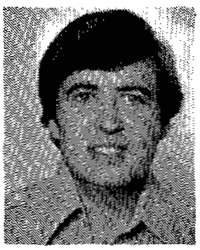
[16] K. R. Pattipati and D. L. Kleinman, "Application of dynamic programming to priority assignment in a class of queueing systems," Univ. Connecticut, Storrs, CT, Tech. Rep., EECS-TR-80-5, Mar. 1980.

Krishna R. Pattipati was born in Chayapuram, A. P., India. He received the B. Tech degree in electrical engineering from the Indian Institute of Technology, Kharagpur, in 1975, and the M.S. and Ph.D. degrees in systems engineering from the University of Connecticut, Storrs in 1977 and 1980, respectively.

He worked as a research assistant at the University of Connecticut from 1975 to 1980, where he worked on problems of human operator modeling and optimal system design. Since November 1980, he has been a member of the technical staff of Alphatech, Inc., Burlington, MA. His current research interests lie in the areas of human decision modeling, large scale systems, queueing networks and numerical optimization.



David L. Kleinman (S'62–M'68) was born in Brooklyn, NY, on January 4, 1942. He received the B.E.E. degree from the Cooper Union for the Advancement of Science and Art, New York, NY, in 1962, and the S.M. and Ph.D. degrees, both in electrical engineering from the Massachusetts Institute of Technology, Cambridge, in 1963 and 1967, respectively.

From 1963 to 1967 he worked as a Research Assistant in the M.I.T. Electronic Systems Laboratory. In 1967 he joined the staff of Bolt, Beranek and Newman, Inc., Cambridge, MA, where he pioneered in the application of modern control theory to study problems in manual control and man-machine systems. He also worked on the application of differential game theory to study pursuit-evasion problems. In 1971 he joined Systems Control, Inc., to establish and direct its Cambridge office. He was responsible for leading applied research projects in both manual and automatic control system analysis. In one effort, modern control theory was used to design the automatic control system for the Trident submarine. In September 1973, he joined the Electrical Engineering and Computer Science Department at the University of Connecticut, Storrs; he became a full Professor in September 1978. He is currently head of the Department's Systems Group and director of the Cyberlab—Laboratory for Man-Machine and Systems Research. He is also a co-founder of Alphatech, Inc., Burlington, MA. His current fields of interest include the application of modern control and decision theory to man-machine systems analysis, digital control methods, and computational algorithm development.

Dr. Kleinman is a member of Tau Beta Pi, Eta Kappa Nu, and Sigma Xi. From 1972–1975 he was a member of the Administrative Committee of the IEEE Control Systems Society, and an Associate Editor of the IEEE TRANSACTIONS ON AUTOMATIC CONTROL.

Stochastic Control of Two Partially Observed Competing Queues

JOHN S. BARAS, MEMBER, IEEE, AND ARTHUR J. DORSEY, STUDENT MEMBER, IEEE

Abstract—**We consider the dynamic control of two queues competing for the services of one server. The problem is to design a server time allocation strategy, when the sizes of the queues are not observable. The performance criterion used is total expected aggregate delay. The server is assumed to observe arrivals but not departures.**

This problem is formulated as a stochastic optimal control problem with partial observations. The framework we adopt is that of stochastic control in discrete time and countable "state space." The observations are modeled as discrete time, 0–1 point processes with rates that are influenced by a Markov chain. Examples from computer control of urban traffic are given, to illustrate the practical motivation behind the present work, and to relate to earlier work by us on the subject. A particular feature of the formulation is that the observations are influenced by transitions of the state of the Markov chain. The classical tools of simple Bayes rule and dynamic programming suffice for the analysis. In particular, we show that the "one step" predicted density for the state of the Markov chain, given the point process observations is a sufficient statistic for control.

This framework is then applied to the specific problem of two queues competing for the services of one server. We obtain explicit solutions for the finite time expected aggregate delay problem. The implications of these results for practical applications as well as implementation aspects of the resulting optimal control laws are discussed.

Manuscript received April 13, 1981. This work was supported in part by the U.S. Department of Transportation under Contract DOT-OS-60134 at the University of Maryland; in part by National Science Foundation under Grant ENG-79-03879 while J. S. Baras was visiting the Electronics Research Laboratory, University of California, Berkeley; and in part by the Minta Martin Fellowship awarded to A. J. Dorsey at the University of Maryland.

The authors are with the Department of Electrical Engineering, University of Maryland, College Park, MD 20742.

I. INTRODUCTION

OPTIMAL control strategies for queuing systems have become exceedingly important in recent years, due to potential applications of analytical studies in performance evaluation and design of computer and communication networks, computer systems for the control of urban traffic, computer systems for the control of inventories or networks of inventories.

Extensive bibliographics and reviews of queuing control models and strategies can be found in Crabil *et al.* [1], Sobel [2], and Stidham and Prabhu [3]. Typically classical

0018-9286/81/1000-1106$00.75

queuing theory methods treat static or steady-state models and strategies. However, several of the intented applications require dynamic (or transient) analysis and control. Furthermore, in earlier studies the queue sizes are assumed observable (i.e., known). Recently, examples of dynamic control of queuing systems based on point process models and their relevant theory have been given in studies by Walrand [4], Brémaud [5], Rosberg *et al.* [6]. Finally, progress towards the development of optimal control strategies in queuing systems requires analysis of systems involving several queues competing for the services of server stations.

This paper analyzes a simple problem involving two competing queues which exhibits some of the desired modeling characteristics mentioned earlier. The problem of selecting which of two parallel queues to serve with a single server is considered. We adopt a discrete time formulation with the arrival rate at each queueing station allowed to depend on the queue size and control value. The service rate is also allowed to depend on queue size and the control value. The control to be selected is the decision at each time slot to serve one of the two queues. The controller observes the arrivals of the two queues but the queue sizes are unobservable, i.e., departures are not observed. The control is to be selected as a function of the past histories of the observed arrival and control processes. The instantaneous cost is linear in the waiting times of the two queues. We thus have a stochastic control problem involving partially observed queues.

Our motivation comes from examples arising in the computer control of urban traffic and in satellite communication problems. We have successfully employed such queueing models earlier [7] in developing estimation and prediction algorithms for queues in urban traffic, based on partial observations. The present paper represents a continuation of [7] by utilizing these estimates to obtain an optimal control strategy for a single traffic intersection. The framework in which the problem is formulated is that of controlling a partially observed Markov process [13], [17], [19]. In particular the results of Smallwood and Sondik [19] hold. It is well known that simple Bayes rule combined with dynamic programming provide an elementary methodology for such problems [13], [17], [19]. It follows that an elementary framework for the analysis of dynamic queueing control problems in discrete time is to consider them as stochastic control problems of Markov chains with 0–1 point process observations. The starting point of this formalism is the (assumed known) joint statistics of the observed point process and of the transitions of the Markov chain. An identical framework in continuous time has been utilized by Segall [10] (using Martingale techniques) in analyzing dynamic file assignment in computer networks. Dependence of arrival and departure rates on queue size has been considered in queueing networks studies by Jackson [8].

Only the finite horizon, average aggregate delay cost problem is considered in the paper. The formalism and methodology we use are applicable to more complex queueing control problems. The tools are standard [19]. Slight modifications are necessary due to the special relationship between observations and state transitions that appear in queueing models. We have followed in the brief presentation of the existing methodology the same sequence as in Varaiya [9]. The contribution of the paper is in the application of these techniques to dynamic queuing control problems. It is established that the optimal server allocation strategy depends on the observed arrivals through a sufficient statistic which can be computed via the filter-predictor equations derived in [7] and in the present paper. This statistic is the "one step" predicted probability vectors for the two queues based on the observed arrivals. This has the form of a separation theorem. Consequently, all the necessary computations needed to implement the optimal policy can be performed off-line. Due to the linearity of the instantaneous cost we show that these computations are quite simple. Furthermore, we discuss how the results of these off-line computations should be stored in order to facilitate the on-line implementation of the strategy. The optimal average delay is shown to be piecewise linear in the sufficient statistic. Finally, the present paper may serve as a basis for further analysis of the two competing queues problem: evaluation of suboptimal policies, alternative performance objectives such as average delay per unit time or infinite horizon discounted aggregate delay, adaptive control.

The paper is organized as follows. In Section II, we formulate a simple two competing queues problem. Extensions to more general models are then presented. The basic questions to be analyzed are formulated. In Section III, general filtering and prediction results for the class of stochastic systems with observations influencing "state transitions," are reviewed briefly and are then applied to queueing models. In Section IV, we review briefly general dynamic programming theorems for partially observed stochastic control problems for the same class. Particular results for the two competing queues problem are then obtained. In Section V, we present computations and evaluations of the strategies obtained, as the theory is applied to a simple problem.

II. The Two Competing Queues Problem

Consider the problem of selecting which of the two parallel queues to serve with a single server. The system is depicted in Fig. 1. The time is divided into uniform time slots; that is we adopt a discrete time formulation. Customers arrive into stations one and two according to two independent Bernoulli streams with constant rates λ^1, λ^2, respectively. If we let $n_a^1(t), n_a^2(t)$ denote the two arrival processes, with t discrete, parameterizing the time slots,

$$\lambda^i = \Pr\left\{n_a^i(t) = 1\right\}, \qquad i = 1,2. \tag{2.1}$$

The two queues compete for the services of a server whose service completions follow a Bernoulli stream with constant rate μ. If we let $n_d(t)$ denote the service process, whenever the server is connected to one of the two queues (when it is nonempty)

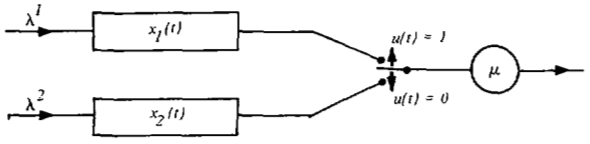


Fig. 1. The server time allocation problem.

$$\mu=\Pr\{n_d(t)=1\}. \tag{2.2}$$

Let $x_i(t)$ be the number of customers in queue i during time slot t, the customer in service included. The control to be selected is clearly of a switching type. When $u(t)=1$ and the server completes a service, the next customer to be served comes from queue 1, while if $u(t)=0$ the next customer comes from queue 2. This is a simple priority assignment (sequencing) problem in a two class queuing system.

The server time allocation is to be selected in order to minimize delays, weighted according to c_1, c_2, two positive constants. Thus, the cost per unit time with queues $x_1(t), x_2(t)$ is $c_1x_1(t)+c_2x_2(t)$. For a finite horizon of length T we wish to minimize the *average aggregate delay*

$$J=E\left\{\sum_{t=0}^{T}\left(c_1x_1(t)+c_2x_2(t)\right)\right\}. \tag{2.3}$$

The queue sizes are not observable, since only the arrival processes $\{n_a^i(t),\ i=1,2\}$ are observable to the controller. The server time allocation strategy minimizing (2.3) is allowed to be a function of the past histories of the arrival processes and the past history of the control.

This problem is the simplest in a sequence of problems that appear in urban traffic control. The controller in the traffic problem is the traffic light, the two queues correspond to the two approaches at a traffic intersection, the arrival processes are the observations from upstreet loop detectors. Therefore, the problem we described above models in a simple way the "critical intersection" traffic control problem. Traffic activated control laws lead to the dependence of the control value $u(t)$ on past histories of arrivals and control processes. We refer the interested reader to [7] for further details on the development of queue models appropriate for urban traffic problems.

Similar problems appear in computer networks, where one allocates files according to demand [10] or in satellite communication networks [11] where one controls retransmission laws according to network traffic load. We shall not give any further details on these problems here and refer the reader to [10], [11] instead.

The problem is easier to analyze when the two queues can grow without bound. In the main body of the paper, we obtain results that can be applied to more general queueing models, than the one described in the beginning of this section. In particular, to represent effects of congestion, we can let the arrival and service rates depend on the queue size. Thus, we shall consider problems where the following assumptions hold:

$$\Pr\left\{n_a^i(t)=1\,\middle|\,\begin{matrix}\text{past histories of } x_1, x_2, n_a^1, n_a^2,\\ \text{and } u, \text{ up to time } t\end{matrix}\right\}$$
$$=\Pr\{n_a^i(t)=1|x_i(t)=k, u(t)=v\}$$
$$=:\lambda^i(t,k,v),\qquad i=1,2, \tag{2.4}$$

$$\Pr\left\{n_d^i(t)=1\,\middle|\,\begin{matrix}\text{past histories of } x_1, x_2, n_a^1, n_a^2\\ \text{and } u, \text{ up to time } t\end{matrix}\right\}$$
$$=\Pr\{n_d(t)=1|x_i(t)=k, u(t)=v\}$$
$$=:\mu^i(t,k,v),\qquad i=1,2. \tag{2.5}$$

Since there are no departures when a queue is empty, we must have

$$\mu^i(t,0,v)=0,\qquad i=1,2. \tag{2.6}$$

For the simple problem described in Fig. 1, $\lambda^i(t,k,v)=\lambda^i$, $i=1,2$ for all t,k,v, while

$$\mu^1(t,k,v)=\mu v,\qquad k\neq 0$$
$$\mu^2(t,k,v)=\mu(1-v),\qquad k\neq 0. \tag{2.7}$$

Most often in applications the queues are bounded in size. If we let N_i, $i=1,2$ be the maximum queue size for each queue then in addition to (2.6) we must have

$$\lambda^i(t,N_i,v)=0,\qquad i=1,2. \tag{2.8}$$

For the simple problem of Fig. 1, (2.8) will imply that

$$\lambda^i(t,k,v)=\begin{cases}\lambda^i, & i=1,2,\ k\neq 0, \text{ all } t,v\\ 0, & i=1,2,\ k=N_i, \text{ all } t,v.\end{cases} \tag{2.9}$$

In queueing models such as the preceding, there is a strong link between the transitions of the queue process and the observed arrival process. Indeed the occurence of an arrival implies that the queue will increase or remain the same for the next time slot. That is the observations imply certain "state transitions" for the underlying queue. Thus, the appropriate way to characterize a descriptive queue model, like the ones discussed here, is by means of the joint statistics of queue transitions and observations:

$$S_{ij}(t,v,\psi):=\Pr\{x(t+1)=j,\ y(t)=\psi|x(t)=i,$$
$$\cdot u(t)=v\},\qquad i,j=0,1,\cdots. \tag{2.10}$$

This will be the starting point of the development of filtering and prediction in Section III.

Finally in modeling the admissible control strategies, one may wish to allow the control during the t time slot to depend on the observations during the t time slot or not. This is clearly a modeling question. Realistically, it is better to allow $u(t)$ to depend on the observations only up to time $t-1$. This is the case considered in the paper. However, in some applications it may be important to know the tradeoff in complexity and performance, when

$u(t)$ is allowed to depend on observations up to time t. A simple relabeling[1] of the observation sequence reduces this case to the one studied here. Therefore, this case will not be analyzed. We refer the interested reader to [21] for explicit results in this case.

It is important to note, that the finite horizon problem discussed here plays a central role in the analysis of other criteria, such as average delay per unit time or discounted aggregate delay.

III. Filtering and Prediction

A. Formulation and Methodology

In this section, we review briefly filtering and prediction formulas for the state of a partially observed controlled Markov chain that influences the observations. Since we are working in discrete time, these results can be derived in an elementary fashion using simple Bayes rule [19]. We then apply these general results to the case of primary interest to us: discrete time 0–1 point process observations influenced by the state *transistions* of a Markov chain. In the case of queues evolving without bound the Markov chain has countable state space, while in the case of bounded queues the Markov chain has finite state space.

Let us assume that we are given the joint statistics of output (observations) and state transitions:

$$S_{ij}(t,v,\psi):=\Pr\{x(t+1)=j, \cdot y(t)=\psi|x(t)=i, u(t)=v\}, \qquad i,j\in I. \quad (3.1)$$

The inclusion of controls in the description of these statistics is meant to emphasize that the filtering and prediction formulas derived here will be used in later sections for the solution of stochastic control problems with partial observations. For our purposes it suffices to assume that the output process takes values in a finite set $\mathcal{Y}$. For queuing systems, as discussed in Section II, $\mathcal{Y}$ will be the set $\{0,1\}$. We further assume that the controls take values in a finite set $\mathcal{U}$. Again for queueing systems with switching controls, $\mathcal{U}$ will be just the set $\{0,1\}$ for one controller, or $\{0,1\}^m$ (mth-fold Cartesian product) in the case of m controllers. The state set I in (3.1) will be finite $I=\{0,1,\cdots,N\}$, as in the case of bounded queues, or countable $I=\{0,1,2,\cdots\}$, as in the case of unbounded queues.

For the queueing problems of interest to us (see Section II) the fundamental modeling assumption is that the joint statistics of observations and state transitions are influenced only by the current state and current control values. More precisely letting x^k, y^{k-1}, u^k denote the past histories of the state, observation and control processes respectively, we assume that the following "semi-Markovian" assumption holds.

[1]We thank the anonymous reviewer for making this observation which helped in the simplification of the presentation.

Assumption SM:

$$\Pr\{x(t+1)=j, y(t)=\psi|x^t, y^{t-1}, u^t\} = \Pr\{x(t+1)=j, y(t)=\psi|x(t)=i, u(t)=v\}. \quad (3.2)$$

This assumption is consistent with a "stochastic dynamical system" model of queues, which as discussed in [7, p. 13, eq. (2.11)] is valid under very general circumstances. It is easy to see that on the basis of (3.2) one can describe the partially observed queue as a probabilistic automaton (see [12]).

The available information to the controller for inference purposes at time k is denoted by

$$z^k:=(y^{k-1}, u^{k-1}) \quad (3.3)$$

where as usual

$$y^{k-1}:=(y(0), y(1),\cdots, y(k-1))$$
$$u^{k-1}:=(u(0), u(1),\cdots, u(k-1)). \quad (3.4)$$

We shall denote by Γ the set of admissible control policies, whereby each $\gamma\in\Gamma$ has the form

$$\gamma:=(g_0, g_1,\cdots, g_k,\cdots) \quad (3.5)$$

where each g_k is a function

$$g_k: \mathcal{Y}^k\times\mathcal{U}^k\to\mathcal{U}, \qquad z^k\mapsto u(k)=g_k(z^k). \quad (3.6)$$

We shall call the policies in Γ *strictly nonanticipative*, following standard terminology. Following standard usage we shall call z^k the *information vector available at time k*.

From the given statistics (3.1), we can obviously obtain the matrix of transition probabilities of the underlying Markov chain.

$$P_{ij}(t,v):=\Pr\{x(t+1)=j|x(t)=i, u(t)=v\} = \sum_{\psi\in\mathcal{Y}} S_{ij}(t,v,\psi). \quad (3.7)$$

Similarly, the output statistics, given the state, are given by

$$\lambda_i(\psi,t,v):=\Pr\{y(t)=\psi|x(t)=i, u(t)=v\} = \sum_{j\in I} S_{ij}(t,v,\psi). \quad (3.8)$$

To simplify later computations we introduce the following matrix:

$$M_{ij}(t,v,\psi):=\begin{cases}\dfrac{S_{ij}(t,v,\psi)}{\sum_{j\in I} S_{ij}(t,v,\psi)}, & \text{if } \sum_{j\in I} S_{ij}(t,v,\psi)>0\\ \delta_{ij}, & \text{otherwise}\end{cases} \quad (3.9)$$

where $i, j\in I$.

Given a control policy $\gamma \in \Gamma$ the conditional probabilities of interest in filtering and prediction are $p^{\gamma}_{k+1|k}(x(k+1)|y^k, u^k)$ and $p^{\gamma}_{k|k}(x(k)|y^k, u^k)$.

To simplify notation further, we introduce the (possibly infinite dimensional) row vectors of probabilities:

$$\Pi^{\gamma}_{k+1|k}(i) := p^{\gamma}_{k+1|k}(i|y^k, u^k), \qquad i \in I$$
$$\Pi^{\gamma}_{k|k}(i) := p^{\gamma}_{k|k}(i|y^k, u^k), \qquad i \in I. \tag{3.10}$$

It is a simple consequence of Bayes rule that with this notation

$$\Pi^{\gamma}_{k+1|k} = \Pi^{\gamma}_{k|k} M(k, u(k), y(k)) \tag{3.11}$$

it is convenient to introduce the diagonal matrix

$$D(k, \psi) = \text{diag}\,\{\lambda_i(\psi, k)\}, \qquad i \in I \tag{3.12}$$

and the column vector

$$e = [1, 1, 1, \cdots, 1, \cdots]^T. \tag{3.13}$$

Again by Bayes rule it is easy to establish that

$$\Pi^{\gamma}_{k+1|k+1} = \frac{\Pi^{\gamma}_{k+1|k} D(k+1, y(k+1))}{\Pi^{\gamma}_{k+1|k} D(k+1, y(k+1)) e}. \tag{3.14}$$

The computations (3.11)–(3.14) are slight modifications of existing results; see, in particular, [19]. Indeed $S_{ij}(t, v, \psi)$ in (3.1) would be written as $p_{ij}(v) r^v_{j\psi}$ in the notation of [19]. Although this product form for S was considered in [19], the arguments of [19] do not make use of this fact.[2]

We follow [9] for the remainder of this brief review. It is easy to establish that the conditional probability vectors $\Pi^{\gamma}_{k+1|k+1}, \Pi^{\gamma}_{k+1|k}$ do not depend on $\gamma \in \Gamma$ (i.e., they depend only on the values of the control not the control policy). They are computed recursively by (3.11) and (3.14). The initial value for the recursion is

$$\Pi_{0|0} = \frac{\Pi_0 D(0, y(0))}{\Pi_0 D(0, y(0)) e}. \tag{3.15}$$

As a consequence of this fact we shall drop the superscript from (3.10), (3.11), and (3.14) for the remainder of this paper. Following [9], [13] we consider information states. Recall that if z^k denotes the information vector available at time k, then a vector $\Phi(k)$ is called an *information state at time k* for the controlled stochastic system described by (3.1) if

i) $\Phi(k)$ can be evaluated from z^k;

ii) there is a function T_k such that

$$\Phi(k+1) = T_k(\Phi(k), (y(k), u(k))) \tag{3.16}$$

where $(y(k), u(k))$ is the new information obtained at time $k+1$.

We can now state the following theorem, which is a slight modification of well-known results [9], [13], [17], [19].

Theorem 3.1: $\Pi_{k|k-1}$ is an information state at time k for (3.1), (3.2).

We note in passing that (3.11), (3.14) suggest the difference equations for the "unnormalized" probability vectors

$$\rho_{k+1|k+1} = \rho_{k|k} M(k, u(k), y(k)) D(k+1, y(k+1))$$
$$\rho_{0|0} = \Pi_{0|0}. \tag{3.17}$$
$$\rho_{k+1|k} = \rho_{k|k-1} S(k, u(k), y(k))$$
$$\rho_{0|-1} := \Pi_0. \tag{3.18}$$

Then

$$\Pi_{k|k} = \frac{\rho_{k|k}}{\rho_{k|k} e}$$
$$\Pi_{k+1|k} = \frac{\rho_{k+1|k}}{\rho_{k+1|k} e}, \qquad k = 0, 1, \cdots \tag{3.19}$$

which is the reason we call $\rho_{k|k}, \rho_{k+1|k}$ the unnormalized probability vectors. As a result of (3.17)–(3.19), we have the following.

Corollary 3.1: $\rho_{k|k-1}$ is an information state at time k for (3.1), (3.2).

The importance of considering the "unnormalized" versions (3.17) and (3.18), rests primarily on their linearity [compared to the nonlinear equations (3.11) and (3.14)], as it has been recently emphasized in nonlinear filtering studies [14], [15]. The recursions (3.17) and (3.18) are slight modifications (appropriate for our framework) of the results of Rudemo [22].

B. Application to Queueing Models in Discrete Time.

In this section we apply the methodology of the previous section to the queueing models developed in Section II. Actually, we will apply these results to more general queueing models where the arrival and departure rates are allowed to depend on the queue size. We first consider a single controlled queue and then we will consider two interacting queues as it is necessary for the application described in Section II.

Let the arrival and departure rates of a controlled queue in discrete time be given by

$$\lambda(t, i, v) = \Pr\left\{\begin{array}{l}\text{an arrival occurs in } [t, t+1) \text{ when}\\ \text{the queue size at time } t \text{ is } i \text{ and}\\ \text{control } u(t) = v\end{array}\right\}$$

and

$$\mu(t, i, v) = \Pr\left\{\begin{array}{l}\text{a departure occurs in } [t, t+1) \text{ when}\\ \text{the queue size at time } t \text{ is } i \text{ and}\\ \text{control } u(t) = v\end{array}\right\}. \tag{3.20}$$

We assume, as stated earlier, that time discretization is such that the probability of more than one arrival or departure in a single time slot is zero. It is convenient to introduce the arrival and departure point processes $n_a(t), n_d(t)$. Clearly,

$$n_a(t) = \begin{cases} 1, & \text{if an arrival occurs}\\ & \text{in } t\text{th time slot}\\ 0, & \text{otherwise}\end{cases}$$

$$n_d(t) = \begin{cases} 1, & \text{if a departure occurs}\\ & \text{in } t\text{th time slot}\\ 0, & \text{otherwise.}\end{cases}$$

[2] We thank the anonymous reviewer for this comment.

Then

$$\lambda(t,i,v)=\Pr\{n_a(t)=1|x(t)=i,u(t)=v\}$$
$$\mu(t,i,v)=\Pr\{n_d(t)=1|x(t)=i,u(t)=v\}. \quad (3.21)$$

Here $x(t)$ denotes the queue size during the tth time slot and is integer valued. We have assumed here that the queue

$$\begin{aligned} M_{ii}(t,v,1)&=\mu(t,i,v)\\ M_{ii+1}(t,v,1)&=1-\mu(t,i,v)\\ M_{ij}(t,v,1)&=0, \qquad \text{elsewhere} \end{aligned}$$

is controlled by controlling the departure and (or) the arrival rate. The departure rate $\mu(t,i,v)$ is also referred to as the service rate. Finally, we make the usual assumption (see [7]) that the state of the departure and arrival processes in slot $t+1$ are conditionally independent given the queue size and the control value in slot t.

We are interested in the case of partially observed queues. Thus, as in Section II, we assume that arrivals, i.e., $n_a(t)$ are observed, while departures, i.e., $n_d(t)$, are not observed. In the terminology of Section III-A

$$y(t)=n_a(t). \quad (3.22)$$

To describe the partially observed queue statistically in the framework of Section III-A, we need to specify the matrix of joint statistics S_{ij} [see (3.1)]. Thus,

$$\begin{aligned} S_{ij}(t,v,1)&=\Pr\{x(t+1)=j,n_a(t)=1|x(t)=i,u(t)=v\}\\ &=\Pr\{x(t+1)=j|n_a(t)=1,x(t)=i,u(t)=v\}\\ &\quad\cdot\Pr\{n_a(t)=1|x(t)=i,u(t)=v\}. \end{aligned}$$

Therefore,

$$\begin{aligned} S_{ii}(t,v,1)&=\lambda(t,i,v)\mu(t,i,v)\\ S_{ii+1}(t,v,1)&=\lambda(t,i,v)(1-\mu(t,i,v))\\ S_{ij}(t,v,1)&=0, \qquad \text{elsewhere.} \end{aligned} \quad (3.23)$$

Similarly,

$$\begin{aligned} S_{ii}(t,v,0)&=(1-\lambda(t,i,v))(1-\mu(t,i,v))\\ S_{ii-1}(t,v,0)&=(1-\lambda(t,i,v))\mu(t,i,v)\\ S_{ij}(t,v,0)&=0, \qquad \text{elsewhere.} \end{aligned} \quad (3.24)$$

In the case of a queue evolving without bounds the only constraint imposed on $\lambda(t,i,v)$, $\mu(t,i,v)$ is that

$$\mu(t,0,v)=0, \qquad \forall t,v\in\mathcal{U}. \quad (3.25)$$

On the other hand, in the case of a finite queue bound, i.e., when the queue is not allowed to grow beyond N, in addition to (3.25) we have

$$\lambda(t,N,v)=0, \qquad \forall t,v\in\mathcal{U}. \quad (3.26)$$

We note that the matrix of state transition probabilities computed from the S matrix given above, according to (3.7), coincides with that used in our earlier work [7, eq. (2.19)]. The point we wish to make here is that the description of S is more appropriate as a starting point for queueing models.

Since we are only interested in queueing systems which are controlled by controlling the departure rate (or service rate), we shall drop the argument v from the arrival rate λ, for the remainder of this paper.

The matrix M introduced in (3.9) now becomes

$$\begin{aligned} M_{ii}(t,v,0)&=1-\mu(t,i,v)\\ M_{ii-1}(t,v,0)&=\mu(t,i,v)\\ M_{ij}(t,v,0)&=0, \qquad \text{elsewhere.} \end{aligned} \quad (3.27)$$

Furthermore, the matrix D introduced in (3.12) now becomes

$$\begin{aligned} D(t,1)&=\operatorname{diag}\{\lambda(t,i)\},\\ D(t,0)&=\operatorname{diag}\{1-\lambda(t,i)\}. \end{aligned} \quad (3.28)$$

Substituting (3.27) and (3.28) in (3.11) and (3.14) we obtain

$$\Pi_{t|t}(i)=\begin{cases} \dfrac{\lambda(t,i)\Pi_{t|t-1}(i)}{\Sigma_{i\in I}\lambda(t,i)\Pi_{t|t-1}(i)}, & \text{if } n_a(t)=1\\ \dfrac{(1-\lambda(t,i))\Pi_{t|t-1}(i)}{\Sigma_{i\in I}(1-\lambda(t,i))\Pi_{t|t-1}(i)}, & \text{if } n_a(t)=0 \end{cases} \quad (3.29)$$

$$\Pi_{t+1|t}(i)=\begin{cases} \mu(t,i,v)\Pi_{t|t}(i)\\ \quad+(1-\mu(t,i-1,v))\Pi_{t|t}(i-1),\\ \quad \text{if } n_a(t)=1\\ (1-\mu(t,i,v))\Pi_{t|t}(i)\\ \quad+\mu(t,i+1,v)\Pi_{t|t}(i+1), & \text{if } n_a(t)=0. \end{cases} \quad (3.30)$$

Thus, we have recovered [7, eqs. (3.14) and (3.15)]. We have obtained via elementary methods the filtering and prediction formulas for queue size obtained previously in [7]. Furthermore, we have also obtained the unnormalized versions of these filtering and prediction formulas [cf. (3.17) and (3.19)].

We next describe the application of the methodology of Section III-A to a generalization of the two competing queues problem of Section II. Each queue will be described as before and we shall use superscripts 1 or 2 to refer to the parameters for each queue. In particular, S^1, S^2 will be matrices like (3.23) and (3.24) describing the stochastic queue model. The observation on the combined two queues system is

$$y(t)=(y_1(t),y_2(t))=(n_a^1(t),n_a^2(t)). \quad (3.31)$$

Again the control is applied through the departure rates of each queue. Although we can accomodate more general models, we shall assume the following independence condition to simplify the computations.

Independence Condition: Each queue's transitions and the state of the arrival process in slot $t+1$ are conditionally independent given the queue sizes and the control value in slot t.

In most practical applications (e.g., urban traffic control, computer or communication networks) this condition is usually satisfied. It basically expresses the observed fact that when queues are competing and the control is through the server, the basic coupling between the two arrival processes is provided through the control and each queue's evolution in response to the control.

The state for the combined system is $(x_1(t), x_2(t))$. The state set for the combined system is clearly $I\times I$. The S matrix [cf. (3.1)] for the combined system is

$$S_{i_1i_2,j_1j_2}(t,v,\psi):=S^1_{i_1j_1}(t,v,\psi_1)S^2_{i_2j_2}(t,v,\psi_2) \quad (3.32)$$

or

$$S(t,v,\psi)=S^1(t,v,\psi_1)\otimes S^2(t,v,\psi_2) \quad (3.33)$$

where $\otimes$ denotes tensor product. Letting $D(t,n^1_a(t))$, $D^2(t,n^2_a(t))$ denote the D-matrices for each queue [see (3.28)], the output probabilities for the combined system are

$$\begin{aligned}\Pr\{y(t)=(\psi_1,\psi_2)|x_1(t)=i_1,x_2(t)=i_2,u(t)=v\}&\\ =\sum_{j_1,j_2\in I}S_{i_1i_2,j_1j_2}(t,v,\psi)&\\ =D^1_{i_1i_1}(t,\psi_1)D^2_{i_2i_2}(t,\psi_2).&\end{aligned}$$

Therefore, the D-matrix for the combined system is

$$D(t,y(t))=D^1\big(t,n^1_a(t)\big)\otimes D^2\big(t,n^2_a(t)\big). \quad (3.34)$$

Similarly, if $M^1(t,v,n^1_a(t))$, $M^2(t,v,n^2_a(t))$ denote the M-matrices for each queue [see (3.27)] the M-matrix for the combined system is

$$M(t,v,y)=M^1\big(t,v,n^1_a(t)\big)\otimes M^2\big(t,v,n^2_a(t)\big). \quad (3.35)$$

Assuming that the initial probability vectors Π^1_0, Π^2_0 for the two queue sizes are independent, implies that the initial probability vector for the combined state can be written as

$$\Pi_0=\Pi^1_0\otimes\Pi^2_0. \quad (3.36)$$

Then the initial condition for the filtering-prediction recursions for the combined system is [cf. (3.15)]

$$\Pi_{0|0}=\frac{\Pi_0D(0,y(0))}{\Pi_0D(0,y(0))e}=\Pi^1_{0|0}\otimes\Pi^2_{0|0} \quad (3.37)$$

in view of (3.34). In view of (3.33) and (3.36) it is plain that (3.11), (3.14) inductively imply

$$\Pi_{k|k}=\Pi^1_{k|k}\otimes\Pi^2_{k|k}, \qquad k=0,1,\cdots \quad (3.38)$$

$$\Pi_{k+1|k}=\Pi^1_{k+1|k}\otimes\Pi^2_{k+1|k}, \qquad k=0,1,\cdots. \quad (3.39)$$

Clearly, similar tensor product expressions are valid for the unnormalized filtered and one-step predicted probability vectors of the combined two queue system. Clearly, $\Pi^l_{t+1|t}, \Pi^l_{t|t}$, $l=1,2$ are given by the recursions (3.29), (3.30) for each queue.

To emphasize the significance of this "decoupling" of the filtering-prediction recursions for the stochastic control problem, we state it as (compare with Theorem 3.1, Corollary 3.1).

Theorem 3.2: For strictly nonanticipative control strategies $(\Pi^1_{k|k-1}, \Pi^2_{k|k-1})$ (or $(\rho^1_{k|k-1}, \rho^2_{k|k-1})$) is an information state at time k for the combined two queues system.

IV. Finite Horizon Stochastic Optimal Control

A. Formulation and Methodology

In this section we review briefly existing basic results that can be used in finite horizon stochastic control problems, when the partially observed controlled system is of the type described in Section III-A.

We shall denote by T the finite time horizon, by $c(k,x,u)$ the running cost and by $c(T,x)$ the terminal cost. For a control policy γ the cost is denoted by

$$J(\gamma):=E\left\{\sum_{k=0}^{T-1}c(k,x^\gamma(k),u^\gamma(k))+c(T,x^\gamma(T))\right\} \quad (4.1)$$

where the superscript γ in x,u, indicates the state and control trajectories induced by the policy γ. For a partially observed stochastic system, modeled by (3.1) and (3.2), we wish to select $\gamma\in\Gamma$ [see (3.3)–(3.6)] in order to minimize $J(\gamma)$. It is well known [16], [17] that due to partial observation of the state the optimal policy will not be Markovian [18]. It has been shown in [16]–[19], for various partially observed stochastic systems, that instead the optimal policy is a function only of the information state which is typically the vector of filtered probabilities $\Pi_{k|k}$ [see (3.10)]. For the intended application a slight modification of this result is needed, to reflect the fact that for the queueing problem of interest the information state is $\Pi_{k|k-1}$ (see in particular Theorem 3.2). We briefly sketch the basic steps here since the methodology is standard dynamic programming and is well known [16]–[19].

Following [9] we shall call a policy $\gamma=\{g_0,g_1,\cdots,g_k,\cdots\}$ *separated* if g_k depends on the available information vector at time k, z^k, "through" the information state $\Phi(k)$ [see (3.16)], i.e., $u(k)=g_k(\Phi(k))$. We let Γ_s be the subset of separated strictly nonanticipative policies.

Let π be the set of all probability vectors $\{\Pi(i)\}_{i\in I}$, $\Pi(i)\geq 0$, $\Sigma_{i\in I}\Pi(i)=1$. Letting $z^{\gamma,k}$ denote the information vector sample path generated while using policy γ we consider as usual the sequence of costs

$$J^\gamma_k:=E\left\{\sum_{l=k}^{T-1}c(l,x^\gamma(l))+c(T,x)|z^{\gamma,k}\right\}, \qquad k=0,1,\cdots,T. \quad (4.2)$$

To simplify later computations let $C_T, C_k(u)$, $k=0,1,\cdots,T-1, u\in\mathcal{U}$ be the column vectors

$$C_T(i)=c(T,i)$$
$$[C_k(u)](i)=c(k,i,u), \qquad i\in I. \tag{4.3}$$

For strictly nonanticipative policies we have the following standard dynamic programming result [9], [16]–[19].

Theorem 4.1: For $0\leq k\leq T$, define the functions $V_k(\cdot)$ on π via

a) $V_T(\Pi)=\Pi C_T$

b) $$V_k(\Pi)=\inf_{u\in\mathcal{U}}\left[\Pi C_k(u)+\Sigma_{\psi\in\mathcal{Y}}V_{k+1}\cdot\left(\frac{\Pi S(k,u,\psi)}{\Pi S(k,u,\psi)e}\right)\cdot\Pi D(k,\psi)e\right].$$

Then for $\gamma\in\Gamma$

$$V_k\left(\Pi_{k|k-1}(z^{\gamma,k-1})\right)\leq J_k^\gamma, \qquad k=0,1,2,\cdots,T. \tag{4.4}$$

Furthermore, let $\gamma^*\in\Gamma_s$, be a separated policy such that $g_k^*(\Pi)$ achieves the infimum in (b). Then γ^* is optimal in Γ and

$$V_k\left(\Pi_{k|k-1}(z^{\gamma^*,k-1})\right)=J_k^{\gamma^*}, \qquad k=0,1,\cdots,T \tag{4.5}$$

with probability 1.

Theorem 4.1, characterizes optimal policies as feedback laws on estimates of the states. In cases where the dynamic programming recursions can be solved explicitly for the functions V_k, the only on line implementation needed for the control policy is that of the filter/predictor [see (3.11) and (3.14)].

B. Application to the Two Competing Queues Problem.

Here we consider the stochastic control formulation of the two queues problem discussed in Section III-B. The queues are partially observed through their arrival processes. They are modeled as described in the second part of Section III-B.

We denote by c_1, c_2 (nonnegative real numbers) the cost per unit time delay in queues 1 and 2, respectively. Here we assume c_1, c_2 are constants. Generalizations will be considered elsewhere. As a running cost we choose the delay cost $c_1x_1(t)+c_2x_2(t)$ per unit time. So for a policy γ the cost is the average aggregate delay cost

$$J(\gamma):=E\left\{\sum_{i=0}^{T}c_1x_1^\gamma(t)+c_2x_2^\gamma(t)\right\}. \tag{4.6}$$

The model for the combined two queues system was described in (3.31)–(3.37).

We consider strictly nonanticipative control policies. The optimal policy and value function are described then by Theorem 4.1. Since the state set is $I\times I$ we order the components of the probability vectors of the combined system according to the sequence $00, 01, 02,\cdots, 10, 11, 12,\cdots, 20, 21, 22,\cdots$. From (4.3) and (4.6)

$$C_k(i_1,i_2)=c_1i_1+c_2i_2, \qquad i_1,i_2\in I. \tag{4.7}$$

Let ν be the vector

$$\nu=[0 \quad 1 \quad 2 \quad 3 \quad 4 \quad \cdots]^T. \tag{4.8}$$

Then

$$C_k=c_1\nu\otimes e+c_2e\otimes\nu \tag{4.9}$$

where $\otimes$ denotes again tensor product. Since C_k does not depend on k, we shall drop the k dependence

$$C_k=C, \qquad k=0,1,\cdots.$$

Due to the "decoupling" of the filtering-prediction recursions discovered in Section III-B [see (3.38)] the dynamic programming recursion of Theorem 4.1b) reduces to

$$V_T(\Pi^1,\Pi^2)=c_1\Pi^1\nu+c_2\Pi^2\nu$$

$$V_k(\Pi^1,\Pi^2)=\inf_{u\in\mathcal{U}}\Bigg[c_1\Pi^1\nu+c_2\Pi^2\nu+\sum_{\psi_1,\psi_2\in\{0,1\}}V_{k+1}\left(\frac{\Pi^1S^1(k,u,\psi_1)}{\Pi^1S^1(k,u,\psi_1)e},\frac{\Pi^2S^2(k,u,\psi_2)}{\Pi^2S^2(k,u,\psi_2)e}\right)\cdot\left(\Pi^1D^1(k,\psi_1)e\right)\left(\Pi^2D^2(k,\psi_2)e\right). \tag{4.10}$$

In (4.10) Π^1,Π^2 are probability vectors over I, S^i, D^i, $i=1,2$, are described for each queue by (3.23)–(3.28). Recall from (3.7) that the matrix of transition probabilities is given for each queue by

$$P^i(t,v)=\sum_{\psi=0}^{1}S^i(t,v,\psi), \qquad i=1,2. \tag{4.11}$$

We next can show that (4.10) can be solved easily off-line and that all functions V_k, $k=0,1,\cdots,T$ are piecewise linear in Π^1,Π^2. The piecewise linearity of the value function has been established previously in [19]. To see this we work inductively backwards.

For $k=T$, (4.10) implies

$$V_T(\Pi^1,\Pi^2)=\Pi^1d_T^1+\Pi^2d_T^2, \qquad \forall\Pi^1,\Pi^2\in\pi \tag{4.12}$$

where

$$d_T^i=c_i\nu, \qquad i=1,2 \tag{4.13}$$

are column vectors of dimension equal to the cardinality of I. Next at $k=T-1$, (4.10) implies

$$V_{T-1}(\Pi^1,\Pi^2)=\min_{u\in\{0,1\}}\left[\Pi^1\left(\mathcal{I}+P^1(T-1,u)\right)d_T^1+\Pi^2\left(\mathcal{I}+P^2(T-1,u)\right)d_T^2\right] \tag{4.14}$$

where $\mathcal{I}$ is the identity operator of dimension equal to the cardinality of I. Clearly, the optimal control as a function of Π^1,Π^2 is described as follows. The set $\pi\times\pi$ is separated in two disjoint subsets

$$A_1=\{(\Pi^1,\Pi^2)\in\pi\times\pi,\quad \text{s.t.}$$
$$\Pi^1\big(P^1(T-1,0)-P^1(T-1,1)\big)d_T^1$$
$$\geqslant\Pi^2\big(P^2(T-1,1)-P^2(T-1,0)\big)d_T^2\}$$
$$A_0=\text{complement of } A_1 \text{ in } \pi\times\pi. \tag{4.15}$$

We associate the index 1 with A_1, the index 0 with A_0, since clearly

$$u^*(T-1)=\begin{cases}1, & \text{on } A_1\\ 0, & \text{on } A_0.\end{cases} \tag{4.16}$$

Let $a_{T-1}(\Pi^1,\Pi^2)$ be the function

$$a_{T-1}(\Pi^1,\Pi^2)=\begin{cases}1, & \text{if } (\Pi^1,\Pi^2)\in A_1\\ 0, & \text{if } (\Pi^1,\Pi^2)\in A_0\end{cases}$$

and

$$d_{T-1}^1(\Pi^1,\Pi^2)=\big(\mathcal{G}+P^1\big(T-1,a_{T-1}(\Pi^1,\Pi^2)\big)\big)d_T^1$$
$$d_{T-1}^2(\Pi^1,\Pi^2)=\big(\mathcal{G}+P^2\big(T-1,a_{T-1}(\Pi^1,\Pi^2)\big)\big)d_T^2. \tag{4.17}$$

It is now clear that

$$V_{T-1}(\Pi^1,\Pi^2)=\Pi^1d_{T-1}^1(\Pi^1,\Pi^2)+\Pi^2d_{T-1}^2(\Pi^1,\Pi^2) \tag{4.18}$$

and, therefore, V_{T-1} is piecewise linear also. The general computation follows from the following lemma.

Lemma 4.1: Define the binary valued functions a_l, $l=0,1,\cdots,T-1$, on $\pi\times\pi$ (where π is the set of probability vectors on I), and the column vectors d_l^i, $i=1,2$, $l=0,1,\cdots,T$ (of dimension equal to the cardinality of I) by the backward recursions

$$d_T^i=c_i\nu$$
$$d_{T-l}^i=d_T^i+P^i(T-l,a_{T-l})d\cdot d_{T-l+1}^i(a_{T-l+1},\cdots,a_{T-1})$$
$$a_{T-l}(\Pi^1,\Pi^2)=\begin{cases}1, & \text{if } \Pi^1\big[P^1(T-l,0)-P^1(T-l,1)\big]\cdot d_{T-l+1}^1(a_{T-l+1},\cdots,a_{T-1})\\ & \geqslant\Pi^2\big[P^2(T-l,1)-P^2(T-l,0)\big]\cdot d_{T-l+1}^2(a_{T-l+1},\cdots,a_{T-1})\\ 0, & \text{otherwise}\end{cases}$$
$$i=1,2,\quad l=1,2,\cdots,T-1.$$

Then for $k=0,1,\cdots,T$, $(\Pi^1,\Pi^2)\in\pi\times\pi$

$$V_k(\Pi^1,\Pi^2)=\Pi^1d_k^1(a_k,\cdots,a_{T-1})+\Pi^2d_k^2(a_k,\cdots,a_{T-1}).$$

That is V_k is piecewise linear for each k.

Note: The recursive backwards computation proceeds along the diagram:

$$\begin{pmatrix}d_T^1\\ d_T^2\\ a_{T-1}\end{pmatrix}\to\begin{pmatrix}d_{T-1}^1\\ d_{T-1}^2\\ a_{T-2}\end{pmatrix}\to\cdots\to\begin{pmatrix}d_{T-l}^1\\ d_{T-l}^2\\ a_{T-l-1}\end{pmatrix}\to\begin{pmatrix}d_{T-l-1}^1\\ d_{T-l-1}^2\\ a_{T-l-2}\end{pmatrix}\to\cdots\to\begin{pmatrix}d_0^1\\ d_0^2\end{pmatrix}.$$

Proof: Let us assume the result holds for $k=T-l$. Then from (4.10), after a computation identical to (4.14), we have

$$V_{T-l-1}(\Pi^1,\Pi^2)=\min_{u\in\{0,1\}}\big[\Pi^1d_T^1+\Pi^2d_T^2+\Pi^1P^1(T-l-1,u)d_{T-l}^1+\Pi^2P^2(T-l-1,u)d_{T-l}^2\big].$$

It is obvious now by the definition of $a_{T-l-1}(\Pi^1,\Pi^2)$ and d_{T-l-1}^1, d_{T-l-1}^2 that the result holds for $k=T-l-1$.

We also have established the following Corollary.

Corollary 4.1: The optimal control policy in feedback form, as function of Π^1, Π^2 is given by

$$u(k)=a_k(\Pi^1,\Pi^2),\qquad k=0,1,\cdots,T-1.$$

Combining now the results of Lemma 4.1, Corollary 4.1, and (4.5) of Theorem 4.1 we have established the following result.

Theorem 4.2: The optimal server time allocation strategy and expected aggregate delay, when using strictly nonanticipative strategies are determined as follows. First, the vectors d_l^i, $i=1,2$, $l=0,1,\cdots,T$ and binary valued functions a_l, $l=0,1,\cdots,T-1$ are computed off-line and stored from Lemma 4.1. For each queue the one-step queue predicted probability vectors $\Pi_{k|k-1}^i$, $i=1,2$ are computed, using the recursions (3.29) and (3.30) with initial conditions (3.36). The optimal strategy at time k is

$$u_k=a_k\big(\Pi_{k|k-1}^1,\Pi_{k|k-1}^2\big),\qquad k=0,1,2,\cdots,T-1. \tag{4.19}$$

The optimal average aggregate delay is

$$V_0\big(\Pi_0^1,\Pi_0^2\big)=\Pi_0^1d_0^1+\Pi_0^2d_0^2. \tag{4.20}$$

Note: The vectors d_0^1, d_0^2 in (4.20) are functions of Π_0^1,Π_0^2.

The implementation is rather interesting. The decision space $\pi\times\pi$ is divided at most to 2^{T-k} subsets which are characterized by binary numbers with $T-k$ binary digits, i.e., $a_ka_{k+1}\cdots a_{T-1}$. The first binary digit of the number associated with the subset provides according to Corollary 4.1 the optimal control in feedback form. These observations are quite useful when implementing these strategies in a microprocessor. The only on-line computation needed, as emphasized earlier, is that of the filter-predictor (3.29) and (3.30) which as we have shown elsewhere [7] are easily

implemented on a microprocessor. Specific computational examples will be given in the next section.

Finally, observe that the value functions V_k are concave in Π^1, concave in Π^2 for $k=0,1,\cdots,T$; a fact that follows easily from the defining backwards recursion (4.10) by an inductive argument.

V. An Example

As an illustration of the foregoing methodology, we consider the example of Section II with the same arrival rates ($\lambda^1=\lambda^2=\lambda$) and waiting costs ($c_1=c_2=c$). The aggregate delay under the optimal strictly nonanticipative policy of Theorem 4.2 is compared with two suboptimal policies by means of Monte Carlo simulation.

The combined two queue system is modelled as described in (3.31)–(3.39) with the performance objective of (4.6). A finite buffer size ($N_1=N_2=10$) is simulated, with each queue and state estimator initialized to zero customers (w.p.1). For the finite-horizon ($T=50$), Bernoulli arrival and departure processes are generated at each time step such that

i) no customers arrive in a queue when it is full; $i=1,2$ [see (3.26)].
ii) no customer departs from queue i when either queue i has zero customers or queue j ($j\neq i$) is being served [see (3.25)].

The selection of the optimal control sequence follows from the normals to the hyperplanes characterizing the value functions $V_k(\Pi^1,\Pi^2)$ of Lemma 4.1. To weigh the merits of the calculation of these normals (optimal policy), two suboptimal policies were simulated. Using the sufficient statistic $(\Pi^1_{k|k-1},\Pi^2_{k|k-1})$ of the one-step predictor, we define

$$B=\{(\Pi^1,\Pi^2)\in\pi\times\pi:\ \Pi^2(0)>\Pi^1(0)\}$$

$$C=\{(\Pi^1,\Pi^2)\in\pi\times\pi:\ \Pi^1\nu>\Pi^2\nu\}$$

and let

$$u(k)^{\Pi}=\begin{cases}1, & \text{if } (\Pi^1_{k|k-1},\Pi^2_{k|k-1})\in B\\ 0, & \text{otherwise}\end{cases}$$

$$u(k)^{\text{MMSE}}=\begin{cases}1, & \text{if } (\Pi^1_{k|k-1},\Pi^2_{k|k-1})\in C\\ 0, & \text{otherwise.}\end{cases}$$

The probability-of-zero strategy $\{u(k)^{\Pi};\ k=0,1,\cdots,T-1\}$ chooses to serve the queue which has the higher probability of being nonempty. The MMSE strategy $\{u(k)^{\text{MMSE}};\ k=0,1,\cdots,T-1\}$ selects the queue having the higher estimated queue size. Recall from (3.10),

$$\hat{x}_i(k|k-1)=E\{x_i(k)|y^{k-1},u^{k-1}\}$$
$$=\Pi^i_{k|k-1}\nu,\qquad i=1,2.$$

Both suboptimal policies are computationally simpler by disregarding the future evolutions of the Markov chain and only consider the immediate cost. On the other hand, the optimal strategy incorporates the coupling of the future states via the dynamic programming formulation. Thus, one expects on the average that the performance of the optimal strategy is superior to the suboptimal ones.

The parameters selected for the model were chosen rather arbitrarily. The case of unity cost ($c_1=c_2=1$) with the same arrival rate was investigated because the symmetry of the problem provides a better insight into the control selection process while not introducing other factors. Clearly, for the two suboptimal policies, the defining sets B and C are different when the waiting cost or the arrival rates differ. The arrival and departure rates were chosen so that two different traffic conditions are represented, light and heavy.

Figs. 2 and 3 show results obtained in the first case, intented to represent light to moderate traffic. Here $\lambda^1=\lambda^2=0.35$, $\mu=0.70$. In Fig. 2 a particular sample path of the optimal and two suboptimal policies is shown. We show for each policy three graphs. The first and second depict the time histories of the queues, while the third depicts the time history of the control policy. The same arrival processes are used under each control law and the aggregate delay for each policy is computed by summing the two queue sizes over the finite horizon. For the case of Fig. 2 the optimal policy results in an aggregate delay of 130, the policy u^{Π} results in 153 and the policy u^{MMSE} in 128. In order to evaluate better these three policies we show in Fig. 3 a table with aggregate delays achieved in 50 samples. It is seen that all policies perform comparably for most sample paths. This raises the very interesting question of obtaining some analytical comparison results. This problem will be studied elsewhere.

Similar results are presented in Figs. 4 and 5 for a heavier traffic case with $\lambda^1=\lambda^2=0.35$, $\mu=0.35$. It is seen that in this heavier traffic case, the optimal policy performs considerably better in almost all samples. These observations seem to imply that in heavier traffic the difference between optimal and suboptimal policies is greater, while in light traffic it is negligible. This conclusion agrees with intuition. Its analytical establishment, however is an open problem.

VI. Conclusions

In this paper we have established a framework for filtering/prediction and stochastic control for queuing systems in discrete time. We have applied this general framework to the problem of two partially observed competing queues and have obtained an explicit solution for the finite time horizon problem. We have explained how the solution can be easily implemented and have presented and evaluated briefly two other policies, suggested on the basis of intuition. An analytical evaluation of practical policies which are easier to implement than the optimal policy has emerged as a significant future problem.

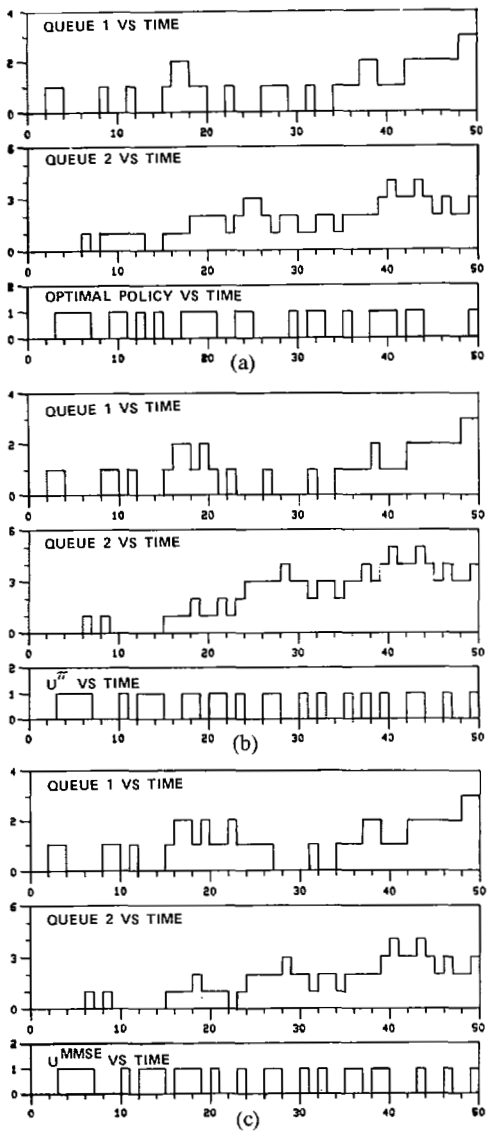


Fig. 2. Comparison of three policies when $\lambda^1=\lambda^2=0.35$, $\mu=0.7$ (a) optimal policy, u^*, (b) u^{π} policy, (c) u^{MMSE} policy.

u^*	u^{π}	u^{MMSE}
130	153	128
172	159	172
164	198	162
196	181	181
254	218	215
120	83	83
203	203	203
74	54	59
158	154	170
175	144	121
172	171	177
101	53	74
392	392	392
216	208	218
186	185	185
270	251	251
197	239	226
261	227	227
125	86	86
95	78	77
64	79	79
91	89	99
85	81	81
57	59	59
183	182	182
178	139	144
458	426	458
79	82	82
254	254	254
412	412	412
366	366	366
228	228	266
201	197	234
229	238	238
54	40	39
78	77	77
160	137	137
176	173	173
231	162	162
219	219	219
144	94	106
232	235	235
215	215	215
138	126	125
51	40	40
113	77	94
161	154	155
340	302	302
203	200	200
132	79	89

Fig. 3. Aggregate delays achieved in 50 samples, by each policy. $\lambda^1=\lambda^2=0.35$, $\mu=0.7$.

ACKNOWLEDGMENT

The first author would like to thank Z. Rosberg, P. Varaiya, and J. Walrand for many stimulating discussions on the subject matter of this paper. He would like to thank again P. Varaiya for making available his unpublished notes on stochastic optimal control [9] and for his hospitality during the author's visit at the University of California, Berkeley, in August 1980.

REFERENCES

[1] T. B. Crabill, D. Gross, and M. J. Magazine, "A classified bibliography of research on optimal design and control of queues," *Operations Res.*, vol. 25, no. 2, pp. 219–232, Mar.–Apr. 1977.
[2] M. J. Sobel, "Optimal operation of queues," in *Mathematical Methods in Queueing Theory*, A. B. Clarke, Ed., *Lecture Notes in Economics and Math. Systems*, vol. 98, pp. 231–261, 1974.
[3] S. Stidham, Jr., and N. V. Prabhu, "Optimal control of queueing systems," in *Mathematical Methods in Queuing Theory*, A. B. Clarke, Ed., *Lecture Notes in Economics and Math. Systems*, vol. 98, pp. 263–294, 1974.
[4] J. Walrand and P. Varaiya, "The output of Jacksonian networks are Poissonian," Univ. California, Berkeley, ERL-M78/60, July 1978.
[5] P. Brémaud, "Optimal thinning of a point process," *SIAM J. Contr. Optim.*, vol. 17, no. 2, pp. 222–230, Mar. 1979.
[6] Z. Rosberg, P. Varaiya, and J. Walrand, "Optimal control of service in tandem queues," Electronics Res. Lab., Univ. California, Berkeley, Memo. UCB/CRL M80/42, Sept. 1980.
[7] J. S. Baras, W. S. Levine and T. L. Lin, "Discrete time point processes in urban traffic queue estimation," *IEEE Trans. Automat. Contr.*, vol. AC-24, pp. 12–27, Feb. 1979.
[8] J. R. Jackson, "Networks of waiting lines," *Operations Res.*, vol. 5, pp. 518–521, 1957.
[9] P. Varaiya, "Notes on stochastic control," unpublished class notes, Univ. California, Berkeley.
[10] A. Segall, "Dynamic file assignment in a computer network," *IEEE Trans. Automat. Contr.*, vol. AC-21, pp. 161–173, Apr. 1976.
[11] B. Hajek, "Dynamic decentralized estimation and control in multiaccess broadcast channel," in *Proc. 19th Decision and Control Conf.*, Albuquerque, NM, pp. 618–623, Dec. 1980.
[12] R. Kalman, P. Falb, and M. Arbib, *Topics in Mathematical System Theory*. New York: McGraw-Hill, 1969.
[13] K. Åstrom, "Optimal control of Markov processes with incomplete state information I," *J. Math. Anal. Appl.*, vol. 10, pp. 174–205, 1965.
[14] R. W. Brockett and J. M. C. Clark, "Geometry of the conditional density equation," in *Proc. Int. Conf. Analysis and Optim. of Stochastic System*, Oxford, England, 1978.
[15] S. K. Mitter, "On the analogy between mathematical problems of non-linear filtering and quantum physics," *Rich. Automatica*, 1980.
[16] K. Åstrom, "Optimal control of Markov processes with incomplete state information II: Convexity of the loss function," *J. Math. Anal. Appl.*, vol. 26, pp. 403–406, 1969.
[17] C. Striebel, "Sufficient statistics in the optimal control of stochastic systems," *J. Math. Anal. Appl.*, vol. 12, pp. 576–592, 1965.

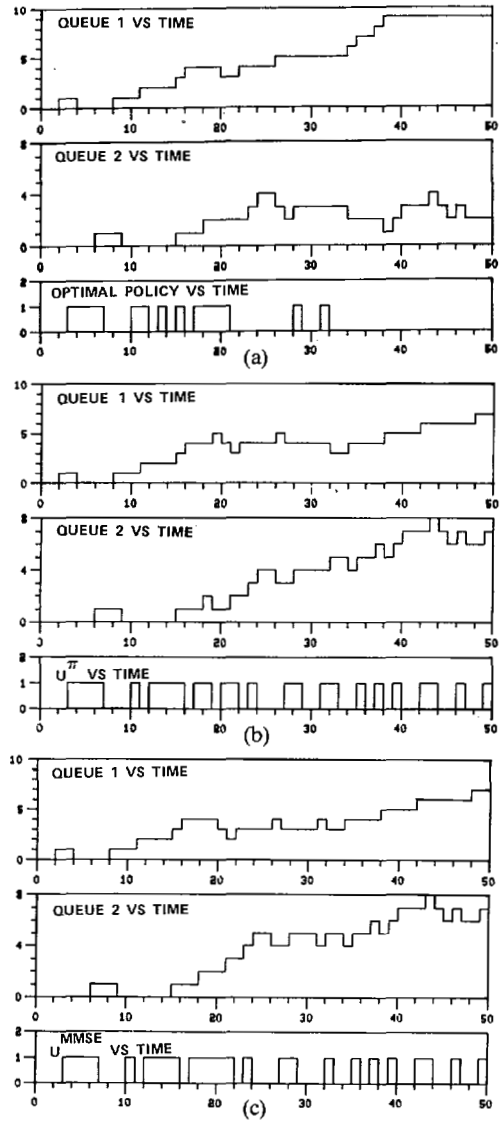


Fig. 4. Comparison of three policies when $\lambda^1=\lambda^2=0.35$, $\mu=0.35$ (a) optimal policy, u^*, (b) u^{π} policy, (c) u^{MMSE} policy.

u^*	u^{π}	u^{MMSE}
329	343	343
383	471	481
448	483	489
463	548	557
407	452	463
342	391	391
389	377	384
273	257	257
406	444	444
384	384	375
342	361	356
354	354	354
542	582	560
488	567	540
404	458	463
536	574	563
570	581	545
479	508	518
389	454	464
279	279	319
318	318	318
236	200	215
351	362	364
383	383	383
435	460	476
378	400	402
479	613	650
345	345	345
470	500	494
594	622	630
569	695	680
555	620	613
496	497	496
457	520	509
188	188	188
345	345	345
334	345	345
482	513	518
393	438	460
457	514	511
343	402	409
425	424	430
535	536	535
364	364	364
220	258	220
318	332	356
366	443	451
620	652	653
406	465	479
303	294	294

Fig. 5. Aggregate delays achieved in 50 samples, by each policy. $\lambda^1=\lambda^2=0.35$, $\mu=0.35$.

[18] D. Bertsekas, *Dynamic Programming and Stochastic Control*. New York: Academic, 1976.
[19] R. D. Smallwood and E. J. Sondik, "Optimal control of partially observable Markov processes over a finite horizon," *Oper. Res.*, vol. 21, no. 5, pp. 1071–1088, 1973.
[20] E. J. Sondik, "The optimal control of partially observable Markov processes over the infinite horizon: Discounted costs," *Oper. Res.*, vol. 26, no. 2, pp. 282–304, 1978.
[21] J. S. Baras and A. J. Dorsey, "Stochastic control of two partially observed competing queues," Tech. Rep., Electrical Engineering Dep., Univ. Maryland, College Park, Dec. 1980.
[22] M. Rudemo, "State estimation for partially observed Markov chains," *J. Math. Anal. Appl.*, vol. 44, pp. 581–611, 1973.

John S. Baras (M'73) was born in Piraeus, Greece, on March 13, 1948. He received the Electrical Engineering Diploma from the National Technical University of Athens, Athens, Greece, in 1970, and the M.S. and Ph.D. degrees in applied mathematics from Harvard University, Cambridge, MA, in 1971 and 1973, respectively.

Since 1973 he has been with the Electrical Engineering Department at the University of Maryland, College Park, where he is an Associate Professor in the Control Systems group. He has held research positions at M.I.T., Harvard University, and Stanford University. His current research interests include distributed parameter systems, nonlinear filtering, control of queueing systems, and large-scale optimization.

Dr. Baras is a member of Sigma Xi, the American Mathematical Society, and the Society for Industrial and Applied Mathematics.

Arthur J. Dorsey (S'78) was born in Baltimore, MD, in 1952. He received the B.S. and M.S. degrees in electrical engineering from the University of Maryland, College Park, in 1974 and 1978, respectively.

From 1974 to 1976, he was employed by IBM as a Development Engineer. From 1976 to 1978, he was a Research Assistant in the Electrical Engineering Department at the University of Maryland. He received a Minta Martin Fellowship from 1978 to 1980 in pursue of a doctorate degree. Presently, he is employed by IBM working in the area of global positioning satellites. His current research interests include estimation, adaptive control, and stochastic control of queuing systems.