min\{N;\ (\mu, N) \in \bar{P}\}. \end{aligned} \tag{56}$$

The minimisation in (56) is achieved on the lower boundary of P. See Figure 4. Corollary 13 follows easily. We write $\mu^\infty = \lim_{b\to\infty} \mu^b$.

Corollary 13

$N_{min}(\mu)$ is a piecewise linear function of μ over the range $[0, \mu^\infty)$ given by

$$N_{min}(\mu) = N^b + \frac{1}{c^{b+1}}(\mu - \mu^b), \quad \mu^b \leq \mu \leq \mu^{b+1}, \quad b \geq 0.$$

Consider now a general stochastic system with performance summarised by a throughput-WIP pair (μ^u, N^u). The stochastic optimisation problem of interest is still (50) and the general system continues to be furnished with a class of threshold policies whose associated performances are (μ^b, L^b), $b \geq 0$. Niño-Mora (1998) describes what needs to be true in general of the set $\{(\mu^b, N^b),\ b \geq 0\}$ for the achievable region to be a *threshold polygon* (as in the above example) whose vertices are the performances of threshold policies. When these conditions are met, threshold policies will be optimal for linear objectives and the minimum WIP level, $N_{min}(\mu)$ will be piecewise linear, as in Corollary 13.

6 Plans for future work

Current plans for further development of the achievable region approach by the authors and coworkers include work in the following three major areas.

(i) *Primal-dual approach*

As mentioned at the end of Section 2, the methodology underlying the performance guarantee in Corollary 2 is derived from the primal-dual structure of LP. The method works by constructing both a heuristic solution to an appropriately defined (primal) LP related to the stochastic optimisation problem of interest and a feasible solution to the dual of a relaxation of it. Our goal is to establish this approach as a central methodology in the analysis of heuristic policies for the control of stochastic systems within achievable region methodology both by extending its application to approximately GCL systems (like those discussed in Section 3) and by introducing it as an analytical tool in new contexts.

(ii) *Load balancing*

There is huge scope for further development of the work in Section 4. We shall mention just two directions for such work: firstly, the delay constrained problem of Ross and Yao (1991) mentioned at the conclusion of Section 4 is both compelling from the perspective of applications, but also a formidable technical challenge when placed in the context of GCL systems. Secondly, in more complex systems than those discussed above for which the model for each station only approximately satisfies GCL then functions $\tilde{Z}_m$ approximating the optimal costs Z_m^{OPT} will have the kind of convexity properties discussed in Section 4. A natural load balancing heuristic can be obtained by replacing Z_m^{OPT} by $\tilde{Z}_m$ in (40). Further work will include the development of performance guarantees for such heuristic approaches.

(iii) *Extension of the approach to new areas*

Strict priority policies for the service of customers in a queueing network may be unattractive because of the heavy penalties they impose on low priority jobs/customers. The latter suffer not only large queues and response times but, perhaps more significantly, large variances in these quantities. Natural formulations to ameliorate this would seek policies to minimise the usual time average linear holding cost rate subject to constraints on variance or to incorporate quadratic terms in the objective. We have begun work in this challenging area and believe that the achievable region approach has an important role to play. Achievable region methodology will also be introduced as an analytical tool for developments of the models discussed in Section 5 to accommodate scheduling of the work in progress (WIP) in addition to control of the arrivals process.

Acknowledgement

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 for supporting the work of the second author through the award of grants GR/K03043 and GR/M09308. We would also like to thank colleagues in the Department of Statistics, Newcastle University for their constructive comments on earlier drafts of the paper. The work of the third author was initiated during his stay at the Center for Operations Research and Econometrics (CORE) of the Université catholique de Louvain, Belgium, where it was supported by EC individual Marie Curie Postdoctoral Fellowship no. ERBFMBICT961480.

References

Atkins, D. and Chen, H. (1995). Performance evaluation of scheduling control of queueing networks: fluid model heuristics. *Queueing Systems*, 21:391–413.

Bertsimas, D. (1995). The achievable region method in the optimal control of queueing systems - formulations, bounds and policies. *Queueing Systems*, 21:337–389.

Bertsimas, D. and Niño Mora, J. (1996). Conservation laws, extended polymatroids and multi-armed bandit problems: a polyhedral approach to indexable systems. *Mathematics of Operations Research*, 21:257–306.

Bertsimas, D., Paschalidis, I. C., and Tsitsiklis, J. N. (1995). Branching bandits and Klimov's problem: achievable region and side constraints. *IEEE Transactions on Automatic Control*, 40(12):2063–2075.

Chen, H. and Yao, D. D. (1990). Optimal intensity control of a queueing system with state-dependent capacity limit. *IEEE Transactions on Automatic Control*, 35(4):459–464.

Coffman, E. and Mitrani, I. (1980). A characterization of waiting time performance realizable by single server queues. *Operations Research*, 28:810–821.

Dacre, M. J. and Glazebrook, K. D. (1998). An approach to load balancing in distributed systems. (in preparation).

Federgruen, A. and Groenevelt, H. (1988). Characterisation and optimisation of achievable performance in queueing systems. *Operations Research*, 36:733–741.

Garbe, R. and Glazebrook, K. D. (1998). Submodular returns and greedy heuristics for queueing scheduling problems. *Operations Research*. (to appear).

Garbe, R. and Glazebrook, K. D. (1998a). Stochastic scheduling with priority classes. *Mathematics of Operations Research*, 23:119–144.

Gelenbe, E. and Mitrani, I. (1980). *Analysis and Synthesis of Computer Systems*. Academic Press, London.

Gelenbe, E. and Pekergin, F. (1993). Load balancing pragmatics. Technical report, EHEI, Université René Descartes.

Gittins, J. C. (1979). Bandit processes and dynamic allocation indices (with discussion). *Journal of the Royal Statistical Society*, B 41:148–177.

Gittins, J. C. (1989). *Multi-armed Bandit Allocation Indices*. Wiley, New York.

Glazebrook, K. D. (1982). On a sufficient condition for superprocesses due to Whittle. *Journal of Applied Probability*, 19:99–110.

Glazebrook, K. D. and Garbe, R. (1998). Almost optimal policies for stochastic systems which almost satisfy conservation laws. *Annals of Operations Research*. (to appear).

Glazebrook, K. D. and Niño Mora, J. (1997). Scheduling multi-class queueing networks on parallel servers: approximate and heavy traffic optimality of Klimov's rule. In Burkard and Woeginger, editors, *Algorithms - ESA 97*, volume 1284 of *Springer Lecture Notes in Computer Science*, pages 232–245.

Harrison, J. M. (1996). The BIGSTEP approach to flow management in stochastic processing networks. In Kelly, Zachary, and Ziedens, editors, *Stochastic Networks: Theory and Applications*, number 4 in Royal Statistical Society Lecture Note Series, pages 57–90.

Harrison, J. M. and Nguyen, V. (1993). Brownian models of multiclass queueing networks: current status and open problems. *Queueing Systems: Theory and Applications*, 13:5–40.

Harrison, J. M. and Wein, L. (1989). Scheduling networks of queues: heavy traffic analysis of a simple open network. *Queueing Systems*, 5:265–280.

Klimov, G. P. (1974). Time sharing systems I. *Theory of Probability and its Applications*, 19:532–551.

Liu, Z. and Towsley, D. (1994). Optimality of the round-robin routing policy. *Journal of Applied Probability*, 31:466–478.

Maglaras, C. (1997). Design of dynamic control policies for stochastic processing networks via fluid models. *IEEE proceedings CDC97.* (to appear).

Niño-Mora, J. (1998). On the throughput-WIP trade-off in queueing systems, diminishing returns and the threshold property: a linear programming approach. Economics Working Paper 276, Universitat Pompeu Fabra.

Ross, K. W. and Yao, D. D. (1991). Optimal load balancing and scheduling in a distributed computer system. *Journal of the Association for Computing Machinery*, 38:676–690.

Shanthikumar, J. G. and Yao, D. D. (1992). Multi-class queueing systems: polymatroidal structure and optimal scheduling control. *Operations Research*, 40:293–299.

Tantawi, A. N. and Towsley, D. (1984). A general model for optimal static load balancing in star network configurations. *Performance '84*, pages 277–291.

Tantawi, A. N. and Towsley, D. (1985). Optimal static load balancing in distributed computer systems. *Journal of the Association for Computing Machinery*, 32:445–465.

Tcha, D. W. and Pliska, S. R. (1977). Optimal control of single-server queueing networks and multi-class M/G/1 queues with feedback. *Operations Research*, 25:248–258.

Walrand, J. (1988). *An Introduction to Queueing Networks.* Prentice Hall, Englewood Cliffs, New Jersey.

Weber, R. R. (1978). On the optimal assignment of customers to parallel queues. *Journal of Applied Probability*, 15:406–413.

Weber, R. R. (1992). On the Gittins index for multi-armed bandits. *The Annals of Applied Probability*, 2:1024–1033.

Weiss, G. (1988). Branching bandit processes. *Probability in Engineering and Informational Sciences*, 2:269–278.

Weiss, G. (1992). Turnpike optimality of Smith's rule in parallel machine stochastic scheduling. *Mathematics of Operations Research*, 17:255–270.

Weiss, G. (1995). On almost optimal priority rules for preemptive scheduling of stochastic jobs on parallel machines. *Advances in Applied Probability*, 27:821–839.

Whittle, P. (1980). Multi-armed bandits and the Gittins index. *Journal of the Royal Statistical Society*, B 42:143–149.