J. Appl. Prob. **37,** 668–684 (2000)
Printed in Israel

OPEN-LOOP ROUTEING TO *M* PARALLEL SERVERS WITH NO BUFFERS

EITAN ALTMAN,* *INRIA*
SANDJAI BHULAI,** *Vrije Universiteit Amsterdam*
BRUNO GAUJAL,*** *LORIA*
ARIE HORDIJK,**** *Leiden University*

Abstract

In this paper we study the assignment of packets to M parallel heterogeneous servers with no buffers. The controller does not have knowledge on the state of the servers and bases all decisions on the number of time slots ago that packets have been sent to the different servers. The objective of the controller is to minimize the expected average cost. We consider a general stationary arrival process, with no independence assumptions. We show that the problem can be transformed into a Markov Decision Process (MDP). We further show under quite general conditions on cost functions that only a finite number of states have to be considered in this MDP, which then implies the optimality of periodic policies. For the case of two servers we obtain a more detailed structure of the cost and optimal policies. In particular we show that the average cost function is multimodular, and we obtain expressions for the cost for MMPP and MAP processes. Finally we present an application to optimal robot scheduling for Web search engines.

Keywords: Markov decision processes; multimodularity; optimal open-loop control; routeing

AMS 2000 Subject Classification: Primary 93E20
Secondary 90C40

1. Introduction

In a recent paper [10], Koole used the theory of MDPs with partial information to solve the problem of routeing packets optimally to M servers with different service times. The objective was to minimize losses or maximize the throughput, and policies were restricted to those where no information is available on the state of the servers. For the special case of symmetrical servers and arrival processes, the optimality of the round-robin policy was established. For the case of two servers, it was shown that a periodic policy whose period has the form $(1, 2, 2, 2, \ldots, 2)$ is optimal, where 2 means sending a packet to the faster server. Similar results have been obtained

Received 20 October 1999; revision received 10 February 2000.
* Postal address: INRIA B.P. 93, 2004 Route des Lucioles, 06902 Sophia Antipolis Cedex, France.
Email address: altman@sophia.inria.fr
** Postal address: Vrije Universiteit Amsterdam, De Boelelaan 1081a, 1081 HV Amsterdam, The Netherlands.
*** Postal address: LORIA B.P. 102, 54602 Villers-les-Nancy, France.
**** Postal address: Leiden University, 2300 RA Leiden, The Netherlands.

in a dual model of server assignment in [7] with a somewhat different cost. The results in [7, 10] heavily depend on the Markovian structure of the arrivals.

An alternative approach based on the multimodularity and Schur convexity of the cost functions has been presented in [4]. It enabled us to use a novel theory for optimization of multimodular functions [9, 2, 3] in order to obtain structural results for the optimal policy without the Markovian assumptions made in [7, 10]. The objective was to minimize losses or maximize the throughput.

In this paper we extend our framework to general convex functions, not necessarily decreasing (as in [4]). We show that under quite general conditions, the problem can be transformed into an MDP with a *finite* state space, even when the arrival times have a general stationary distribution. No independence assumptions are required. We show that packets should be routed to every server infinitely often, and that there exists a periodic optimal policy. We then present some further properties for the case of two servers. As a first application, we consider an optimal routeing problem in which losses are to be minimized. In addition to the structure of the optimal policy for this problem, we provide explicit expressions for the costs for MMPP and MAP arrival processes. Finally, we apply our framework to the problem of robot scheduling for Web search engines.

2. Problem formulation

Consider a system, to which packets arrive at times $(T_n)_{n\in\mathbb{N}}$. We use the convention that $T_1 = 0$ and we assume that the process $(\tau_n)_{n\in\mathbb{N}}$ of interarrival times $\tau_n = T_{n+1} - T_n$ is stationary (thus $(\tau_m, \dots, \tau_{m+h})$ and $(\tau_n, \dots, \tau_{n+h})$ are equal in distribution for all m, n and h).

Assume that the system has M parallel servers with no waiting room and independent service times, independent of the arrival process. Upon arrival of a packet a controller must route this packet to one of the M servers in the system. The service time has a general service distribution G_m when routed to server $m \in \{1, \dots, M\}$. If there is still a packet present at the server, where the packet is routed to, then the packet in service is lost (we call this the preemptive discipline). In the special case of an exponential distribution, one can consider instead the non-preemptive discipline in which the arriving packet is lost; the results below will still hold.

We assume that the controller, which wishes to minimize some cost function, has no information on the state of the servers (busy or idle). The only information which the controller possesses is its own previous actions, i.e. to which server previous packets were routed. We assume that this information does not include the actual time that has elapsed since a packet was last routed to that server. This assumption enables us to study the embedded Markov chain of the continuous-time process; i.e. we can consider the discrete-time process embedded at the epochs when packets arrive. Now the mathematical formulation of this problem is given by the following T-horizon Markov decision process.

Let $\mathcal{X} = (\mathbb{N} \cup \{\infty\})^M$ be the state space. The mth component x_m of $x \in \mathcal{X}$ denotes the number of arrivals since a packet was last routed to server m. Let $\mathcal{A} = \{1, \dots, M\}$ be the action space, where action $a \in \mathcal{A}$ means that a packet is routed to server a. Since actions are taken before state transitions, we have the following transition probabilities:

$$p(x' \mid x, a) = \begin{cases} 1, & \text{if } x'_a = 1 \text{ and } x'_m = x_m + 1 \text{ for all } m \neq a \\ 0, & \text{otherwise.} \end{cases}$$

Define the immediate costs by $c_t(x, a) = f_a(x_a)$, which reflects that the costs only depend on the chosen server and the state of that server for $t < T$. The terminal costs are given by

$$c_T(x) = \sum_{a \in \mathcal{A}} f_a(x_a) \tag{1}$$

(note that the terminal costs use the same functions f_a). Defining these terminal costs will be essential for the mathematical results, as will be illustrated later in Example 3.5. It has also a natural physical interpretation, as will be illustrated in Remark 6.2.

The set of histories at epoch t of this Markov decision process is defined as the set $\mathcal{H}_t = (\mathcal{X} \times \mathcal{A})^{t-1} \times \mathcal{X}$. A policy π is a set of decision rules $(\pi_1, \pi_2, \dots)$ with $\pi_t : \mathcal{H}_t \to \mathcal{A}$. For each fixed policy π and each realization h_t of a history, the variable A_t is given by $A_t = \pi_t(h_t)$. The variable X_{t+1} takes values $x_{t+1} \in \mathcal{X}$ with probability $p(x_{t+1} \mid x_t, a_t)$. (In general MDPs, the variables A_t and X_t are random; here they are deterministic since the transition probabilities take only values of 0's or 1's.) With these definitions the expected average cost criterion function $C(\pi)$ is defined by

$$C(\pi) = \limsup_{T \to \infty} \mathrm{E}_x^{\pi} \frac{1}{T} \sum_{t=1}^{T} c_t(X_t, A_t),$$

where $x = (x_1, \dots, x_M)$ is the initial state. Let Π denote the set of all policies. The Markov decision problem is to find a policy π^*, if it exists, such that $C(\pi^*) = \min\{C(\pi) \mid \pi \in \Pi\}$. This Markov decision model is characterized by the tuple $(\mathcal{X}, \mathcal{A}, p, c)$.

Example 2.1. Suppose that the controller wishes to minimize the number of lost packets (i.e. the number of preempted packets) per unit time. The cost function $f_m(n)$ will typically be a decreasing function in n, because a longer time interval between an assignment to server m results in a smaller probability that a packet that was previously assigned there is still in service.

Assume that the arrival process is a Poisson process with rate λ and that services at server i are exponentially distributed with rate μ_i independent of the other servers. Let S_i be a random variable which is exponentially distributed with rate μ_i. Then $f_m(n) = \mathrm{P}(S_m \geq \tau_1 + \cdots + \tau_n) = [\lambda/(\lambda + \mu_m)]^n$.

3. Structural results

In the setting described in Example 2.1 we obtained a decreasing cost function $f_m(x)$. In Section 5 we discuss this application in more detail. In Section 6 we describe another application of our model, where the obtained cost function is increasing. In this section, we make some general assumptions on the cost function in order to cover all these different structures. Moreover we investigate structural properties of the optimal policy.

Assume that all the f_m are convex. Moreover, one of Conditions (2)–(4) defined below holds:

$$\lim_{x \to \infty} (f_m(x+1) - f_m(x)) = \infty, \qquad m = 1, \dots, M. \tag{2}$$

$$f_m \text{ are strictly convex and } \lim_{x \to \infty} (f_m(x) - a_m x) = C, \qquad m = 1, \dots, M \tag{3}$$

(by strictly convex we mean that for all x, $f_m(x+2) - f_m(x+1) > f_m(x+1) - f(x)$).

$$f_m(x) = a_m x + C, \qquad m = 1, \dots, M, \tag{4}$$

where $a_m \geq 0$ and C are constants (and C does not depend on m). Note that Condition (4) is not included in (3): it violates its first part. Condition (2) covers the case where f_m grows more than linearly, whereas (3)–(4) cover the case where f_m grows asymptotically linearly. These conditions are complementary to (2) since any one of them implies that $\lim_{x\to\infty}(f_m(x+1) - f_m(x)) < \infty$. In Conditions (2) and (3), f_m is strictly convex.

Theorem 3.1. *Assume that one of Conditions (2)–(4) holds. Then:*

(i) *There exists an optimal policy that uses every server infinitely many times, thus* $\sup\{j \mid \pi_j = m\} = \infty$ *for* $m \in \{1, \dots, M\}$.

(ii) *There exists a periodic optimal policy.*

Before proving the theorem we note that there are certain ways in which the Conditions (3)–(4) cannot be relaxed. We illustrate this below with cost functions for which the above theorem does not hold.

Example 3.2. Consider the case of two servers with the costs

$$f_i(x) = a_i\, x + b_i\, \exp(-d_i\, x) + c_i, \quad i = 1, 2,$$

where $c_1 < c_2$, and where $b_i \geq 0$ and $d_i > 0$ are some constants (as follows from the next remark, the sign of a_i is not important). Then for a sufficiently small value of b_1, the policy that always routes to server 1 is the only optimal policy for any finite horizon T.

Indeed, assume first $b_i = 0$ for all i and let u be any policy that routes at its nth step to server 2. By changing the action at this step into an assignment to server 1 we gain $c_2 - c_1$.

By continuity of the cost in the b_i's, we also gain a positive amount using this modified policy if $b_i \neq 0$ provided the b_i's are sufficiently small. Hence, for b_i sufficiently small, a policy cannot be optimal if it does not always route to server 1.

When using the average cost, the cost is no longer affected by any changes in the actions provided that the frequency of such changes converges to zero. Hence, for the average cost, there may be other policies that are optimal, but still, any policy for which the fraction of customers routed to server 2 does not converge to 0 cannot be optimal.

We conclude that we cannot relax (3) or (4) and replace C by C_m.

Remark 3.3. Note that when the cost $f_i(x)$ contains a linear term $a_i x$ then the total accumulated cost that corresponds to the term $a_i x$ over any horizon of length T is $a_i(T + x_i)$, where $x = (x_1, x_2)$ is the initial state. This part does not depend on the policy. If we use a policy π and then modify it by changing an assignment at time $t < T$ from server i to server $j \neq i$ then the linear part of the cost at time t under the modified policy decreases by $a_i\, x_i(t) - a_j\, x_j(t)$, but it then increases by the same amount at time $t + 1$. Thus the accumulated linear cost is independent of the policy. (Note that this argument is valid due to the definition of the cost at time T in (1).)

Example 3.4. Consider the case of two servers with the costs $f_1(x) = a_1 x$ and $f_2(x) = \exp(-d_2 x)$. For any finite horizon T and for $d_2 > 0$, the only optimal policy is the one that always routes to server 1. Note that the average cost until time T of the policy that always routes to server 1 is

$$\frac{T f_2(1) + f_1(T)}{T} = \mathrm{e}^{-d_2} + a_1.$$

The average cost of the policy that always routes to server 2 is

$$\frac{T f_1(1) + f_2(T)}{T} = a_1 + \frac{\mathrm{e}^{-d_2 T}}{T}.$$

Again, for the average cost there are other optimal policies but they have to satisfy the following: the fraction of customers routed to queue 2 by time T should converge to zero as $T \to \infty$.

This illustrates the necessity of the first part of Condition (3). For $d_2 < 0$, the only optimal policy is the one that always routes to server 2.

Next we present an example to illustrate the importance of the terminal cost.

Example 3.5. Assume that there are two servers, and that the costs are given by $f_1(x) = x^2$ and $f_2(x) = 2x^2$. Assume that the terminal costs $c_T(x)$ are zero, then the policy that always routes to server 1 is optimal.

Proof of Theorem 3.1. First suppose the cost function satisfies Condition (4). Interchanging assignments between any two servers for any finite horizon does not result in changes in cost for that horizon, due to the linearity of the cost function and the terminal cost. Hence any periodic policy that routes packets to all servers is optimal.

We consider next Conditions (2) and (3). Instead of describing a policy using a sequence π, we use an equivalent description using time distances between packets routed to each server. More precisely, given an initial state x, define the jth instant at which a packet is routed to server m by

$$\eta^m(0) = -x_m \quad \text{and} \quad \eta^m(j) = \min\{i \mid \max(\eta^m(j-1), 0) < i \leq T \text{ and } \pi_i = m\},$$

for $j \in \mathbb{N}$ (the minimum of an empty set is taken to be infinity). Define the distance sequence δ^m by $\delta^m = (\delta_1^m, \delta_2^m, \dots)$, with $\delta_j^m = \eta^m(j) - \eta^m(j-1)$, for $j \in \mathbb{N}$. (For simplicity we do not include the T in the notation.)

Let π be an arbitrary policy and m be an arbitrary server. Assume that the distance sequence $\delta = \delta^m$ for this server has the property that $\limsup_{T\to\infty}\{\delta_j \mid j \in \mathbb{N}\} = \infty$. We shall construct a policy π' with distance sequence $\delta^{\pi'}$ such that $\limsup_{T\to\infty}\{\delta_j^{\pi'} \mid j \in \mathbb{N}\}$ is finite and $C(\pi') \leq C(\pi)$.

Assume first that f satisfies Condition (2). Choose n_0 such that for all $n > n_0$

$$\min_{1\leq k\leq 2M+1}(f_m(n+k) - f_m(k) - f_m(n)) > \max_{1\leq k<2M+1}(f_l(2M+1) - f_l(k) - f_l(2M+1-k)),$$

for all l. Since the supremum of the distance sequence is infinity, there is a j (and T) such that $\delta_j > n + 2M + 1$. Consider the $2M+1$ consecutive assignments starting at $\eta^m(j-1)$. Since there are M servers, it follows that there is at least one server, to which a packet is assigned three times during that period, say m'. Denote the distance (or interarrival) times of assignments to m' by δ^1 and δ^2. Replace the second assignment to m' in this sequence by an assignment to server m. Denote the new distance (or interarrival) times to m by δ_j' and δ_j'' (if $\eta(j) = T$ then the distance δ_j'' is not a real interarrival time). Consider the cost for a horizon of length l where l is an arbitrary integer larger than $\eta(j-1) + n_0 + 2M + 1$,

$$\begin{aligned}[]
&[f_m(\delta_j) + f_{m'}(\delta^1) + f_{m'}(\delta^2)] - [f_m(\delta_j') + f_m(\delta_j'') + f_{m'}(\delta^1 + \delta^2)] \\
&\qquad = [f_m(\delta_j) - f_m(\delta_j') - f_m(\delta_j'')] - [f_{m'}(\delta^1 + \delta^2) - f_{m'}(\delta^1) - f_{m'}(\delta^2)] > 0,
\end{aligned}$$

where the last inequality follows from the choice of n_0.

Now consider Condition (3). Since by assumption the supremum of the distance sequence $\delta_l = \delta_l^m$, $l = 1, 2, \dots$ is infinity, there is a j (and T) such that $\delta_j > 2n + 2M + 1$, for some n. Let

$$p := \min\{f_m(k) + a_m - f_m(k+1) \mid m = 1, \dots, M,\ k = 1, \dots, M\}.$$

Note that p is positive, since Condition (3) implies that $(f_l(x) - a_l x - C)$ is positive and strictly decreasing (for all l). Now choose n such that $2q = 2(f_m(n) - a_m n - C) < p$. Note that this is possible, since $f_m(n) - a_m n - C$ goes to zero as n goes to infinity. Consider the $2M + 1$ consecutive assignments starting n units after $\eta(j-1)$. There is at least one server, to which a packet is assigned three times, say m'. Replace the second assignment to m' in this sequence by an assignment to server m.

Define the function $g_i(k) = f_i(k) - a_i k - C$ for all i and consider the cost for a horizon of length l where l is an arbitrary integer larger than $\eta(j-1) + 2n + 2M + 1$. The decrease in cost due to the interchange is

$$\begin{aligned}
&[f_m(\delta_j) + f_{m'}(\delta^1) + f_{m'}(\delta^2)] - [f_m(\delta'_j) + f_m(\delta''_j) + f_{m'}(\delta^1 + \delta^2)] \\
&\qquad = [g_m(\delta_j) + g_{m'}(\delta^1) + g_{m'}(\delta^2)] - [g_m(\delta'_j) + g_m(\delta''_j) + g_{m'}(\delta^1 + \delta^2)] \\
&\qquad > [g_m(\delta_j) + g_{m'}(\delta^1) + g_{m'}(\delta^2)] - [2g_m(n) + g_{m'}(\delta^1 + 1)] \\
&\qquad = g_m(\delta_j) + g_{m'}(\delta^2) + [g_{m'}(\delta^1) - g_{m'}(\delta^1 + 1)] - 2g_m(n) \\
&\qquad > g_m(\delta_j) + g_{m'}(\delta^2) + p - 2q > 0,
\end{aligned}$$

where δ^1, δ^2, δ'_j and δ''_j are defined as before. The first inequality follows from the fact that $n < \delta'_j$, $n < \delta''_j$, $\delta^1 + 1 \le \delta^1 + \delta^2$ and $f_m(x) - a_m x - C$ is decreasing. The second inequality follows from the definition of p. Since $f_m - a_m - C$ is positive it follows by construction of n that the last inequality holds.

Repeating the interchange procedure for every j for which $\delta_j > 2n + 2M + 1$ (when dealing with Condition (3)) or for which $\delta_j > n + 2M + 1$ (when dealing with Condition (2)) provides us with a policy π' such that $C(\pi') \le C(\pi)$ and $\sup\{\delta_j^{\pi'} \mid j \in \mathbb{N}\} < 2n + 2M + 1$. By repeating this procedure for every server, we get an optimal policy that visits a finite number of states. By Chapter 8 of [15] we know that the optimal policy can be chosen stationary. It follows that $\pi_n(h_n) = \pi_0(x_n)$. Since the state transitions are deterministic it follows that the optimal policy is periodic.

4. Multimodularity

Let $\mathbb{Z}$ be the set of integers. Let $e_i \in \mathbb{Z}^n$ for $i = 1, \dots, n$ denote the vector having all entries zero except for a 1 in the ith entry. Let d_i be given by $d_i = e_{i-1} - e_i$ for $i = 2, \dots, n$. Then the base of vectors for multimodularity is defined as the collection $\mathcal{B} = \{b_0 = -e_1, b_1 = d_2, \dots, b_{n-1} = d_n, b_n = e_n\}$. Following [9], a function g defined on $\mathbb{Z}^n$ is called multimodular if for all $x \in \mathbb{Z}^n$ and $b_i, b_j \in \mathcal{B}$ with $i \ne j$

$$g(x + b_i) + g(x + b_j) \ge g(x) + g(x + b_i + b_j). \tag{5}$$

We define the notion of an atom in order to study some properties of multimodularity. In $\mathbb{R}^n$ the convex hull of $n + 1$ affine independent points in $\mathbb{Z}^n$ forms a simplex. This simplex, defined on $x_0, \dots, x_n$, is called an atom if for some permutation σ of $(0, \dots, n)$

$$x_1 = x_0 + b_{\sigma(0)}, \quad x_2 = x_1 + b_{\sigma(1)}, \quad \dots, \quad x_n = x_{n-1} + b_{\sigma(n-1)}, \quad x_0 = x_n + b_{\sigma(n)}.$$

This atom is referred to as $A(x_0, \sigma)$ and the points $x_0, \dots, x_n$ are called the extreme points of this atom. Each unit cube is partitioned into $n!$ atoms and all atoms together tile $\mathbb{R}^n$. We can use this to extend the function $g : \mathbb{Z}^n \to \mathbb{R}$ to the function $\overline{g} : \mathbb{R}^n \to \mathbb{R}$ as follows. If $x \in \mathbb{Z}^n$

then define the function by $\overline{g}(x) = g(x)$. Thus $\overline{g}$ agrees with g on $\mathbb{Z}^n$. If x is not an extreme point of an atom, then x is contained in some atom A. The value of $\overline{g}$ on such a point $x \in \mathbb{R}^n$ is obtained as the corresponding linear interpolation of the values of g on the extreme points of A. The following theorem shows the relation between multimodularity and convexity.

Theorem 4.1. (See [9], [2], Theorem 2.1, [6].) *A function g is multimodular if and only if the function $\overline{g}$ is convex.*

Let $g : \mathbb{Z}^n \to \mathbb{R}$ be an objective function in a mathematical model which has to be minimized. In general this is a hard problem to solve. However if the function g is multimodular, then we can use a local search algorithm, which converges to the globally optimal solution. In [11] the local search algorithm has been proved for the search space $\mathbb{Z}^n$. The following theorem shows that it also holds for any convex subset S, which is a union of a set of atoms; thus in particular for $S = \mathbb{N}_0^n$ which is essential for our application. Multimodularity of g and convexity of $\overline{g}$ still hold when restricted to S. Multimodularity on this convex subset means that (5) must hold for all $x \in S$ and $b_i, b_j \in \mathcal{B}$ with $i \neq j$, such that $x + b_i$, $x + b_j$ and $x + b_i + b_j \in S$.

Theorem 4.2. *Let g be a multimodular function on S, a convex subset which is a union of atoms. A point $x \in S$ is a global minimum if $g(x) \leq g(y)$ for all $y \neq x$ which is an extreme point of $A(x, \sigma)$ for some σ.*

Proof. Let $x \in S$ be a fixed point. Now suppose that there is a $z \in \mathbb{R}^n$ such that $x + z \in S$ and $\overline{g}(x + z) < \overline{g}(x) = g(x)$. We show that there is an atom $A(x, \sigma)$ in S with an extreme point y such that $g(y) < g(x)$.

Since any n vectors of $\mathcal{B}$ form a basis of $\mathbb{R}^n$, we can write z as $z = \sum_{i=0}^n \beta_i b_i$. Furthermore, this can be done such that $\beta_i \geq 0$, since any $\beta_i b_i$ with $\beta_i < 0$ can be replaced by $-\beta_i \sum_{j=0, j\neq i}^n b_j$.

Now reorder the elements of $\mathcal{B}$ as $(b'_0, \ldots, b'_n)$ and the elements $(\beta_0, \ldots, \beta_n)$ as $(\beta'_0, \ldots, \beta'_n)$ such that $\beta'_0 \geq \cdots \geq \beta'_n \geq 0$ and $z = \sum_{i=0}^n \beta'_i b'_i$. Note that this notation is equivalent to the notation $z = \beta''_0 b'_0 + \beta''_1 (b'_0 + b'_1) + \cdots + \beta''_n (b'_0 + \cdots + b'_n)$ with all $\beta''_i \geq 0$ and with $b'_0 + \cdots + b'_n = 0$. Now fix a non-zero point $z' = \alpha z$ with $\alpha < 1$ such that $\alpha(\beta''_0 + \cdots + \beta''_{n-1}) \leq 1$. The set S is convex with $x, x + z \in S$, hence $x + z' \in S$. Since by Theorem 4.1 $\overline{g}$ is convex and $\overline{g}(x + z) < g(x)$, it follows that $\overline{g}(x + z') < g(x)$.

Let σ be the permutation induced by $(\beta''_0, \ldots, \beta''_n)$. Now consider the atom $A(x, \sigma)$, then by construction $x + z' \in A(x, \sigma)$. Since x is an extreme point and $\overline{g}$ is linear, there must be another extreme point, say y, such that $g(y) < g(x)$.

This theorem shows that to check whether x is a globally optimal solution, it is sufficient to consider all extreme points of all atoms with x as an extreme point. When a particular extreme point of such an atom has a lower value than x, then we repeat the same algorithm with that point. Repeating this procedure is guaranteed to lead to the globally optimal solution.

Every multimodular function is also integer convex (see Theorem 2.2 in [2]). One could wonder if local search also works with integer convexity instead of multimodularity, which is a stronger property. The next counterexample in $(\{0, 1, 2\})^2$ shows that this is not true and that multimodularity is indeed the property needed for using local search. Define the function g such that

$$
\begin{array}{lll}
g(0,2) = -1, & g(1,2) = 2, & g(2,2) = 5, \\
g(0,1) = 2, & g(1,1) = 1, & g(2,1) = 0, \\
g(0,0) = 5, & g(1,0) = 4, & g(2,0) = 3.
\end{array}
$$

One can easily check that g is an integer convex function, but not multimodular since

$$g((1,2)+b_0)+g((1,2)+b_2)=0<4=g((1,2))+g((1,2)+b_0+b_2).$$

Starting the local search algorithm at coordinate (2, 1) shows that all neighbours have values which are greater than 0. However, the global minimum is $g((0,2))=-1$.

Since all the extreme points of all atoms with x as an extreme point can be written as $x+\sum_{i=0}^{n}\alpha_i\, b_i$ with $\alpha_i\in\{0,1\}$, $b_i\in\mathcal{B}$, the neighbourhood of a point x consists of $2^{n+1}-2$ points (we subtract 2 because when $\alpha_i=0$ or when $\alpha_i=1$ for all i, then the points coincide).

Although the complexity of the local search algorithm is big for large n, the algorithm is worthwhile studying. First of all, compared to minimizing a convex function on the lattice, this algorithm gives a reasonable improvement. Secondly, the algorithm can serve as a basis for heuristic methods.

In the previous section we showed that the optimal policy for all stationary arrival processes is periodic. In this section we shall show that the value function is multimodular when restricted to periodic policies. This can then be used to get a much more detailed structure of optimal policies (in particular, the regularity of optimal policies in the case of two servers), see [2]. Moreover the local search algorithm can be applied to find an optimal policy.

We note that some results on multimodularity have been obtained already in [4] for each individual server with respect to the routeing sequence to that server. This allows us to obtain some structure for the optimal policies as in [3]. The novelty of the results here is that, for the case of two servers to which we restrict ourselves, multimodularity is obtained directly for the global cost of all servers rather than for each one separately.

The notation of the distance sequence can be beneficially used to approach the decision problem. After the first assignment to server m, the distance sequence δ^m for server m is periodic, say with period $d(m)$. Therefore in future discussions we will write $\pi=(\pi_1,\dots,\pi_n)$ for the periodic assignment sequence with period n and with a slight abuse of notation we denote the periodic distance sequence for server m by $\delta^m=(\delta_1^m,\dots,\delta_{d(m)}^m)$.

The periodicity reduces the cost function in complexity. Since we use the expected average cost function, we only have to consider the costs incurred during one period. It would be interesting to establish multimodular properties for any M. Unfortunately it is not clear how even to define multimodularity for $M>2$. We thus consider below $M=2$. The expected average cost is given by

$$g(\pi)=\sum_{m=1}^{M} g_m(\pi)=\frac{1}{n}\sum_{m=1}^{M}\sum_{j=1}^{d(m)} f_m(\delta_j^m). \tag{6}$$

It is tempting to formulate that $g_m(\pi)$ is multimodular in π for $m=1,2$. Note that this is not necessarily true, since an operation $v\in\mathcal{B}$ applied to π leads to different changes in the distance sequences for the different servers.

We shall thus use an alternative description for the average cost through the period of server 1. Define g'_m as follows:

$$g'_m(\pi)=\frac{1}{n}\sum_{j=1}^{d(1)} f_m(\delta_j^1). \tag{7}$$

We note that the function $g'_m(\pi)$ only looks at the distance sequence assigned to the first server with respect to π using cost function f_m. By the symmetry between the assignments to the two servers $g(\pi)$ can now be expressed as $g(\pi) = g_1'(\pi) + g_2'(\vec{3} - \pi)$. ($\vec{3}$ is the vector whose components are all 3.) Note that $\delta_j^1 = \delta_j^1(\pi)$ is a function of π, and we have

$$\delta_j^1(\vec{3} - \pi) = \delta_j^2(\pi).$$

We first prove that $g_m'(\pi)$ is multimodular in π. Then multimodularity of $g(\pi)$ follows as the sum of two multimodular functions.

Lemma 4.3. *Assume that f_m are convex. Let π be a fixed periodic policy with period n. Let $g_m'(\pi)$ be defined as in (7). Then $g_m'(\pi)$ is a multimodular function in π.*

Proof. Since π is a periodic sequence, the distance sequence $\delta = \delta^1$ is also a periodic function, say with period $p = d(1)$. Now, define the function h_j for $j = 1, \ldots, p$ by $h_j(\pi) = f_m(\delta_j)$. The function h_j represents the cost of the $(j+1)$th assignment to server m by looking at the jth interarrival time. We will first check the conditions for multimodularity for $\mathcal{V} = \{b_1, \ldots, b_{n-1}\}$.

Let $v, w \in \mathcal{V}$ with $v \neq w$. If neither of these elements changes the length of the jth interarrival time then $h_j(\pi) = h_j(\pi + v) = h_j(\pi + w) = h_j(\pi + v + w)$. Suppose that only one of the elements changes the length of the interarrival time, say v, then $h_j(\pi + v) = h_j(\pi + v + w)$ and $h_j(\pi) = h_j(\pi + w)$. In both cases the function $h_j(\pi)$ satisfies the conditions for multimodularity.

Now suppose that v increases and w decreases the length of the jth interarrival time by one. Then $\delta_j^1(\pi + v) - \delta_j^1(\pi) = \delta_j^1(\pi + v + w) - \delta_j^1(\pi + w)$. Since h_j is a convex function, it follows that $h_j(\pi + w) - h_j(\pi + v + w) \geq h_j(\pi) - h_j(\pi + v)$. Now the multimodularity condition in (5) directly follows by rearranging the terms. Since $g_m'(\pi)$ is a sum of $h_j(\pi)$ it follows that $g_m'(\pi)$, is multimodular for $\mathcal{V}$.

Now consider the elements b_0 and b_n and note that the application of b_0 and b_n to π splits an interarrival period and merges two interarrival periods respectively. Therefore

$$\begin{aligned}
n\, g_m'(\pi + b_0) &= n\, g_m'(\pi) - f_m(\delta_1) - f_m(\delta_p) + f_m(\delta_1 + \delta_p),\\
n\, g_m'(\pi + b_n) &= n\, g_m'(\pi) - f_m(\delta_p) + f_m(\delta_p - 1) + f_m(1),\\
n\, g_m'(\pi + b_0 + b_n) &= n\, g_m'(\pi) - f_m(\delta_1) - f_m(\delta_p) + f_m(\delta_1 + 1) + f_m(\delta_p - 1).
\end{aligned}$$

Now

$$\begin{aligned}
&n[g_m'(\pi + b_0) + g_m'(\pi + b_n) - g_m'(\pi) - g_m'(\pi + b_0 + b_n)]\\
&\qquad = [f_m(\delta_1 + \delta_p) + f_m(1)] - [f_m(\delta_1 + 1) + f_m(\delta_p)].
\end{aligned}$$

Let $k = \delta_1 + \delta_p + 1$. Since the function $f_m(x) + f_m(y)$ with $x + y = k$ is a symmetric and convex function, it follows from Proposition C2 of Chapter 3 of [13], that $f_m(x) + f_m(y)$ is also Schur-convex. Since $(\delta_1 + 1, \delta_p) \prec (\delta_1 + \delta_p, 1)$, the quantity above is non-negative.

In the case that we use $w = b_0$ and $v \in \mathcal{V}$ such that v does not alter δ_1, then it follows that $g_m'(\pi + v + w) = g_m'(\pi + v) + g_m'(\pi + w) - g_m'(\pi)$. The same holds for $w = b_n$ and $v \in \mathcal{V}$ such that v does not alter δ_p. Suppose that v does alter δ_1, then we have

$$\begin{aligned}
&n[g_m'(\pi + b_0) + g_m'(\pi + v) - g_m'(\pi) - g_m'(\pi + b_0 + v)]\\
&\qquad = [f_m(\delta_1 + \delta_p) + f_m(\delta_1 - 1)] - [f_m(\delta_1 + \delta_p - 1) + f_m(\delta_1)].
\end{aligned}$$

When v alters δ_p we have

$$n[g'_m(\pi + b_n) + g'_m(\pi + v) - g'_m(\pi) - g'_m(\pi + b_n + v)] \\ = [f_m(\delta_p + 1) + f_m(l)] - [f_m(\delta_p) + f_m(l + 1)]$$

for some $l < \delta_p$. Now by applying the same argument as in the case of b_0 and b_n we derive multimodularity of $g'_m(\pi)$ for the base $\mathcal{B}$.

Now we will prove that $g(\pi)$, which is given by $g(\pi) = g'_1(\pi)+g'_2(\vec{3}-\pi)$, is multimodular. The proof is based on the fact that if a function is multimodular with respect to a base $\mathcal{B}$, then it is also multimodular with respect to $-\mathcal{B}$.

Theorem 4.4. *Let g'_1 and g'_2 be multimodular functions. Then the function $g(\pi)$ given by $g(\pi) = c_1\, g'_1(\pi) + c_2\, g'_2(\vec{3} - \pi)$ for positive constants c_1 and c_2 is multimodular in π.*

Proof. Let $v, w \in \mathcal{B}$, such that $v \neq w$. Then

$$\begin{aligned} g(\pi + v) + g(\pi + w) & \\ &= c_1\, g'_1(\pi + v) + c_2\, g'_2(\vec{3} - \pi - v) + c_1\, g'_1(\pi + w) + c_2\, g'_2(\vec{3} - \pi - w) \\ &= c_1[g'_1(\pi + v) + g'_1(\pi + w)] + c_2[g'_2(\vec{3} - \pi - v) + g'_2(\vec{3} - \pi - w)] \\ &\geq c_1[g'_1(\pi) + g'_1(\pi + v + w)] + c_2[g'_2(\vec{3} - \pi) + g'_2(\vec{3} - \pi - v - w)] \\ &= c_1\, g'_1(\pi) + c_2\, g'_2(\vec{3} - \pi) + c_1\, g'_1(\pi + v + w) + c_2\, g'_2(\vec{3} - \pi - v - w) \\ &= g(\pi) + g(\pi + v + w). \end{aligned}$$

The inequality in the fourth line holds, since g'_1 is multimodular with respect to $\mathcal{B}$ and g'_2 is multimodular with respect to $-\mathcal{B}$.

5. Examples of arrival processes

In today's information and communication systems the traffic pattern may be quite complex, as it may represent a variety of data, such as customer phone calls, compressed video frames and other electronic information. Modern communication systems are designed to accommodate such a heterogeneous input and therefore the arrival process used in mathematical models is of crucial importance to the engineering and performance analysis of these systems. In this section we elaborate on the setting of Example 2.1 with different arrival processes and derive explicit formulae for the cost function for the corresponding arrival process.

Assume that the controller wishes to minimize the number of lost packets (i.e. the number of preempted packets) per unit time; note that this is equivalent to maximizing the throughput of the system. Furthermore let the services at server i be exponentially distributed with rate μ_i independent of the other servers. Since we know that there exists a periodic optimal policy, we can write the cost induced by using policy π by

$$g(\pi) = \frac{1}{n} \sum_{m=1}^{M} \sum_{j=1}^{d(m)} \left[\frac{\lambda}{\lambda + \mu_m} \right]^{\delta_j^m},$$

in the case that the arrival process is a Poisson process with parameter λ. In [10] it was shown that the optimal policy has a period of the form $(1, 2, \ldots, 2)$, where 2 is the faster server. In [1]

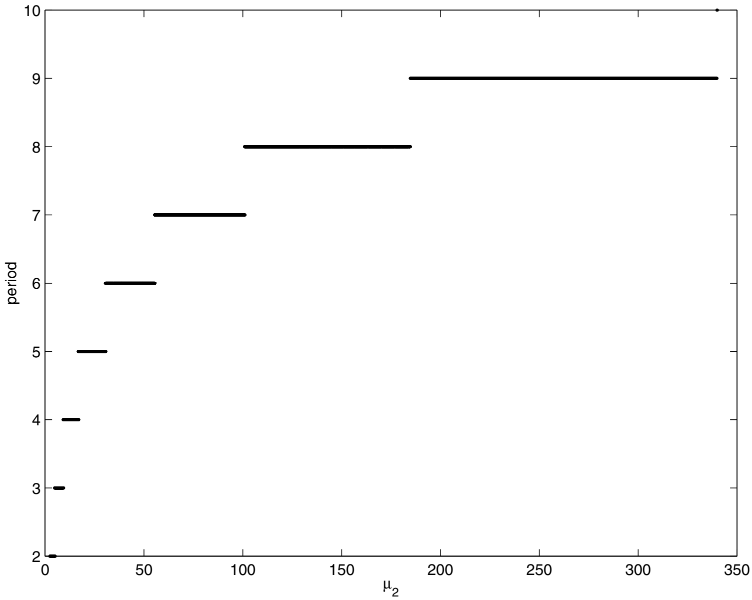

FIGURE 1: Relationship between n and μ_2.

this result was generalized for general stationary arrival processes. Hence suppose that $\lambda = 1$ and $\mu_1 = 1$, then the cost function can be parameterized by the period n and the server speed $\mu_2 \geq \mu_1$ given by

$$g(n, \mu_2) = \frac{1}{n}\left(\frac{1}{2}\right)^n + \frac{1}{n}\left(\frac{1}{1+\mu_2}\right)^2 + \frac{n-2}{n}\left(\frac{1}{1+\mu_2}\right).$$

By solving the equations $g(n, \mu_2) = g(n+1, \mu_2)$ for $n \geq 2$ we can compute the server rates μ_2 for which the optimal policy changes period. For example: the optimal policy changes from $(1, 2)$ to $(1, 2, 2)$ when $\mu_2 \geq 1 + \sqrt{2}$. The results of the computation are depicted in Figure 1.

5.1. Markov modulated Poisson process

An important class of models for arrival processes is given by Markov modulated models. The key idea behind this concept is to use an explicit notion of states of an auxiliary Markov process in the description of the arrival process. The Markov process evolves as time passes and its current state modulates the probability law of the arrival process. The utility of such arrival processes is that they can capture bursty inputs.

The Markov modulated Poisson process (MMPP) is the most commonly used Markov modulated model and is constructed by varying the arrival rate of a Poisson process according to an underlying continuous-time Markov process, which is independent of the arrival process. Therefore let $\{X_n \mid n \geq 0\}$ be a continuous-time irreducible Markov process with k states. When the Markov process is in state i, arrivals occur according to a Poisson process with rate λ_i. Let p_{ij} denote the transition probability to go from state i to state j and let Q be the infinitesimal generator of the Markov process. Let $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_k)$ be the matrix with the arrival rates on the diagonal and $\lambda = (\lambda_1, \dots, \lambda_k)$ the vector of arrival rates. With this notation, we can use the matrix analytic approach to derive a formula for $f_m(n)$.

Theorem 5.1. (Section 5.3, [14].) *The sequence* $\{(X_n, \tau_n) \mid n \geq 0\}$ *is a Markov renewal sequence with transition probability matrix* $F(t)$ *given by*

$$F(t) = \int_0^t \mathrm{e}^{(Q-\Lambda)u}\, \mathrm{d}u\, \Lambda = [I - \mathrm{e}^{(Q-\Lambda)t}]\,(\Lambda - Q)^{-1}\Lambda.$$

The interpretation of the matrix $F(t)$ is as follows. The elements $F_{ij}(t)$ are given by the conditional probabilities $\mathrm{P}(X_{n+1} = j,\ \tau_{n+1} \leq t \mid X_n = i)$ for $n \geq 1$. Since $F(t)$ is a transition probability matrix, it follows that $F(\infty)$ given by $(\Lambda - Q)^{-1}\Lambda$ is a stochastic matrix.

The MMPP is fully parameterized by specifying the initial probability vector q, the infinitesimal generator Q of the Markov process and the vector λ of arrival rates. Let the row vector s be the steady state vector of the Markov process. Then s satisfies the equations $sQ = 0$ and $se = 1$, where $e = (1, \dots, 1)$. Define the row vector $q = s\Lambda/s\lambda$. Then q is the stationary vector of $F(\infty)$ and makes the MMPP interval stationary (see [8]). This is intuitively clear since the stationary vector of $F(\infty)$ means that we obtain the MMPP started at an arbitrary arrival epoch.

In order to find an explicit expression for the cost function, we compute the Laplace–Stieltjes transform $f^*(\mu)$ of the matrix F. Since F is a matrix, we use matrix operations in order to derive $f^*(\mu)$, which will also be a matrix. The interpretation of the elements $f^*_{ij}(\mu)$ is given by $\mathrm{E}[\mathrm{e}^{-\mu\tau_{n+1}}\, \mathbf{1}_{\{X_{n+1}=j\}} \mid X_n = i]$ for $n \geq 1$, where $\mathbf{1}$ is the indicator function. Let I denote the identity matrix, then $f^*(\mu)$ is given by

$$\begin{aligned} f^*(\mu) &= \int_0^\infty \mathrm{e}^{-\mu I t} F(\mathrm{d}t) = \int_0^\infty \mathrm{e}^{-\mu I t}\mathrm{e}^{(Q-\Lambda)t}\,(\Lambda - Q)(\Lambda - Q)^{-1}\Lambda\,\mathrm{d}t \\ &= \int_0^\infty \mathrm{e}^{-(\mu I - Q + \Lambda)t}\,\mathrm{d}t\,\Lambda = (\mu I - Q + \Lambda)^{-1}\Lambda. \end{aligned}$$

Now we can compute $f_m(n) = \mathrm{P}(S_m \geq \tau_1 + \cdots + \tau_n) = \mathrm{E}\exp[-\mu\sum_{k=1}^n \tau_k]$. The next lemma shows that this is simply given by the product of $f^*(\mu)$ with itself. Note that we do not need the assumption of independence of the interarrival times to derive this result.

Lemma 5.2. *Let* $f^*(\mu)$ *be the Laplace–Stieltjes transform of* F*, where* F *is a transition probability matrix of a stationary arrival process. Then*

$$\mathrm{E}\exp\biggl(-\mu\sum_{k=1}^n \tau_k\biggr) = q\,[f^*(\mu)]^n\, e.$$

Proof. Define a matrix Q_n with entries $Q_n(i, j)$ given by

$$Q_n(i, j) = \mathrm{E}\biggl[\exp\biggl(-\mu\sum_{k=1}^n \tau_k\biggr)\mathbf{1}_{\{X_n=j\}}\,\biggm|\, X_0 = i\biggr].$$

Note that Q_1 is given by $f^*(\mu)$. By using the stationarity of the arrival process it follows that

$Q_n(i, j)$ is recursively defined by

$$\begin{aligned} Q_n(i, j) &= \sum_{l=1}^{m} Q_{n-1}(i, l)\ \mathrm{E}[\exp(-\mu\tau_n)\, \mathbf{1}_{\{X_n=j\}} \mid X_{n-1} = l\,] \\ &= \sum_{l=1}^{m} Q_{n-1}(i, l)\ \mathrm{E}[\exp(-\mu\tau_1)\, \mathbf{1}_{\{X_1=j\}} \mid X_0 = l\,] \\ &= \sum_{l=1}^{m} Q_{n-1}(i, l) \cdot Q_1(l, j). \end{aligned}$$

Note that the last line exactly denotes the matrix product, thus $Q_n = Q_{n-1} \cdot Q_1$. By induction it follows that $Q_n = (Q_1)^n$. Then it follows that

$$\mathrm{E}\ \exp\Big(-\mu \sum_{k=1}^{n} \tau_k \Big) = \sum_{i=1}^{m} \sum_{j=1}^{m} \mathrm{P}(X_0 = i)\ Q_n(i, j) = q\,[f^*(\mu)]^n\, e.$$

The last equation holds since the row vector q is the initial state of the Markov process and summing over all j is the same as right multiplying by e.

Hence $g(\pi)$ is given by

$$g(\pi) = \frac{1}{n} \sum_{m=1}^{M} \sum_{j=1}^{d(m)} q[(\mu_m I - Q + \Lambda)^{-1}\Lambda]^{\delta_j^m}\, e.$$

Note that although in case of two servers we know the structure of the optimal policy, it is not intuitively clear that it is optimal in the case of the MMPP. The following argument will clarify this statement. Suppose that one has an MMPP with two states. Choose the rates λ_1 and λ_2 of the Poisson processes such that the policies would have periods 2 and 3m, respectively, if the MMPP is not allowed to change state. One could expect that if the transition probabilities to go to another state are very small, the optimal policy should be a mixture of both policies. But this is not the case.

5.2. Markovian arrival process

The Markovian arrival process model (MAP) is a broad subclass of models for arrival processes. It has the special property that every marked point process is the weak limit of a sequence of Markovian arrival processes (see [5]). In practice this means that very general point processes can be approximated by appropriate MAPs. The utility of the MAP follows from the fact that it is a versatile, yet tractable, family of models, which captures both bursty inputs and regular inputs.

The MAP can be described as follows. Let $\{X_n \mid n \geq 0\}$ be a continuous-time irreducible Markov process with k states. Assume that the Markov process is in state i. The sojourn time in this state is exponentially distributed with parameter γ_i. After this time has elapsed, there are two transition possibilities. Either the Markov process moves to state j with probability p_{ij} with generating an arrival or the process moves to state $j \neq i$ with probability q_{ij} without generating an arrival.

This definition also gives rise to a natural description of the model in terms of matrix algebra. Define the matrix C with elements $C_{ij} = \gamma_i\, q_{ij}$ for $i \neq j$. Set the elements C_{ii} equal to $-\gamma_i$.

Define the matrix D with elements $D_{ij} = \gamma_i\, p_{ij}$. The interpretation of these matrices is given as follows. The elementary probability that there is no arrival in an infinitesimal interval of length $\mathrm{d}t$ when the Markov process moves from state i to state j is given by $C_{ij}\,\mathrm{d}t$. A similar interpretation holds for D, but in this case it represents the elementary probability that an arrival occurs. The infinitesimal generator of the Markov process is then given by $C + D$. Note that the MMPP can be derived by choosing $C = Q - \Lambda$ and $D = \Lambda$.

In order to derive an explicit expression for the cost function, we use the same approach as in the case of the MMPP. The transition probability matrix $F(t)$ of the Markov renewal process $\{(X_n, \tau_n) \mid n \geq 0\}$ is of the form (see [12])

$$F(t) = \int_0^t \mathrm{e}^{Cu}\,\mathrm{d}u = [I - \mathrm{e}^{Ct}]\,(-C^{-1}D).$$

Again the elements of the matrix $F(t)$ have the interpretation that $F_{ij}(t)$ is given by

$$\mathrm{P}(X_{n+1} = j,\ \tau_{n+1} \leq t \mid X_n = i) \quad \text{for } n \geq 1.$$

It also follows that $F(\infty)$ defined by $-C^{-1}D$ is a stochastic matrix. Let the row vector s be the steady state vector of the Markov process. Then s satisfies the equations $s(C + D) = 0$ and $se = 1$. Define the row vector $q = sD/sDe$. Then q is the stationary vector of $F(\infty)$. This fact can be easily seen upon noting that $sD = s(C + D - C) = s(C + D) - sC = -sC$. With this observation it follows that $q\,F(\infty) = (sDe)^{-1}\,s\,CC^{-1}D = q$. The MAP defined by q, C and D has stationary interarrival times.

The Laplace–Stieltjes transform $f^*(\mu)$ of the matrix F is given by

$$\begin{aligned} f^*(\mu) &= \int_0^\infty \mathrm{e}^{-\mu I t} F(\mathrm{d}t) = \int_0^\infty \mathrm{e}^{-\mu I t}\mathrm{e}^{Ct}\,(-C)(-C^{-1})D\,\mathrm{d}t \\ &= \int_0^\infty \mathrm{e}^{-(\mu I - C)t}\,\mathrm{d}t\; D = (\mu I - C)^{-1}D. \end{aligned}$$

The interpretation of f^* is given by the elements $f^*_{ij}(\mu)$, which represent the expectation $\mathrm{E}[\mathrm{e}^{\mu\tau_{n+1}}\mathbf{1}_{\{X_{n+1}=j\}} \mid X_n = i]$. By Lemma 5.2 we know that $f_m(n)$ is given by the product of f^*. Therefore the cost function, when using the MAP as arrival process, is given by

$$g(\pi) = \frac{1}{n}\sum_{m=1}^{M}\sum_{j=1}^{d(m)} q[(\mu_m I - C)^{-1}D]^{\delta^m_j}\,e.$$

6. Robot scheduling for Web search engines

In [4] we specified a routeing problem where the expected average weighted loss rate was to be minimized (or equivalently, the average weighted throughput or average weighted number of packets at service was to be maximized). This gave rise to an immediate cost of the form:

$$c(x, a) = \mathrm{E}\,\exp\left[-\mu_a \sum_{i=1}^{x_a} \tau_i\right].$$

Due to stationarity of the interarrival times, this cost function satisfies Condition (3) (with $a_m = C = 0$). We assume of course that τ_i are not all zero, which then implies the strict convexity of f_m. Indeed, denote

$$y = \exp\left[-\mu_a \sum_{k=2}^{m} \tau_k \right].$$

Let x be a state such that $x_a = m > 0$ for a particular action a. Since the interarrival times are a stationary sequence,

$$\begin{aligned} c(x, a) &= \mathrm{E}\, y e^{-\mu_a \tau_{m+1}} = \mathrm{E}\, y e^{-\mu_a \tau_1}, \\ c(x + e_a, a) &= \mathrm{E}\, y e^{-\mu_a [\tau_{m+1} + \tau_{m+2}]} = \mathrm{E}\, y e^{-\mu_a [\tau_1 + \tau_{m+1}]}, \\ c(x - e_a, a) &= \mathrm{E}\, y. \end{aligned}$$

Since the function $r(x) := y e^{-\mu_a x}$ is convex in x, it follows that $r(\tau_1 + z) - r(z)$ is increasing in z, so that

$$r(\tau_1 + \tau_{m+1}) - r(\tau_{m+1}) \geq r(\tau_1) - r(0).$$

By taking expectations, this implies the convexity. Thus the results of the previous sections apply. In this section we present another application studied in [7] under assumptions that are more restrictive than ours.

The World Wide Web offers search engines, such as Altavista, Lycos, Infoseek and Yahoo, that serve as a database that allows information search on the Web. These search engines often use robots that periodically traverse a part of the Web structure in order to keep the database up-to-date.

We consider a problem where we assume that there is a fixed number of M Web pages. The content of page i is modified at time epochs that follow a Poisson process with parameter μ_i. The time a page is considered up-to-date by the search engine is the time since the last visit by the robot until the next time instant of modification; at this point the Web page is considered out-of-date until the next time it is visited by the robot. The times between updates by the robot are given by a sequence τ_n. In [7] these times are assumed to be i.i.d., but in our framework we may allow them to form a general stationary sequence.

Let r_i denote the obsolescence rate of page i, i.e. the fraction of time that page i is out-of-date. Then the problem is to find an optimal visiting schedule such that the sum of the obsolescence rates r_i weighted by specified constants c_i is minimized. A reasonable choice for the weights c_i would be the customer page-access frequency, because the total cost then represents the customer total error rate. The case where the customer access frequency $c_i = k\,\mu_i$ is proportional to the page-change rate μ_i is reasonable under this interpretation, since the greater the interest for a particular page is, the more likely the frequency of page modification.

We now show that this problem is equivalent to the problem studied in Section 2. Indeed, the robot can be considered as the controller in the previous problem. An assignment of the nth packet to server i in the original problem corresponds with sending the robot to page i and requires τ_n time in order to complete the update. The lengths of the periods that page i is up-to-date correspond to the service times of packets before server i in the original problem. Page i is considered to be out-of-date when server i is idle.

Let S_i be an exponential random variable with parameter μ_i. The cost which one incurs when sending a packet to server a should reflect the expected obsolescence time. Some straightforward computations yield

$$c(x,a) = k\,\mu_a \mathrm{E}\Bigg[\sum_{i=1}^{x_a} \tau_i - S_a\Bigg]^+ = k\,\mu_a\,\mathrm{E}\Bigg[\sum_{i=1}^{x_a} \tau_i - S_a\Bigg] + k\,\mu_a\,\mathrm{E}\Bigg[S_a - \sum_{i=1}^{x_a} \tau_i\Bigg]^+$$

$$= k\,\mu_a\,x_a \mathrm{E}\,\tau_1 + k\left[\mathrm{E}\exp\bigg(-\mu_a \sum_{i=1}^{x_a} \tau_i\bigg) - 1\right].$$

This cost function clearly satisfies Condition (3). Hence the theorems from the previous sections can indeed be applied to this scheduling problem.

Remark 6.1. The assumption that the weights c_i are proportional to the page-change rates μ_i is essential in this problem. The cost function for general c_i is given by

$$c(x,a) = c_a\,x_a \mathrm{E}\,\tau_1 + \frac{c_a}{\mu_a}\left[\mathrm{E}\exp\bigg(-\mu_a \sum_{i=1}^{x_a} \tau_i\bigg) - 1\right].$$

When the c_i are not proportional to the page-change rates, then the cost function is of the type mentioned in Example 3.2. Therefore if for some i, c_i/μ_i is sufficiently large (in comparison to others) then it becomes optimal never to update page i. This is an undesirable situation and it shows that the problem is not always well posed when the costs are not proportional to the page-change rates μ_i.

Remark 6.2. This problem of robot scheduling for Web search engines illustrates the importance of including the terminal cost c_T in terms of modelling. Indeed, for any finite horizon T, if we wish that the cost indeed represents the obsolescence time of a page, we have to make sure that if this page is never updated (or at least it stops being updated after some time), this will be reflected in the cost. It is easy to see that the terminal cost defined in Section 2 indeed takes care of that.

Acknowledgements

We wish to thank the referee for many helpful comments and suggestions. This work was supported by the Van Gogh project No. 98001 and INRIA Action Coopérative Maddes.

References

[1] Altman, E., Bhulai, S., Gaujal, B. and Hordijk, A. (1999). Optimal routing problems and multimodularity. Res. Rept RR–3727, INRIA.

[2] Altman, E., Gaujal, B. and Hordijk, A. (2000). Multimodularity, convexity and optimization properties. *Math. Operat. Res.* **25**, 324–347.

[3] Altman, E., Gaujal, B. and Hordijk, A. (1997). Balanced sequences and optimal routing. *J. Assoc. Comput. Mach.* **47**, 752–775.

[4] Altman, E., Gaujal, B., Hordijk, A. and Koole, G. (1998). Optimal admission, routing and service assignment control: the case of single buffer queues. *IEEE Conf. Decision Control*, Dec. 1998, Tampa, Florida.

[5] Asmussen, S. and Koole, G. (1993). Marked point processes as limits of Markovian arrival streams. *J. Appl. Prob.* **30**, 365–372.

[6] Bartroli, M. and Stidham Jr., S. (1987). Towards a unified theory of structure of optimal policies for control of network of queues. Tech. Rept, Dept of Operations Research, University of North Carolina, Chapel Hill.

[7] Coffman Jr., E. G., Liu, Z. and Weber, R. R. (1998). Optimal robot scheduling for Web search engines. *J. Scheduling* **1**, 15–19.

[8] Fisher, W. and Meier-Hellstern, K. S. (1992). The Markov-modulated Poisson process (MMPP) cookbook. *Perf. Eval.* **18**, 149–171.

[9] HAJEK, B. (1985). Extremal splitting of point processes. *Math. Operat. Res.* **10**, 543–556.

[10] KOOLE, G. (1999). On the static assignment to parallel servers. *IEEE Trans. Automat. Control* **44**, 1588–1592.

[11] KOOLE, G. AND VAN DER SLUIS, E. (1998). An optimal local search procedure for manpower scheduling in call centers. Tech. Rept WS-501, Vrije Universiteit Amsterdam.

[12] LUCANTONI, D., MEIER-HELLSTERN, K. S. AND NEUTS, M. F. (1990). A single-server queue with server vacations and a class of non-renewal arrival processes. *Adv. Appl. Prob.* **22**, 675–705.

[13] MARSHALL, A. W. AND OLKIN, I. (1979). *Inequalities: Theory of Majorization and its Applications*. Academic Press, New York.

[14] NEUTS, M. F. (1989). *Structured Stochastic Matrices of M/G/1 Type and their Applications*. Marcel Dekker, New York.

[15] PUTERMAN, M. L. (1994). *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley, New York.