Admission Control and Routing to Parallel Queues with Delayed Information via Marginal Productivity Indices

Peter Jacko
Universidad Carlos III de Madrid
Department of Statistics
Av. de Universidad 30
289 11 Leganés (Madrid), Spain
peter.jacko@uc3m.es

José Niño-Mora
Universidad Carlos III de Madrid
Department of Statistics
Av. de Universidad 30
289 11 Leganés (Madrid), Spain
jose.nino@uc3m.es

ABSTRACT

This paper addresses the problem of designing and computing a tractable index policy for dynamic job admission control and/or routing in a discrete time Markovian model of parallel loss queues with one-period delayed state observation, which comes close to optimizing an infinite-horizon discounted or average performance objective involving linear holding costs and rejection costs. Instead of devising some ad hoc indices, we deploy a unifying fundamental design principle for design of priority index policies in dynamic resource allocation problems of multiarmed restless bandit type, based on decoupling the problem into subproblems and defining an appropriate marginal productivity index (MPI) for each subproblem. In the model of concern, such subproblems represent admission control problems to a single queue with one-period feedback delay, for which the structure of optimal policies has been characterized in previous work as being of bi-threshold type, yet without giving an algorithm to compute the optimal thresholds. We deploy in such subproblems theoretical and algorithmic results on restless bandit indexation, which yields a fast algorithm that computes the MPI for a subproblem with a buffer size of n performing only $O(n)$ arithmetic operations. Such MPI values can be used both to immediately obtain the optimal thresholds for the subproblem, and to design an index policy for the admission control and/or routing problem in the multi-queue system. The results readily extend to models with infinite buffer space.

Categories and Subject Descriptors

G.3 [**Probability and Statistics**]: Markov processes;
G.3 [**Probability and Statistics**]: Queueing theory;
G.4 [**Mathematical Software**]: Efficiency

General Terms

Algorithms

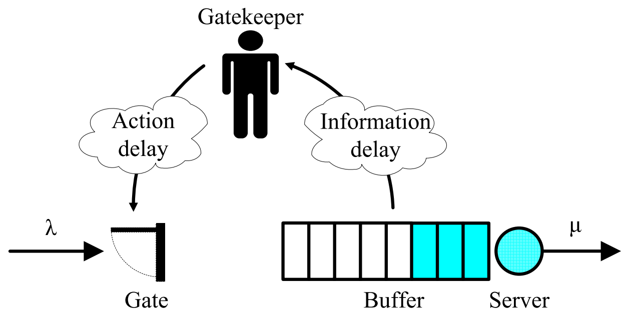


Figure 1: A design of the admission control problem with delay. The gatekeeper's job (work) consists of rejecting of some of the arriving customers at the gate.

Keywords

admission control, routing, delay, algorithm, index policy

1. INTRODUCTION

Sequential decision making arises in many real-life situations. Optimal solutions of such problems are usually very difficult to characterize due to their dynamic and stochastic patterns. Modeling these problems in the framework of Markov decision processes and using dynamic programming techniques have proved useful to obtain explicit structural properties of the optimal decisions in several sequential decision making problems, if the controller knows the full information history.

In practice, however, the controller may only know old information. Such is the case, for instance, in telecommunication networks or distributed computer systems, where the physical distance of network nodes creates a propagation delay. Other recent applications include long-distance-controlled robots, and situations in which an advanced processing of observations is necessary.

It is well-known [6] that sequential decision-making problems with delay can be formulated as Markov decision processes (MDP). Yet, the resulting model is usually significantly more complex and harder to analyze than that of the problem's no-delay analogue, while other approaches to tackle with the delay difficulty usually lead to a higher degradation in gain [6, 17].

In this paper we deal with the problem of admission control and routing to parallel queues with a one-period delay. Our goal is to design an optimal or nearly optimal policy, which can be efficiently implemented and provides intuitive insights. Such are the policies based on the concept of the *marginal productivity index* (cf. [11, 12, 13]). Our approach allows us to derive marginal productivity index policies for four problems at the same time:

(a) admission control and routing to parallel queues with delay;

(b) routing to parallel queues with delay;

(c) servers assignment problem with delay;

(d) admission control problem with delay.

Our analytical focus will be on the admission control problem with delay, whose marginal productivity indices will serve as building blocks for policies for the three remaining, harder problems. This problem is an important special case of a problem of routing to parallel queues with delay [9, 4] and of a delayed control problem of a network of interconnected subsystems [1].

A related problem of server allocation to multi-class customers with delayed queue observation was treated in [15]. An ad-hoc MDP formulation of the single-class problem with delay led to the derivation of structural properties of the marginal productivity indices, which were proposed to be used in a policy for a harder, multi-class problem. [12, 14] derived marginal productivity index policy for the problem of admission control and routing to parallel queues without delay, when the customers are patient and impatient, respectively. In all these papers, the corresponding marginal productivity index policy was shown to be nearly-optimal and to substantially outperform benchmark policies.

In this paper we develop an MDP formulation of the admission control problem with delay as a transformation of an MDP formulation of its no-delay version. This transformation is closed on the set of restless bandits (binary-action MDPs), and can be iteratively employed in order to obtain MDP formulations of restless bandits with arbitrary delays. The admission control problem with a delay of more than one period is, however, not treated here. To prepare the ground, we next describe the no-delay analogue of the admission control problem, a fundamental problem in queueing theory.

1.1 Admission Control Problem

Consider the following problem of admission control to a buffer of length $I \geq 1$ (including the service position, possibly infinite). Time is slotted into discrete-time *epochs* $t = 0, 1, 2, \ldots$ At each time epoch, the gatekeeper (controller) must choose between closing the gate and letting the gate open during the current *period* (i.e., between the current time epoch and the next time epoch).

A new customer arrives *at the beginning* of the period (just after the current decision epoch) to the gate with probability $0 < \lambda \leq 1$. A serviced customer leaves the buffer *at the end* of the period (just before the next decision epoch) with probability $0 < \mu < 1$. We further assume that a customer that is admitted to an empty buffer may leave the buffer with probability μ at the end of the same period.

If there are i customers queued (i.e., waiting or in service) at a decision epoch, then a holding cost c_i is accrued at that time epoch. Further, there is a rejection cost ν accrued at time epoch t for each customer that arrives during the period t and finds either a closed gate or a full buffer. Notice that rejection cost ν can alternatively be viewed as the gatekeeper's wage for rejecting an arriving customer. We will use both interpretations of ν as convenient along the paper.

Let $0 < \beta < 1$ be a one-period discount factor. The objective is to find a non-anticipative policy minimizing the expected sum of β-discounted costs (under the *discounted criterion*), and a non-anticipative policy minimizing the expected average cost per period (under the *time-average criterion*) over an infinite horizon. Our model under both the discounted and time-average criterion targets the trade-off between the throughput and delay (waiting time) experienced by the customers in the system, and is known as optimization of the throughput/delay criterion.

1.2 Admission Control Problem with Delay

Next we introduce a modified version of the above problem, in which the gatekeeper's action takes effect one period after the actual change in the queue length (see Figure 1). Such a delay can be well due to delayed action implementation, or due to delayed queue length observation, or both (see [3, 2, 8]). Yet, it is natural to assume that the gatekeeper knows perfectly the history of actions taken.

Costs are accrued for the true (though not observed) queue length process. We will show in section 2, where the problem's formal formulation is given, that this is equivalent to paying the costs with interest once the previous-epoch queue length is revealed. For instance, if there are i customers queued at a time epoch, then holding cost with interest c_i/β is paid once that information becomes available, i.e., at the next time epoch. Similarly, the rejection cost with interest ν/β is paid once rejected customers are observed. The remaining characteristics and the objectives remain as in the no-delay problem.

In their pioneering works, [3] and [9] analyzed the admission control problem with delay with infinite-length buffer under the discounted criterion. They independently (as noted in [9]) showed that there exists a bi-threshold optimal policy prescribing to close the gate if and only if the previous-epoch queue length is greater than an appropriate threshold depending on whether the gate was open (*open-gate threshold*) or closed (*closed-gate threshold*) during the previous period. Moreover (see [3]), these optimal thresholds are either equal or differ by one (the open-gate threshold is smaller than the closed-gate threshold).

To the best of our knowledge, there is no known algorithm to calculate such thresholds. Only [10] provided calculable upper bounds for the thresholds, when a specific relationship between parameters is satisfied, without giving any guarantee on preciseness of such upper bounds.

1.3 Routing to Parallel Queues with Delay

Suppose that there are $N \geq 2$ servers such that server n is endowed by a dedicated buffer of length $I_n \geq 1$ and serves customers at rate $0 < \mu_n < 1$. If there are i customers (waiting or in service) in the queue n at a decision epoch, then a holding cost $c_{n,i}$ is accrued at that time epoch.

At the beginning of each time period, a customer arrives to the router/controller with probability λ and is routed to a server chosen in the preceding time epoch by the router. Further, there is a rejection cost ν accrued if the routed customer finds a full buffer and the customer is lost.

The router, however, observes the actual queue lengths of all queues with a one period delay. As before, her routing decisions in all the previous time epochs are known to her. Given the complexity of the problem, our objective is to find an easily-implementable and nearly-optimal (or optimal, if possible) non-anticipative policy under the discounted criterion and under the time-average criterion over an infinite horizon.

[9] considered this problem when there are two symmetric queues with infinite buffers under the discounted criterion. They proved optimality of the following simple rule: *join the shortest expected queue* (JSEQ). They also presented an example showing that, in general, JSEQ is not optimal if the delay os greater than one period.

For the more general setting, in which the two buffers have heterogeneous holding costs and heterogeneous rejection costs, [4] proved that there is an optimal policy of threshold type. In particular, there exists a threshold function $l_1(i_2, a)$ nondecreasing in i_2 such that the optimal action is to route to server 1 if $i_1 < l_1(i_2, a)$, and to server 2 otherwise. This result holds both under the discounted and the time-average criteria, and also applies to the problem described in the following subsection.

1.4 Admission Control and Routing to Parallel Queues with Delay

In the problems of routing to N servers, it is often desirable to additionally control congestion. Indeed, heavy traffic may cause that all the buffers become full and the rejection costs reach undesirable levels. This is even more important in the presence of delayed queue length observations, because a possibly non-full buffer may not be observed and the customer may be wrongly routed to a server with a full buffer.

The two basic ways for controlling congestion, as noted in [16], are *blocking*, which randomly rejects a fixed fraction of arriving customers, and *gapping*, which admits a customer, then rejects all subsequent customers for a fixed time period, then repeats this process. He considers the problem without buffers (i.e., $I_n = 1$ for all n) and presents a dynamic blocking mechanism, which reacts to congestion (high utilization) in such a way that significantly decreases redials (i.e., the attempts to repetitively reach a server if having encountered full buffer), while throughput decreases modestly. [5] even presents an example with impatient customers in which rejections improve throughput of not-abandoned service attempts.

We therefore consider the problem of routing to parallel queues with delay, enhanced with the possibility to reject customers by the router. If such a router rejection occurs, rejection cost ν must be paid. The remaining problem parameters are as above. Intuitively, the router would reject an arriving customer if there is no non-full buffer, routing to which would be more profitable than rejecting her.

1.5 Servers Assignment Problem with Delay

Finally, of special interest is the problem of assignment of one of N available servers to an arriving customer, recovered as the special case of the routing problem with $I_n = 1$ for all servers n. The no-delay version was treated in [7], who showed optimality of the following rule: *join the fastest non-busy server*. As above, if there is a delay in the observation whether the servers are busy or free, it becomes untrivial to identify an optimal policy.

1.6 Contributions

In this paper we build on the results of [3] and [9] and prove the existence of an optimal *index policy*, a sort of dual concept to threshold policies, for the admission control problem with delay (see section 3 for the methodology overview). The main algorithmic idea is to construct the optimal policy (i.e., to calculate a set of all marginal productivity indices) by varying the rejection cost parameter ν over the set of real numbers.

We obtain that an optimal index policy, and hence also an optimal bi-threshold policy, can be found by performing $\mathcal{O}(I)$ arithmetic operations. Our algorithm is two orders of magnitude faster than the best existing general algorithm for optimal index policy calculation, which needs $(2/3)I^3 + \mathcal{O}(I^2)$ arithmetic operations after an initialization stage [13]. If the buffer's length is infinite, we propose a modification of the algorithm to find an optimal bi-threshold policy after performing a finite number of operations. Further, we present an algorithm to calculate the marginal productivity index for any particular state by performing at most $\mathcal{O}(\log_2 I)$ arithmetic operations.

In section 4 we further show that both the theoretical and the algorithmic results apply to the same problem under the time-average criterion. This, to the best of our knowledge, has not been proved before.

2. MDP FORMULATIONS

In order to see the analogy, in this section we formulate as a Markov decision process (MDP) both the admission control problem without delay and the admission control problem with delay.

2.1 Admission Control Problem

First we formulate as an MDP the no-delay admission control problem. Let $X(t)$ be the state process, denoting the *queue length* (including customers in service, if any) at time epoch t. If $a(t)$ denotes the action process, then the task at time epoch t is to choose between closing the gate ($a(t) = 1$) and letting the gate open ($a(t) = 0$). The MDP elements are as follows:

- The *action space* is denoted by $\mathcal{A} := \{0, 1\}$.
- The *state space* is $\mathcal{I} := \{0, 1, \dots, I\}$, where state $i \in \mathcal{I}$ represents the number of customers in the buffer or in service.
- Denoting by $\zeta := \lambda(1-\mu)$, $\eta := \mu(1-\lambda)$, and $\varepsilon := 1 - \zeta - \eta$, the *one-period transition probabilities* $p^a_{ij} := \mathbb{P}[X(t) = j | X(t-1) = i, a(t-1) = a]$ from state $1 \leq i \leq I-1$ to state j under action a are

$$p^0_{ij} = \begin{cases} \eta & \text{if } j = i-1 \\ \varepsilon & \text{if } j = i \\ \zeta & \text{if } j = i+1 \end{cases} \qquad p^1_{ij} = \begin{cases} \mu & \text{if } j = i-1 \\ 1-\mu & \text{if } j = i \end{cases} \tag{1}$$

and for the boundary cases, $p^1_{00} = 1$, and

$$p^0_{0j} = \begin{cases} 1-\zeta & \text{if } j = 0 \\ \zeta & \text{if } j = 1 \end{cases} \quad p^a_{Ij} = \begin{cases} \mu & \text{if } j = I-1 \\ 1-\mu & \text{if } j = I \end{cases} \tag{2}$$

The remaining transition probabilities are zero.

- If the queue length is $i \in \mathcal{I}$ and action $a \in \mathcal{A}$ is chosen, then the gatekeeper's *one-period reward* is defined as the negative of the expected holding cost at the current epoch,

$$R^a_i := -c_i.$$

At the same time, the gatekeeper's *one-period work* is defined as the expected number of rejected customers during the current period,

$$W^1_i := \lambda \qquad W^0_i := \begin{cases} \lambda & \text{if } i = I \\ 0 & \text{otherwise} \end{cases}$$

Thus, for rejection cost (gatekeeper's wage) ν, the *one-period overal cost* is

$$-R^a_i + \nu W^a_i = c_i + \lambda\nu a + (1-a)\mathbf{1}\{i = I\}\lambda\nu,$$

where $\mathbf{1}\{Y\}$ is the 0/1 indicator function of statement Y.

Given the definition above, we call state I *uncontrollable*, because in this state both the actions result in identical consequences (for having identical one-period reward, one-period work, and transition probabilities), and there is actually no decision to make. This is not the case for the remaining states, henceforth called *controllable*.

Finally, to ease later reference we summarize here our model parameters assumptions:

$$0 < \beta, \mu, \varepsilon, \zeta < 1, \qquad 0 < \lambda \leq 1, \qquad 0 \leq \eta < 1. \tag{3}$$

2.2 Admission Control Problem with Delay

In this subsection we follow the classic reformulation as MDP of problems with a discrete-time delay, which is a special case of *partially observed MDPs*, by augmenting the state space [6].

In the admission control problem with delay[1], the decision at epoch t is based on $\widetilde{X}(t) := (a(t-1), X(t-1))$, which is henceforth called an *augmented state* process. Thus, $\widetilde{X}(t)$ is the observed state at time epoch t, while $X(t)$ is the actual (hidden) queue length process. The MDP elements of the admission control problem with delay are as follows:

- The *action space* is $\mathcal{A}$ as in the no-delay problem.

- Recall that in the no-delay problem, state I is uncontrollable. Consequently, states $(0, I)$ and $(1, I)$ in the problem with delay are *duplicates*, having identical one-period reward, one-period work, and transition probabilities, so they can and should be merged into a unique state $(*, I)$. We therefore define the *augmented state space*

$$\widetilde{\mathcal{I}} := (\mathcal{A} \times \{0, 1, \dots, I-1\}) \cup \{(*, I)\}.$$

- The *one-period transition probabilities* are

$$\begin{aligned} p^{a'}_{(a,i),(b,j)} &:= \mathbb{P}\left[\widetilde{X}(t+1) = (b,j) | \widetilde{X}(t) = (a,i), a(t) = a'\right] \\ &= \mathbb{P}\left[X(t) = j, a(t) = b | X(t-1) = i, a(t-1) = a, a(t)\right. \\ &= p^a_{ij} \cdot \mathbf{1}\{a' = b\}. \end{aligned}$$

For the merged state $(*, I)$, we have

$$p^{a'}_{(a,i),(*,I)} := p^{a'}_{(a,i),(0,I)} + p^{a'}_{(a,i),(1,I)} = p^a_{ij}.$$

- If the current-epoch augmented state is (a, i), then the gatekeeper's *one-period reward* is defined as the negative of the expected holding cost at the previous epoch,

$$R^b_{(a,i)} := \beta(-c_i/\beta) = -c_i. \tag{4}$$

Similarly, the gatekeeper's *one-period work* is defined as the expected number of rejected customers during the previous period,

$$W^b_{(1,i)} := \lambda \qquad W^b_{(0,i)} := \begin{cases} \lambda & \text{if } i = I \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

Thus, for rejection cost (gatekeeper's wage) ν, the *one-period overal cost* is $-R^b_{(a,i)} + \nu W^b_{(a,i)}$.

To evaluate a policy π under the discounted criterion, we consider the following two measures. Let $g^\pi_{(a,i)}$ be the *expected total β-discounted work* (or, the expected total β-discounted number of rejected customers) if starting from state $(a(-1), X(-1)) := (a, i)$ under policy π,

$$g^\pi_{(a,i)} := \mathbb{E}^\pi_{(a,i)}\left[\sum_{t=0}^{\infty} \beta^t W^{a(t)}_{(a(t-1),X(t-1))}\right]. \tag{6}$$

Analogously for $f^\pi_{(a,i)}$, the *expected total β-discounted reward* if starting from state $(a(-1), X(-1)) := (a, i)$ under policy π,

$$f^\pi_{(a,i)} := \mathbb{E}^\pi_{(a,i)}\left[\sum_{t=0}^{\infty} \beta^t R^{a(t)}_{(a(t-1),X(t-1))}\right]. \tag{7}$$

If the rejection cost ν is interpreted as the wage paid to gatekeeper for each rejected customer, then the objective is to solve the following ν-wage problem for each ν:

$$\min_{\pi \in \Pi} -f^\pi_{(a,i)} + \nu g^\pi_{(a,i)}, \tag{8}$$

where Π is the set of all non-anticipative control policies.

Notice that the alternative one-period reward $R^b_{(a,i)}$ and the one-period work $W^b_{(a,i)}$ are independent of the current-epoch action (superscript b), therefore we will omit the superscript in the remaining sections.

3. METHODOLOGY

In the previous section we have formulated the admission control problem with delay as a binary-action Markov decision process (MDP), i.e., a *restless bandit*. Closing the gate will be referred to as the active action, and opening the gate as the passive action.

[1] We use the 'tilded' notation for the delayed version when not doing so might be confusing; note that state-dependent quantities are easy to distinguish since the original state is uni-dimensional, while the augmented state of the delayed problem is bi-dimensional.

In the following we closely follow the work-reward analysis proposed for restless bandits by [11, 12]. A more detailed review of the methodology and several applications are surveyed in [13]. Considered is the finite-length buffer problem under the discounted criterion. The solution to the problem under time-average criterion is obtained in the limit and is treated in subsection 4.3.

The MDP theory ensures existence of an optimal policy, which is stationary, deterministic and independent of the initial state. We represent a stationary deterministic policy in terms of a *active set*, i.e., the set of states in which it is prescribed to close the gate; in the remaining states it is prescribed to let the gate open. The task to find an optimal non-anticipative policy thus transforms into finding an optimal active set,

$$\max_{\mathcal{S}\in 2^{\widetilde{\mathcal{I}}}} f^{\mathcal{S}}_{(a,i)} - \nu g^{\mathcal{S}}_{(a,i)}. \tag{9}$$

For every rejection cost ν, the optimal policy is characterized by the unique solution vector $(v^*_{(a,i)}(\nu))_{(a,i)\in\widetilde{\mathcal{I}}}$ to the *Bellman equations* for all $(a,i)\in\widetilde{\mathcal{I}}$

$$v^*_{(a,i)}(\nu) = \max_{a'\in\mathcal{A}}\left[R_{(a,i)} - \nu W_{(a,i)} + \beta \sum_{(b,j)\in\widetilde{\mathcal{I}}} p^{a'}_{(a,i),(b,j)} v^*_{(b,j)}(\nu)\right], \tag{10}$$

where $v^*_{(a,i)}(\nu)$ denotes the optimal value of (8) starting at (a,i) under rejection cost ν. Hence, there exists a *maximal optimal active set* (i.e., a set of states in which it is optimal to close the gate) $\mathcal{S}^*(\nu)\subseteq\widetilde{\mathcal{I}}$ for (8), which is characterized by

$$\mathcal{S}^*(\nu) := \left\{(a,i)\in\widetilde{\mathcal{I}} : \sum_{(b,j)\in\widetilde{\mathcal{I}}} p^0_{(a,i),(b,j)} v^*_{(b,j)}(\nu) \le \sum_{(b,j)\in\widetilde{\mathcal{I}}} p^1_{(a,i),(b,j)} v^*_{(b,j)}(\nu)\right\}$$

Problem (9) can be viewed as a bi-criteria parametric optimization problem. Intuitively, if the rejection cost $\nu\to-\infty$, the optimal active set should be $\widetilde{\mathcal{I}}$, whereas if the rejection cost $\nu\to\infty$, the optimal active set should be the empty set. In fact, we set out to show a stronger, so-called *indexability* property: Active sets $\mathcal{S}^*(\nu)$ diminish monotonically from $\widetilde{\mathcal{I}}$ to the empty set as the rejection cost ν increases from $-\infty$ to ∞. Such a property was introduced in [18] for the restless bandits with one-periods works equal to 1 under the active action, and equal to 0 under the passive action, and extended to restless bandits without these limitations in [12].

Problem indexability is equivalent to existence of break-even values of the rejection cost ν, at which the maximal optimal active-set policy changes. Each break-even value is associated with the augmented state that leaves the maximal optimal active set when the rejection cost increases over this value. The break-even value, or *index*, of augmented state (a,i) is denoted by $\nu_{(a,i)}$. In that state, it is then optimal to close the gate if $\nu<\nu_{(a,i)}$, and it is optimal to let the gate open if $\nu>\nu_{(a,i)}$. When $\nu=\nu_{(a,i)}$, both opening and closing the gate are optimal. Since we have defined $\mathcal{S}^*(\nu)$ as the *maximal* optimal active set, state $(a,i)\in\mathcal{S}^*(\nu)$ if $\nu=\nu_{(a,i)}$, though this choice is arbitrary. Our objective is to identify the set of indices $\nu_{(a,i)}$ for all $(a,i)\in\widetilde{\mathcal{I}}$.

3.1 Exploiting Special Structure

$\widehat{\mathcal{S}}_0 := \widetilde{\mathcal{I}}$;
`for` $k=1$ `to` $2I+1$ `do`
  `pick` $(a_k,i_k)\in\arg\min\left\{\nu^{\widehat{\mathcal{S}}_{k-1}}_{(a,i)} : (a,i)\in\widehat{\mathcal{S}}_{k-1}\right.$
    `and` $\left.\widehat{\mathcal{S}}_{k-1}\setminus\{(a,i)\}\in\mathcal{F}\right\}$;
  $\widehat{\nu}_{(a_k,i_k)} := \nu^{\widehat{\mathcal{S}}_{k-1}}_{(a_k,i_k)}$;
  $\widehat{\mathcal{S}}_k := \widehat{\mathcal{S}}_{k-1}\setminus\{(a_k,i_k)\}$;
`end {for}`;
`{Output` $\{\widehat{\mathcal{S}}_k\}_{k=0}^{2I+1}, \{\widehat{\nu}_{(a_k,i_k)}\}_{k=1}^{2I+1}$`}`

Figure 2: Algorithmic scheme of $AG_{\mathcal{F}}$.

While one could test numerically whether a given instance is indexable and calculate the indices $\nu_{(a,i)}$ for all $(a,i)\in\widetilde{\mathcal{I}}$, we aim instead to establish analytically indexability of the admission control problem with delay in general. This will further allow us to achieve our second objective of obtaining a fast way of computing the indices. In this subsection we present how to exploit special structure of the model by aligning indexability to a known family of optimal bi-threshold policies.

Suppose that we postulate a family $\mathcal{F}\subseteq 2^{\widetilde{\mathcal{I}}}$ of active sets, satisfying certain connectivity conditions (see [13] for the details). Before presenting such a family for the admission control problem with delay, we review a test (deployed in section 4) to verify whether a postulated family $\mathcal{F}$ can be used to establish indexability, via the sufficient conditions termed PCL($\mathcal{F}$)-indexability introduced in [11, 12].

Let policy $\langle a,\mathcal{S}\rangle$ be the policy where action a is applied in the current period and policy $\mathcal{S}$ proceeds. Notice that policy $\langle a,\mathcal{S}\rangle$ implies that the next-epoch augmented state will be (a,j) for some state $j\in\mathcal{I}$. We define the *marginal work* of closing the gate instead of letting it open (or, of rejecting possible customers instead of admitting them), if starting from state (a,i) under active-set policy $\mathcal{S}$, as

$$w^{\mathcal{S}}_{(a,i)} := g^{\langle 1,\mathcal{S}\rangle}_{(a,i)} - g^{\langle 0,\mathcal{S}\rangle}_{(a,i)}, \tag{11}$$

i.e., as the increment in total work that results from closing the gate instead of opening it at current epoch. Analogously, we define the *marginal reward*,

$$r^{\mathcal{S}}_{(a,i)} := f^{\langle 1,\mathcal{S}\rangle}_{(a,i)} - f^{\langle 0,\mathcal{S}\rangle}_{(a,i)}, \tag{12}$$

as the analogous increment in total reward. Finally, we define the *marginal productivity rate*

$$\nu^{\mathcal{S}}_{(a,i)} := \frac{r^{\mathcal{S}}_{(a,i)}}{w^{\mathcal{S}}_{(a,i)}},$$

provided that the denominator does not vanish. As we will see, the denominator is positive for the admission control problem with delay. It can be shown that if the indices exist, then $\nu_{(a,i)}=\nu^{\mathcal{S}}_{(a,i)}$ for some active set $\mathcal{S}$, and therefore the indices are appropriately called the *marginal productivity indices*.

In Figure 2 is given a scheme of the *adaptive-greedy* algorithm $AG_{\mathcal{F}}$, which calculates the candidates for the maximal optimal active sets $\{\widehat{\mathcal{S}}_k\}_{k=0}^{2I+1}$ and the candidates for the marginal productivity indices $\{\widehat{\nu}_{i_k}\}_{k=1}^{2I+1}$. It is greedy, since in each step it picks the state with the lowest marginal productivity rate $\nu^{\widehat{\mathcal{S}}_{k-1}}_{(a_k,i_k)}$ (out of the feasible ones), and it is adaptive, because in each step it updates the marginal productivity rates for the actual active set $\widehat{\mathcal{S}}_{k-1}$.

Now we are ready to define PCL($\mathcal{F}$)-indexability, based on partial conservation laws (PCL), which determines both the computational and analytical value of the adaptive-greedy algorithm $AG_{\mathcal{F}}$.

Definition 1 (PCL($\mathcal{F}$)-indexability). *The admission control problem with delay is called* PCL($\mathcal{F}$)-indexable*, if*

(a) [Positive Marginal Works under $\mathcal{F}$] for each active set $\mathcal{S} \in \mathcal{F}$ and for each controllable state $(a,i) \in \widetilde{\mathcal{I}}$, the marginal work $w^{\mathcal{S}}_{(a,i)} > 0$;

and at least one of the following conditions holds:

(ii) for every rejection cost ν, there exists an optimal active set $\mathcal{S} \in \mathcal{F}$;

(ii') the output $\{\widehat{\nu}_{(a_k,i_k)}\}_{k=1}^{2I+1}$ of the algorithm $AG_{\mathcal{F}}$ are marginal productivity indices in nondecreasing order.

[11, 12, 13] introduced variants of PCL($\mathcal{F}$)-indexability and proved that PCL($\mathcal{F}$)-indexability implies indexability, i.e., the existence of marginal productivity indices, which are calculated as $\{\widehat{\nu}_{(a_k,i_k)}\}_{k=1}^{2I+1}$ by the adaptive-greedy algorithm $AG_{\mathcal{F}}$. To ease later reference, we summarize the above in the following theorem.

Theorem 1. *If marginal works are positive under $\mathcal{F}$ (cf. Definition 1(i)) for problem (8), then for that problem the following statements are equivalent:*

(a) for every rejection cost ν, there exists a maximal optimal active set $\mathcal{S} \in \mathcal{F}$;

(b) the problem is indexable and all active sets $\mathcal{S}^*(\nu) \in \mathcal{F}$;

(c) the output $\{\widehat{\nu}_{(a_k,i_k)}\}_{k=1}^{2I+1}$ of the algorithm $AG_{\mathcal{F}}$ are marginal productivity indices in nondecreasing order.

In section 4 we show that for a certain family $\mathcal{F}$ (defined below), Definition 1(i) holds and, given the existing results, Theorem 1(i) is true. In this way indexability of the admission control problem with delay will be established, and the algorithm $AG_{\mathcal{F}}$ can be used to obtain the indices.

Definition 1(i) has an intuitive interpretation [12, cf., Proposition 6.2]: positivity of marginal work $w^{\mathcal{S}}_{(a,i)}$ (where $\mathcal{S} \in \mathcal{F}$ and state $(a,i) \in \widetilde{\mathcal{I}}$ is controllable) is equivalent to monotonicity of total work,

$$g^{\mathcal{S}\setminus\{(a,i)\}}_{(a,i)} < g^{\mathcal{S}}_{(a,i)}, \qquad \text{if } (a,i) \in \mathcal{S},$$
$$g^{\mathcal{S}}_{(a,i)} < g^{\mathcal{S}\cup\{(a,i)\}}_{(a,i)}, \qquad \text{if } (a,i) \notin \mathcal{S}.$$

Informally stated, rejecting in a larger number of states corresponds to a larger expected total discounted number of rejected customers. Definition 1(i) is a natural assumption in many models, though, in general, it is neither a sufficient nor a necessary condition for indexability.

3.2 Postulated Active-Set Family

We use the results of [3], who characterized the optimal bi-threshold policies, and identify an active-set family $\mathcal{F}$ for which Theorem 1(i) holds. A bi-threshold active-set policy with open-gate threshold K_0 and closed-gate threshold K_1 will be denoted by

$$\widetilde{\mathcal{I}}_{K_0,K_1} := \{(0,K_0),(0,K_0+1),\ldots,(0,I)\} \cup \{(1,K_1),(1,K_1+1),\ldots,(1,I)\}, \qquad (13)$$

which is well-defined for all $0 \le K_0, K_1 \le I+1$ except the active sets $\widetilde{\mathcal{I}}_{I+1,I}$ and $\widetilde{\mathcal{I}}_{I,I+1}$, because states $(1,I)$ and $(0,I)$ are duplicates, and by definition either both or none of them can belong to $\widetilde{\mathcal{I}}_{K_0,K_1}$.

In words, active set $\widetilde{\mathcal{I}}_{K_0,K_1}$ prescribes to open or close the gate depending on the previous-epoch action and previous-epoch state. If the gate was open in the previous period, then we open the gate if and only if the queue length in the previous epoch was equal to or larger than the open-gate threshold K_0. Similarly, if the gate was closed in the previous period, then we open the gate if and only if the queue length in the previous epoch was equal to or larger than the closed-gate threshold K_1.

Intuitively, if an active set $\widetilde{\mathcal{I}}_{K_0,K_1}$ is optimal for some rejection cost ν, then $K_0 \le K_1$. Indeed, for a given previous-epoch queue length, we would be less prone to close the gate if it was closed than if it was open in the preceding period, because the queue length could not get larger under a closed gate, and therefore the rejection costs become relatively more harmful than the holding costs. On the other hand, it can be shown that $K_1 \le K_0+1$ (see below). Thus, the postulated family of optimal active sets for the admission control problem with delay is

$$\mathcal{F} := \{\widetilde{\mathcal{I}}_{K,K} : K = 0,1,\ldots,I+1\} \cup \{\widetilde{\mathcal{I}}_{K,K+1} : K = 0,1,\ldots,I-1\}. \qquad (14)$$

Theorem 2 ([3], Theorem 3.1). *If the holding cost c_i is nondecreasing and convex on $\mathcal{I}$, then $\mathcal{F}$ as defined in (14) contains an optimal active set for every rejection cost ν.*

Though the above result was shown for the problem with infinite buffer, it directly applies to the finite-buffer variant. Notice that if a bi-threshold policy is optimal for the infinite-buffer problem, then it is also optimal for all problems with buffer equal to or larger than both the thresholds. If the buffer is smaller than the larger optimal threshold (K_1), then it is optimal open the gate all the time.

For active-set family $\mathcal{F}$ given in (14), picking (a_k,i_k) becomes trivial, because there is only a unique feasible augmented state in each step. For instance, in step $k=1$, only state $(1,0)$ both belongs to $\widehat{\mathcal{S}}_0$ and $\widehat{\mathcal{S}}_0 \setminus \{(1,0)\} = \widetilde{\mathcal{I}}_{0,1} \in \mathcal{F}$, since $\widehat{\mathcal{S}}_0 := \widetilde{\mathcal{I}} = \widetilde{\mathcal{I}}_{0,0}$. Similarly, in step $k=2$, only state $(0,0)$ both belongs to $\widehat{\mathcal{S}}_1$ and $\widehat{\mathcal{S}}_1 \setminus \{(0,0)\} = \widetilde{\mathcal{I}}_{1,1} \in \mathcal{F}$. In general, $(a_k,i_k) = (0,(k/2)-1)$ for all even $1 \le k \le 2I$, and $(a_k,i_k) = (1,(k-1)/2)$ for all odd $1 \le k \le 2I$. Finally, in step $k = 2I+1$, the picked state is $(*,I)$.

To summarize, the sequence of candidate active sets $\{\widehat{\mathcal{S}}_k\}_{k=0}^{2I+1}$ in algorithm $AG_{\mathcal{F}}$ under active-set family $\mathcal{F}$ given in (14) is

$$\widehat{\mathcal{S}}_0 = \widetilde{\mathcal{I}} = \widetilde{\mathcal{I}}_{0,0}, \widehat{\mathcal{S}}_1 = \widetilde{\mathcal{I}}_{0,1}, \widehat{\mathcal{S}}_2 = \widetilde{\mathcal{I}}_{1,1}, \widehat{\mathcal{S}}_3 = \widetilde{\mathcal{I}}_{1,2}, \widehat{\mathcal{S}}_4 = \widetilde{\mathcal{I}}_{2,2}, \ldots$$
$$\ldots, \widehat{\mathcal{S}}_{2I-1} = \widetilde{\mathcal{I}}_{I-1,I}, \widehat{\mathcal{S}}_{2I} = \widetilde{\mathcal{I}}_{I,I}, \widehat{\mathcal{S}}_{2I+1} = \widetilde{\mathcal{I}}_{I+1,I+1} = \emptyset, \quad (15)$$

```
for K = 1 to I do
  ν̂_(1,K−1) := ν^{Ĩ_{K−1,K−1}}_(1,K−1) ;
  ν̂_(0,K−1) := ν^{Ĩ_{K−1,K}}_(0,K−1) ;
end {for};
ν̂_(*,I) := ν^{Ĩ_{I,I}}_(*,I) ;
{Output {ν̂_(a,i)}_(a,i)∈Ĩ}
```

Figure 3: Algorithmic scheme of $AG_{\mathcal{F}}$ under active-set family $\mathcal{F}$ given in (14).

and the sequence of picked states $\{(a_k, i_k)\}_{k=1}^{2I+1}$ is

$$(a_1, i_1) = (1, 0), (a_2, i_2) = (0, 0), (a_3, i_3) = (1, 1),$$
$$(a_4, i_4) = (0, 1), \ldots, (a_{2I-1}, i_{2I-1}) = (1, I-1),$$
$$(a_{2I}, i_{2I}) = (0, I-1), (a_{2I+1}, i_{2I+1}) = (*, I).$$

Given the above, in Figure 3 we present the reduction of the algorithmic scheme $AG_{\mathcal{F}}$ as it applies to the postulated family $\mathcal{F}$ given in (14). Notice that the computational complexity remains at the same level since the main difficulty lies in the calculation of $\nu_{(a_k,i_k)}^{\widehat{\mathcal{S}}_{k-1}}$, for which no computational details are given. Therefore we also call them algorithmic schemes, not algorithms. The goal of this paper is to establish the validity of $AG_{\mathcal{F}}$ for our problem and to develop its implementation of low computational complexity.

4. RESULTS

In this section we focus on the admission control problem with delay to a buffer (i.e., $I \geq 2$) under the discounted criterion. The case $I = 1$, referring to the admission control problem with delay to server with no dedicated buffer, is treated in subsection 4.2. The results under the time-average criterion are summarized in subsection 4.3.

Our main results are twofold. First, we prove the positivity of marginal works (cf. Definition 1(i)) for $\mathcal{F}$ given in (14), so that the algorithm $AG_{\mathcal{F}}$ can be applied to compute the indices. Second, we simplify $AG_{\mathcal{F}}$ obtaining a procedure that performs only a linear number of arithmetic operations to compute all the indices and the optimal thresholds.

Let us introduce a more compact notation. For any augmented-state-dependent variable $x_{(a,i)}$, we will use the backward difference operator in the first dimension, i.e., the *action-difference operator*,

$$\Delta_1 x_{(1,i)} := x_{(1,i)} - x_{(0,i)} \tag{16}$$

and in the second dimension, i.e., the *state-difference operator*,

$$\Delta_2 x_{(a,i)} := x_{(a,i)} - x_{(a,i-1)} \tag{17}$$

whenever the right-hand side expressions are defined. For definiteness, we further let $\Delta_2 x_{(a,0)} := 0$ for $a \in \mathcal{A}$. Directly from these definitions we obtain the following auxiliary identity,

$$\Delta_2 x_{(1,i)} - \Delta_2 x_{(0,i)} = \Delta_1 x_{(1,i)} - \Delta_1 x_{(1,i-1)}. \tag{18}$$

In the following we list our main results, drawing on the technical analysis of work measures which is omitted due to space restrictions.

Proposition 1.

(a) The marginal works in problem (9) are positive under the active-set family $\mathcal{F}$ given in (14), i.e., Definition 1(i) holds.

(b) If the holding cost c_i is nondecreasing and convex on $\mathcal{I}$, then the admission control problem with delay in (9) is PCL($\mathcal{F}$)-indexable, and therefore it is indexable and algorithm $AG_{\mathcal{F}}$ calculates the marginal productivity indices for this problem.

4.1 A Fast Algorithm for Calculation of All Marginal Productivity Indices

In order to avoid unnecessary technical complications, in this subsection we narrow our attention to the *linear holding costs* case, i.e., $c_i := c \cdot i$ for some positive constant c. We develop an algorithm for calculation of *all* marginal productivity indices in $\mathcal{O}(I)$, which is two orders of magnitude faster that the best general implementation of algorithm $AG_{\mathcal{F}}$ performing $\mathcal{O}(I^3)$ arithmetic operations.

The algorithmic scheme $AG_{\mathcal{F}}$ in Figure 3 is exhibited in its *bottom-up* version, as it calculates the marginal productivity indices in nondecreasing order (cf. Definition 1(ii')). This is closely related to our definition of indexability in section 3 as the property that "active sets $\mathcal{S}^*(\nu)$ diminish monotonically from $\widetilde{\mathcal{I}}$ to the empty set as the rejection cost ν increases from $-\infty$ to ∞," being emulated by the bottom-up version of the algorithm. Notice that we could equivalently define indexability as "active sets $\mathcal{S}^*(\nu)$ expand monotonically from the empty set to $\widetilde{\mathcal{I}}$ as the rejection cost ν decreases from ∞ to $-\infty$." This intuitively leads to consideration of algorithm $AG_{\mathcal{F}}$ in its equivalent, *top-down* version, starting with the empty set and calculating the indices in nonincreasing order.

In other words, while the bottom-up version of algorithm $AG_{\mathcal{F}}$ traverses the active-set family $\mathcal{F}$ in the order (cf. (15))

$$\widetilde{\mathcal{I}}_{0,0}, \widetilde{\mathcal{I}}_{0,1}, \widetilde{\mathcal{I}}_{1,1}, \widetilde{\mathcal{I}}_{1,2}, \ldots, \widetilde{\mathcal{I}}_{I-1,I}, \widetilde{\mathcal{I}}_{I,I}, \widetilde{\mathcal{I}}_{I+1,I+1},$$

the top-down version does that in the reverse order

$$\widetilde{\mathcal{I}}_{I+1,I+1}, \widetilde{\mathcal{I}}_{I,I}, \widetilde{\mathcal{I}}_{I-1,I}, \ldots, \widetilde{\mathcal{I}}_{1,2}, \widetilde{\mathcal{I}}_{1,1}, \widetilde{\mathcal{I}}_{0,1}, \widetilde{\mathcal{I}}_{0,0}.$$

For instance, index $\nu_{(1,0)}$ is calculated as the marginal productivity rate $\nu_{(1,0)}^{\widetilde{\mathcal{I}}_{0,0}}$ in the bottom-up version, while the same index is calculated as the marginal productivity rate $\nu_{(1,0)}^{\widetilde{\mathcal{I}}_{0,1}}$ in the top-down version. In fact, [12, Theorem 6.4(b)] implies that $\nu_{(a_k,i_k)}^{\widehat{\mathcal{S}}_{k-1}} = \nu_{(a_k,i_k)}^{\widehat{\mathcal{S}}_{k}}$, using the notation of Figure 3. Thus, since the active set of type $\widetilde{\mathcal{I}}_{K,K}$ is efficient every two steps of the algorithm (except for the last step, where $\widetilde{\mathcal{I}}_{I+1,I+1}$ follows $\widetilde{\mathcal{I}}_{I,I}$), we can formulate the indices in terms of marginal productivity rates under active sets $\widetilde{\mathcal{I}}_{K,K}$ only. Such an algorithmic scheme is presented in Figure 4.

Next we develop an efficient implementation of the algorithmic scheme $AG_{\mathcal{F}}$, which we present in Figure 5. The algorithm FA is two orders of magnitude faster than the best existing general implementation of the algorithm $AG_{\mathcal{F}}$. We characterize the marginal productivity indices calculated as indicated in Figure 4 in terms of closed-form expressions of pivot state-differences.

Proposition 2.

```
ν_(1,0) := ν_(1,0)^{Ĩ_{0,0}};
for K = 1 to I − 1 do
    ν_(0,K−1) := ν_(0,K−1)^{Ĩ_{K,K}};
    ν_(1,K) := ν_(1,K)^{Ĩ_{K,K}};
end {for};
ν_(0,I−1) := ν_(0,I−1)^{Ĩ_{I,I}};
ν_(*,I) := ν_(*,I)^{Ĩ_{I+1,I+1}};
{Output {ν_(a,i)}_{(a,i)∈Ĩ}}
```

Figure 4: Algorithmic scheme of calculation of marginal productivity indices for the admission control problem with delay in terms of active sets $\widetilde{\mathcal{I}}_{K,K}$ only.

(a) The algorithm FA in Figure 5 computes the marginal productivity indices for problem (9) under the discounted criterion.

(b) The algorithm FA in Figure 5 performs $\mathcal{O}(I)$ arithmetic operations.

Once the optimal index policy is known, the optimal thresholds for a given rejection cost ν can easily be obtained. The optimal open-gate threshold is

$$K_0 := \min\{i \in \mathcal{I} : \nu_{(0,i)} \geq \nu\}.$$

Similarly, the optimal closed-gate threshold is

$$K_1 := \min\{i \in \mathcal{I} : \nu_{(1,i)} \geq \nu\}.$$

If $\nu > \nu_{(*,I)}$, then $K_0 := I + 1$ and $K_1 := I + 1$.

Notice that the indices calculated in the algorithm FA's "Loop" are *independent* of the buffer length I (only the indices of states $(0, I-1)$ and $(*, I)$ depend on I). In other words, considering two buffers with lengths $I_1 < I_2$, the marginal productivity indices of states

$$(1,0), (0,0), (1,1), \ldots, (0, I_1 - 2), (1, I_1 - 1)$$

are the same for both buffers; the indices of states $(0, I_1 - 1)$ and $(*, I_1)$ would differ, while the remaining states only exist under buffer I_2. Therefore, the algorithm FA can be used to obtain the indices for infinite-length buffer. However, in such a case, "Loop" would never stop.

We present a simple algorithmic check (Figure 6) that can be run before "Loop" (and after "Initialization") to verify whether $K_0 = K_1 = \infty$, i.e., whether it is optimal to let the gate open always. It is because the indices are calculated in nondecreasing order and they converge as the buffer length $I \to \infty$.

LEMMA 1. *If the buffer length $I = \infty$, the marginal productivity indices calculated in "Loop" of algorithm FA under the discounted criterion in Figure 5 converge.*

If the algorithmic check does not confirm the infinite thresholds, the algorithm FA can be run, stopping the loop once an index greater than ν is found and omitting "Finalization" part.

4.2 Admission Control Problem with Delay to Server with no Dedicated Buffer

In this section we solve the admission control problem with delay for $I = 1$, i.e., no customer is allowed to be queued, except for the one in service. While this problem may not be of intrinsic interest, its solution given next will serve as a basis for the servers assignment problem with delay discussed in **??**. Considered is the linear holding cost case.

PROPOSITION 3. *The marginal productivity index of state $(a, i) \in \widetilde{\mathcal{I}}$ in case $I = 1$ is state-independent and equals*

$$\nu_{(a,i)} := \frac{\zeta\beta C}{\lambda} = \frac{c\lambda\beta(1-\mu)}{1-\beta(1-\mu)}.$$

These indices can be obtained in the same way as the general case $I \geq 2$. The state-independent marginal productivity index means that, given a rejection cost ν, it is optimal either to admit always, or to reject always, regardless of the previous-epoch state and previous-epoch action, i.e., regardless of information available.

4.3 Admission Control Problem with Delay under Time-Average Criterion

Our results extend directly to the admission control with delay under the time-average criterion.

PROPOSITION 4. *By setting $\beta := 1$, the algorithm FA in Figure 5 computes the marginal productivity indices for problem (9) under the time-average criterion.*

In case $I = \infty$, the algorithmic check in Figure 6 is only valid under $\beta < 1$, and therefore is not suitable for the time-average criterion. In fact, it is not necessary to perform such a check, because under the time-average criterion the indices diverge.

4.4 Further Remarks

If in state $(1, 0)$, the buffer is empty, because it was empty a period ago and the gate has been closed since then. Therefore, one could expect that the index of state $(1, 0)$ is the same as the index of state 0 in the no-delay problem, which is in fact true. Moreover, there is a simple interpretation of that expression.

If the buffer is empty, the expected total β-discounted holding cost is

$$\zeta\beta c\left[1 + \beta(1-\mu) + (\beta(1-\mu))^2 + \ldots\right] = \frac{\beta\zeta c}{1-\beta+\beta\mu},$$

because ζ is the probability that the customer remains in the buffer for more than a period. The above expression is equal to $\lambda\nu_{(1,0)}$, the expected (total β-discounted) rejection cost is if the rejection cost $\nu = \nu_{(1,0)}$. Thus, in state $(1, 0)$ it is optimal to close the gate if the expected rejection cost is lower than the expected discounted total holding cost of an admitted customer. Further, in state $(1, 0)$ it is optimal to let the gate open if the expected rejection cost is greater than the expected discounted total holding cost of an admitted customer. If the two expected costs are equal, both closing and opening is optimal. It is also clear that under the former condition it is optimal to close the gate in *any* state, and therefore the indices of all states must not be smaller than $\nu_{(1,0)}$.

```
{Input I, λ, μ, c, β}
{Inicialization}
```

$\zeta := \lambda(1-\mu);\quad \eta := \mu(1-\lambda);\quad \varepsilon := 1-\zeta-\eta;$

$A_0 := 0;\quad A_0' := \beta\zeta;\quad B := \beta\mu/(1-\beta+\beta\mu);\quad B' := \beta\zeta B + \beta(\mu-\eta);\quad C := c/(1-\beta+\beta\mu);\quad D_0 := 0;$

$\nu_{(1,0)} := \zeta\beta C/\lambda;$

$$\nu_{(0,0)} := \frac{\zeta}{\lambda}\beta C_1 \frac{(1-\beta+\beta\mu)(1+\beta\zeta)+\beta\zeta(\mu+\beta\mu+\beta\zeta)+(1-\beta+\beta\mu)(1+\beta)(\mu-\eta)}{(1-\beta+\beta\mu)(1+\beta\zeta)+\beta\zeta\beta\zeta(1-B)-\beta\zeta\beta(\mu-\eta)};$$

`{Loop}`

for $K = 1$ **to** $I-1$ **do**

$A_K := \beta\zeta/[1-\beta+\beta\zeta+\beta\eta(1-A_{K-1})];\quad A_K' := \beta\zeta+\beta(\mu-\eta)A_K;\quad D_K := (c+\beta\eta D_{K-1})\,A_K/(\beta\zeta);$

$$f^0 := -\frac{\frac{\beta\zeta}{A_K}D_K+\beta\zeta(c+\beta\mu B D_{K-1})+[c-\beta(\mu-\eta)\beta D_{K-1}]\,B'}{\frac{A_K'}{A_K}+\beta A_{K-1}'B'+\beta\zeta\beta\mu(1-BA_{K-1})};$$

$$f^1 := -\frac{\frac{\beta\zeta}{A_K}D_K+c\beta\zeta BA_{K-1}+[\beta\mu\beta\zeta+(1-\beta)\beta(\mu-\eta)]\,D_{K-1}+A_{K-1}'(c-\beta\zeta\beta C)}{\frac{A_K'}{A_K}+\beta A_{K-1}'B'+\beta\zeta\beta\mu(1-BA_{K-1})};$$

$$g^0 := \frac{\beta\lambda\,(1+B')}{\frac{A_K'}{A_K}+\beta A_{K-1}'B'+\beta\zeta\beta\mu(1-BA_{K-1})};\quad g^1 := \frac{1+A_{K-1}'}{1+B'}g^0;$$

if $K > 1$ **then**

$$\nu_{(0,K-1)} := \frac{[\beta(\mu-\eta)(D_{K-1}-c)+\beta\eta\beta\zeta D_{K-1}+\beta\zeta\beta C]-[\beta\eta A_{K-1}A_{K-2}'+\beta\varepsilon A_{K-1}']\,f^0-\beta\zeta B'f^1}{\beta\lambda-\left[\beta\eta A_{K-1}A_{K-2}'+\beta\varepsilon A_{K-1}'\right]g^0-\beta\zeta B'g^1};$$

end {if};

$$\nu_{(1,K)} := \frac{[\beta(1-\mu)\beta\zeta C+\beta\mu\beta(\mu-\eta)D_{K-1}]-\beta\mu A_{K-1}'f^0-\beta(1-\mu)B'f^1}{\beta\lambda-\beta\mu A_{K-1}'g^0-\beta(1-\mu)B'g^1};$$

end {for};

`{Finalization}`

$A_I := \beta\zeta/[1-\beta+\beta\zeta+\beta\eta(1-A_{I-1})];\quad A_I' := \beta\zeta+\beta(\mu-\eta)A_I;\quad D_I := (c+\beta\eta D_{I-1})\,A_I/(\beta\zeta);$

$$f^0 := -\frac{\frac{\beta\zeta}{A_I}D_I-\beta(\mu-\eta)\beta\mu D_{I-1}}{\frac{A_I'}{A_I}+\beta\mu A_{I-1}'};\quad g^0 := \frac{\lambda(1+\beta\mu)}{\frac{A_I'}{A_I}+\beta\mu A_{I-1}'};$$

$$\nu_{(0,I-1)} := \frac{[\beta(\mu-\eta)(D_{I-1}-c)+\beta\eta\beta\zeta D_{I-1}]-[\beta\eta A_{I-1}A_{I-2}'+\beta\varepsilon A_{I-1}']\,f^0}{\beta(1-\zeta)\lambda-\left[\beta\eta A_{I-1}A_{I-2}'+\beta\varepsilon A_{I-1}'\right]g^0};$$

$$\nu_{(*,I)} := \frac{\beta(\mu-\eta)D_{I-1}A_I'+\beta\zeta D_I A_{I-1}'}{\lambda A_I'-\lambda A_I A_{I-1}'};$$

{Output $\left\{\nu_{(a,i)}\right\}_{(a,i)\in\widetilde{\mathcal{I}}}$}

Figure 5: Fast algorithm FA for calculation of an optimal index policy.

$A := \left[1-\beta+\beta\zeta+\beta\eta-\sqrt{(1-\beta+\beta\zeta+\beta\eta)^2-4\beta\eta\beta\zeta}\right]/(2\beta\eta);\quad A' := \beta\zeta+\beta(\mu-\eta)A;\quad D := cA/(\beta\zeta-\beta\eta A);$

$$f^0 := -\frac{\frac{\beta\zeta}{A}D+\beta\zeta(c+\beta\mu BD)+[c-\beta(\mu-\eta)\beta D]\,B'}{\frac{A'}{A}+\beta A'B'+\beta\zeta\beta\mu(1-BA)};$$

$$f^1 := -\frac{\frac{\beta\zeta}{A}D+c\beta\zeta BA+[\beta\mu\beta\zeta+(1-\beta)\beta(\mu-\eta)]\,D+(c-\beta\zeta\beta C)A'}{\frac{A'}{A}+\beta A'B'+\beta\zeta\beta\mu(1-BA)};$$

$$g^0 := \frac{\beta\lambda\,(1+B')}{\frac{A'}{A}+\beta A'B'+\beta\zeta\beta\mu(1-BA)};\quad g^1 := \frac{1+A'}{1+B'}g^0;$$

$$\nu_{(1,\infty)} := \frac{[\beta(1-\mu)\beta\zeta C+\beta\mu\beta(\mu-\eta)D]-\beta\mu A'f^0-\beta(1-\mu)B'f^1}{\beta\lambda-\beta\mu A'g^0-\beta(1-\mu)B'g^1};$$

if $\nu \geq \nu_{(1,\infty)}$ **then** $K_0 := \infty;\quad K_1 := \infty;$ **end** {if};

Figure 6: Algorithmic check for the problem with infinite-length buffer.

Figure 7 shows the indices for a number of instances of the admission control problem with delay. An extensive simulation study we have performed suggests a convergence of the indices:

$$\nu_{(1,i)} \to \nu_{(0,i)} \qquad \text{as } \lambda \to 0,$$
$$\nu_{(1,i)} \to \nu_{(0,i-1)} \qquad \text{as } \zeta \to 1,$$
$$\nu_{(0,i)} \to \frac{\beta c}{1-\beta} \qquad \text{as } i \to \infty,$$
$$\nu_{(1,i)} \to \frac{\beta c}{1-\beta} \qquad \text{as } i \to \infty.$$

The convergence of the marginal productivity indices to $\beta c/(1-\beta)$ is intuitive. If the buffer is almost full (say, the pervious-epoch queue length is $I-2$), then admitting a customer means to increase the overal holding cost by c at least in the following $I-2$ periods, because the admitted customer cannot leave the system earlier that the previous $I-2$ customers. Therefore, the expected total β-discounted holding cost is at least

$$\beta c \left[1 + \beta + \beta^2 + \cdots + \beta^{I-2}\right] = \frac{\beta c(1-\beta^{I-1})}{1-\beta}.$$

On the other hand, it is not greater than the expected holding cost of remaining in the buffer forever, which is

$$\beta c \left[1 + \beta + \beta^2 + \ldots\right] = \frac{\beta c}{1-\beta}.$$

Now it is clear that the marginal productivity indices converge to $\beta c/(1-\beta)$ as $I \to \infty$.

5. ACKNOWLEDGMENTS

The authors' work has been supported in part by the Spanish Ministry of Education and Science under grant MTM2007-63140, by the European Union's Network of Excellence EuroFGI, and by the Autonomous Community of Madrid under grant CCG07-UC3M/ESP-3389. The work of P. Jacko has been further supported by the Spanish Ministry of Education and Science under grant MTM2004-02334 and an associated Postgraduate Research Fellowship funded by the European Social Fund.

6. REFERENCES

[1] S. Adlakha, R. Madan, S. Lall, and A. Goldsmith. Optimal control of distributed Markov decision processes with network delays. In *Proceedings of the 46th IEEE Conference on Decision and Control*, pages 3308–3314, New Orleans, LA, USA, December 2007.

[2] E. Altman, T. Başar, and R. Srikant. Flow control in communication networks with multiple users and action delays: A team-theoretic approach. In *Proceedings of the 36th IEEE Conference on Decision and Control*, pages 2387–2392, San Diego, California, December 1997.

[3] E. Altman and P. Nain. Closed-loop control with delayed information. *Performance Evaluation Review*, 20(1):193–204, 1992.

[4] D. Artiges. Optimal routing into two heterogeneous service stations with delayed information. *IEEE Transactions on Automatic Control*, 40(7):1234–1236, 1995.

[5] A. W. Berger. Comparison of call gapping and percent blocking for overload control in distributed switching systems and telecommunications networks. *IEEE Transactions of Communications*, 39(4):574–580, 1991.

[6] D. M. Brooks and C. T. Leondes. Markov decision processes with state-information lag. *Operations Research*, 20(4):904–907, 1972.

[7] C. Derman, G. J. Lieberman, and S. M. Ross. On the optimal assignment of servers and a repairman. *Journal of Applied Probability*, 17(2):577–581, 1980.

[8] K. V. Katsikopoulos and S. E. Engelbrecht. Markov decision processes with delays and asynchronous cost collection. *IEEE Transactions on Automatic Control*, 48(4):568–574, April 2003.

[9] J. Kuri and A. Kumar. Optimal control of arrivals to queues with delayed queue length information. *IEEE Transactions on Automatic Control*, 40(8):1444–1450, 1995.

[10] J. Kuri and A. Kumar. On the optimal control of arrivals to a single queue with arbitrary feedback delay. *Queueing Systems*, 27:1–16, 1997.

[11] J. Niño-Mora. Restless bandits, partial conservation laws and indexability. *Advances in Applied Probability*, 33(1):76–98, 2001.

[12] J. Niño-Mora. Dynamic allocation indices for restless projects and queueing admission control: A polyhedral approach. *Mathematical Programming, Series A*, 93(3):361–413, 2002.

[13] J. Niño-Mora. Dynamic priority allocation via restless bandit marginal productivity indices. *TOP*, 15(2):161–198, 2007.

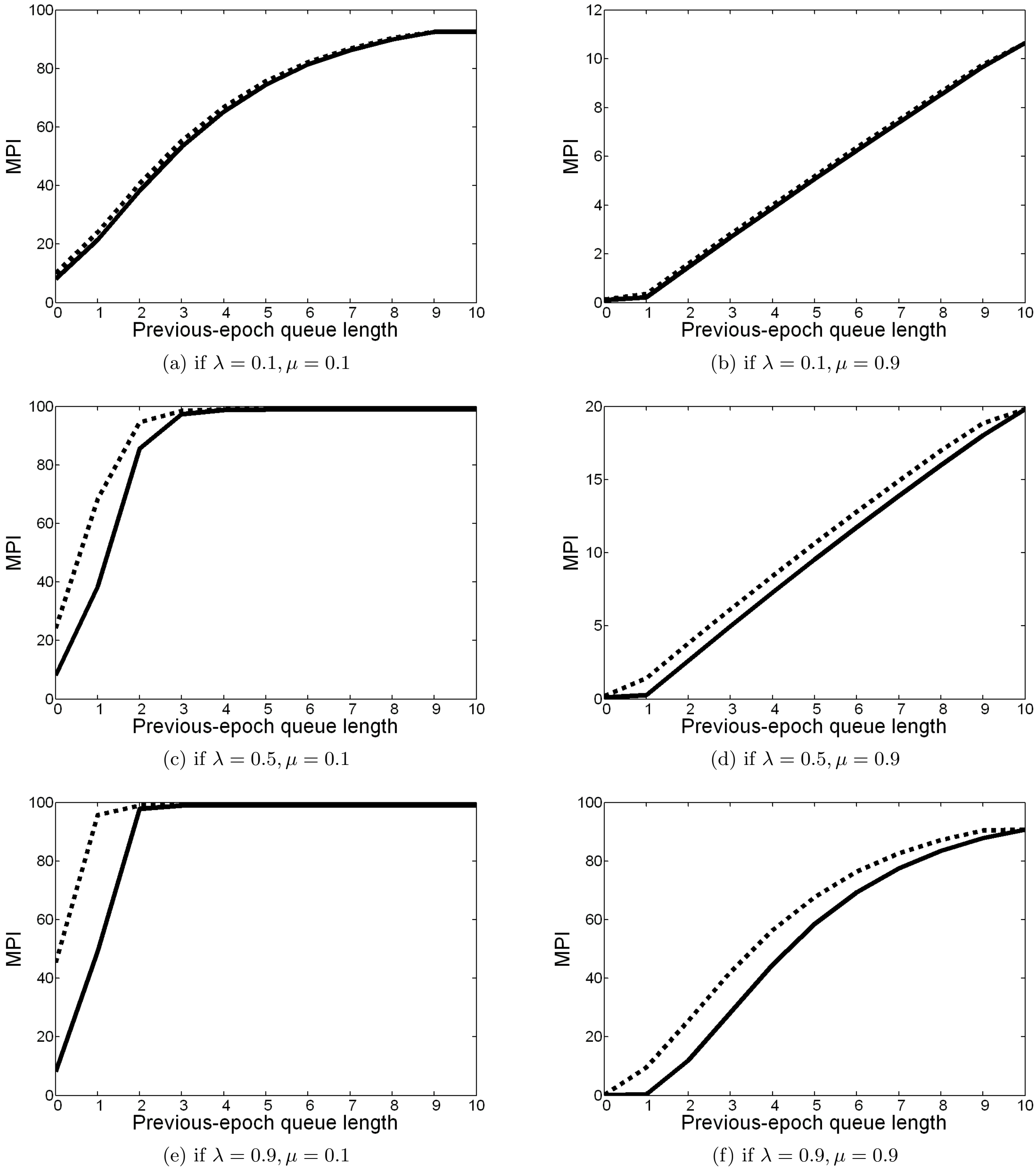
[14] J. Niño-Mora. Marginal productivity index policies for admission control and routing to parallel multi-server loss queues with reneging. *Lecture Notes in Computer Science*, 4465:138–149, 2007.

[15] J. Niño-Mora. Marginal productivity index policies for scheduling multiclass delay/loss-sensitive traffic with delayed state observation. In *Proceedings of the 3rd EuroNGI Conference on Next Generation Internet Networks - Design and Engineering for Heterogeneity*, pages 209–217, 2007.

[16] J. F. Shortle. Dynamic call-blocking algorithms for telecommunications networks. *IEEE Transactions of Communications*, 51(5):741–747, 2003.

[17] T. J. Walsh, A. Nouri, L. Li, and M. L. Littman. Planning and learning in environments with delayed feedback. In *Proceedings of the 18th European Conference on Machine Learning*, volume 4701 of *Lecture Notes in Artificial Intelligence*, pages 442–453, Warsaw, Poland, 2007.

[18] P. Whittle. Restless bandits: Activity allocation in a changing world. *A Celebration of Applied Probability, J. Gani (Ed.), Journal of Applied Probability*, 25A:287–298, 1988.



(a) if $\lambda = 0.1, \mu = 0.1$

(b) if $\lambda = 0.1, \mu = 0.9$

(c) if $\lambda = 0.5, \mu = 0.1$

(d) if $\lambda = 0.5, \mu = 0.9$

(e) if $\lambda = 0.9, \mu = 0.1$

(f) if $\lambda = 0.9, \mu = 0.9$

Figure 7: Optimal marginal productivity indices (MPI) for the admission control problem with delay with parameters $I = 10, c = 1, \beta = 0.99$. The solid line exhibits indices $\nu_{(1,i)}$ and the dotted line exhibits indices $\nu_{(0,i)}$.