ADAPTIVE CONTROL OF A SIMPLE QUEUEING SYSTEM

by

Arthur J. Dorsey

Dissertation submitted to the Faculty of the Graduate School
of the University of Maryland in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
1983

ABSTRACT

Title of Dissertation: Adaptive Control of a Simple Queueing System

Arthur Joseph Dorsey, Doctor of Philosophy, 1983

Dissertation directed by: John S. Baras
Professor
Electrical Engineering Department

The dynamic control of two queues competing for the service of a single server is treated. The problem is to optimally design a server time allocation strategy, under various performance criteria and conditioned on various information patterns. The instantaneous cost is selected as the total expected aggregate delay. The problem is formulated in discrete time. The arrival and departure processes at each queue are modelled as independent Bernoulli processes.

The research divides into three topic areas. First, the problem is formulated as a stochastic optimal control problem with complete observations, i.e. the controller's information at a decision epoch includes the past histories of the control values, departure and arrival data. The arrival rates and departure (service) rates are considered constant and known. The finite horizon, the infinite horizon discounted and the expected long-run average cost per unit time performance criteria are analyzed. In all cases, for the unbounded system the optimal service allocation strategy reduces to the "μc rule."

Second, the service allocation problem is studied as a stochastic optimal control problem with partial observations. Here, the information available to the controller includes the past histories of the control values and the arrival data, no past histories of departures are observed. The observations are modelled as discrete-time, 0-1 point processes whose rates are influenced by the size of each queue. By a simple application of Bayes rule and dynamic programming, we show that the "one step" predicted density for the state is a sufficient statistic for the control; the finite time expected aggregate delay criterion is considered. A special relationship between the queue transitions and the observations in such queueing system is noted.

Third, the competing queue problem is formulated as an adaptive control problem. The arrival rates and service rates are considered constant but unknown. The information available to the controller includes the past histories of the control values, departures and arrival data. The infinite horizon discounted and expected long-run average cost performance criteria are analyzed. Convergence results for certainty-equivalence type, adaptive control schemes are established. Several possible extensions of the results are discussed.

DEDICATION

This dissertation is dedicated to my Lord, my family and my friend, Lisa through the passage:

> "Neither he who plants nor he who waters is of any special account, only God who gives the growth. He who plants and he who waters work to the same end. Each will receive his wages in proportion to his toil."
>
> 1 Corinthians 3:8-10.

ACKNOWLEDGEMENTS

I wish to thank my advisor, Professor John S. Baras, for introducing me to this set of optimal control problems, and for patiently encouraging me through the years of this research. I would also like to acknowledge many helpful conversations with Professors Armand M. Makowski and William S. Levine. In addition, I extend my appreciation to the Electrical Engineering Department for providing a simulating environment and to my fellow graduate students for their diligent pursue for knowledge.

Special acknowledgement is given to IBM Corporation for providing me with a Leave of Absence during the early stages of this degree. This thesis was done in conjunction with research supported by the U.S. Department of Transportation under contract DOT-OS-60134, in part by the Minta Martin Fellowship Program and in part by the Computer Science Center of the University of Maryland.

Finally, I would like to thank Patricia Keehn for an excellent preparation of the manuscript.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure Page

1. INTRODUCTION

Classical queueing theory has been concerned with the properties of the queueing system under fixed modes of operation. Considerable attention has been directed towards optimal static control of these systems. Recently though, interest has been directed towards dynamical control. Despite their deceptively simple formulation, these dynamical optimization problems have highly nontrival solutions. Within this class of optimization problems is the priority assignment problem. The priority assignment problem simply stated regulates the customers in a system by assigning priorities to their respective service times. This important problem is encountered in such diverse applications as urban traffic control, multi-mode, packet-radio networks routing, inventory control and computer communication polling schemes. In this dissertation, a dynamic priority assignment problem is studied with the feature that its control strategy be adaptive.

The basic discrete-time priority assignment problem is loosely described as follows. At each unit time slot, two classes of customers, with different arrival and departure rates, enter the system and join their respective buffers. At most one arrival and one departure of each customer class can occur during any given time slot. Also, the buffers have either finite or infinite capacity. The two parallel queues then compete for the services of a single server. The server obeys a non-idling, non-preemptive priority queue discipline. At each service completion time, the controller selects which queue to service next. The controller's information at a decision epoch includes the past histories of the control values, departure and arrival data. The objective is to develop a service time allocation strategy to minimize various performance objectives. The expected finite horizon, the infinite horizon expected discount and the expected long-run average cost per unit time criteria are analyzed. The instantaneous cost is selected as the total aggregate delay.

This simple priority assignment problem is considered within the framework of control theory. First, the basic problem is formulated as a completely observed, stochastic optimal control problem. The arrival and departure rate parameters are assumed to be known constants. Under each criterion for the unbounded system, the optimal service allocation strategy reduces to the so called "μc rule." Second, the problem is analyzed with partial information. Specifically, the controller has available only the past histories of the control and the arrival processes; the departure processes are not observed. Under the finite horizon expected aggregate delay criterion, the "one-step" predicted density of the state is shown to be a sufficient statistic for the control. Finally, the priority assignment problem is studied when the rate parameters are assumed constant and *unknown*. Here a certainty equivalence, adaptive control scheme is analyzed and its convergence properties discussed.

The text is organized as follows. The complete observation problem is studied in Chapter 2. The μc-rule for the discrete-time, priority assignment problem is shown to be optimal for the infinite capacity system. Also, the value function of the optimal cost is shown to satisfy a monotonic property in its arguments. The finite capacity system is analyzed via numerical results. In Chapter 3, the partial observation stochastic control problem is considered. By a simple application of Bayes rule and dynamic programming, it is shown that the optimal control satisfies the separation principle. Moreover the optimal value function is shown to be linear in the sufficient statistic. The practical implications of these results are discussed. In Chapter 4, the adaptive control problem is presented. For a persistent excitation of the server, the adaptive control scheme is shown to converge (a.s.) to the optimal control strategy achieved if the true parameters were known. Finally in each chapter, an introduction is provided to survey the relevant results in the literature.

2. STOCHASTIC CONTROL OF TWO COMPLETELY OBSERVED COMPETING QUEUES

2.1 Introduction

Dynamic control of queueing systems is a subject of great interest presently, due to the potential applications in performance evaluation and design of computer and communication networks and systems. Classical queueing theory applied to such control problems has generally been directed towards deriving various limiting properties of such quantities as the queue length or waiting time distributions, under appropriate stability conditions. Extensive bibliographies on queueing control models and strategies can be found in Crabill et al [1], Sobel [2], Stidham and Prabhu [3]. Recently though, the analysis has matured to extend the results of these static or steady-state models and strategies to allocation schemes that are dynamic. By dynamic strategies, we mean a policy which at each time, t utilizes the information available up to time t. Examples of such dynamic control of queueing systems can be found in studies by Hajek et al [4], Bremaud [5], Baras et al [6], Lin et al [7] and Rosberg et al [8]. We follow this last reference in some of the results presented here.

The present chapter analyzes a simple stochastic control problem for two competing queues. We consider the problem of allocating the resources of a single server to serve requests from two parallel competing queues.

The problem is formulated in discrete time with the arrival and departure processes modelled by Bernoulli streams. The arrival rate and the service rate at each queueing station are allowed to depend on the queue size and the control value. At each service completion time, the controller decides which of the two queues to serve next. The controller observes both the arrival and departure processes. The instantaneous cost is linear in the waiting times of the two queues and three different performance criteria are considered. Thus, we have a stochastic control problem with complete observations.

One can view this problem as a dynamic priority assignment problem in a single-server queueing system. Within this context, several related results have been obtained [9] - [14]. Cox and Smith [9, p. 77] considered priority assignment in a single server queue with k classes for arrivals modelled by independent Poisson processes with rates $\lambda_1, \lambda_2, \ldots, \lambda_k$. For each class i $(1 \le i \le k)$ customer, the service times were modelled as independent random variables with probability distribution, $B_i(\cdot)$ and a waiting cost, c_i was incurred per unit time. The performance objective was to minimize, over all admissible open loop policies (i.e. strategies that did not incorporate current information such as queue size), the expected long-run average cost per unit time criterion.

They showed that the optimal open loop strategy was the so called "μc rule." Specifically if

$$\mu_i = 1/\nu_i$$

where ν_i is the average service time for the i^{th} class, then the optimal priority assignment ranks classes according to the products $\{\mu_i c_i;\ i=1,2,\ldots,k\}$ such that the classes with higher μc values are given higher priority. Rykov and Lembert [10] and Kakalik [11] generalized this result by proving that among all feedback control laws (i.e. the controller knows the queue size at each decision epoch), the optimal policy is the same as the μc rule. In other words, given the additional information of the past history of the queue size, the optimal dynamic priority assignment reduced to the simple static policy. This result is not surprising. Heuristically, one can argue for an ergodic system that since the instantaneous costs are uniformly weighted over the infinite horizon, the controller attempts to minimize his immediate cost, for which he has direct control and disregards the influence of the future cost since they are uniformly weighted anyway. In the analysis [9] - [11], the problem was formulated in continuous time with general service time distributions and unbounded queues.

The aforementioned queueing system [9] under an infinite horizon discount performance criterion was investigated by Harrison [12], [13]. For an arbitrary service time distribution and a linear cost structure, he maximized the expected net present value of service rewards received minus holding cost incurred over an infinite planning horizon. In other words, upon entrance into the system a customer incurs an entrance cost equal to the total discount holding cost if he remained in the system forever. At his departure epoch, the system is rewarded with that portion of the entrance cost not incurred (corresponding to his departure from the system). Harrison showed that there exists a special type of priority assignment, called a modified static policy, which is optimal among all feedback policies (the controller has knowledge of the queues sizes). In particular once the customer classes are appropriately ranked, there exists an integer $k^*(0 \leq k^* \leq k)$ such that

(i) for customers in classes 1 through k^*, the "μc priority rule" holds

(ii) for customers in classes k^*+1 through k, serve is never provided.

The explicit algorithm for computing the class ordering and the threshold value, k^* is given in [13]. An important feature of this modified static policy is that in the case of two customer classes, the optimal dynamic strategy reduces to the μc-rule. However for queueing systems with three or more classes, the determination of the lower priorities depends on the arrival rates of the higher ranked classes [12]. In the analysis [12] -

[13], the problem was formulated in continuous time with unbounded queues.

A related problem for bounded queues was analyzed by Mova and Ponamarenko [14] using Markov decision theory. They considered a multi-server queueing system with k classes of arrivals modelled by independent Poisson processes. The arrival times for each class and each server were modelled by identical, exponentially distributed random variables. The performance criterion was chosen to minimize the probability of losing an arrival of the first kind. In other words, a penalty of value c_i was incurred on the system when an arrival of the i^{th} class was rejected entry into the system. Due to the finite bound on the queue length, they demonstrated via numerical examples that the simple "μc rule" is not optimal. The equations characterizing the optimal policy showed that the optimal solution is a true feedback strategy in the sense that it depends on the current queue size and on the arrival rates of all classes. The optimal strategy was obtained by a variation of the simplex algorithm used in solving linear programming problems.

The queueing model considered here differs in two respects from the previously mentioned studies. First, the optimal priority assignment problem is formulated as a discrete-time stochastic control problem. This framework extends readily to the adaptive control problem presented later in the text. Second, our analysis considers the system with a finite capacity; specifically under the finite horizon criterion. The optimal policy, in general, depends on all the parameters of the queueing process. For the finite horizon problem, we demonstrate that this is the case. Our approach is via the solution of the Hamilton-Jacobi-Bellman equation, whose implementation is simplified.

This chapter is organized as follows. In Section 2.2, the mathematical assumptions of the optimal priority assignment problem are formulated. The optimality equations characterizing the solution of the finite horizon problem are presented in Section 2.3. Extension of these methods to the multi-class case of [9] - [13] is theoretically straightforward, but computationally burdensome. Our development simplifies the on-line solution for the optimal policy. In Section 2.4, the infinite horizon discounted problem is considered. Here the general results of the unbounded, finite horizon problem are extended by a limiting argument. To overcome the difficulty of an unbounded instantaneous cost, we use the results of Lippman [16]. Our proof of optimality is based on the convexity of the value function [8], [17]. The expected long-run, average cost per unit time criterion is discussed in Section 2.5. Our results are obtained by taking the limit of the finite **horizon**, T as $T \to \infty$. Finally, in Section 2.6, some numerical examples for the bounded queueing system are presented.

2.2 Problem Formulation and Notation

We consider two queues served by the same server in discrete time. The time is divided into equal length time slots (which are prespecified). During each time slot, arrivals and service completions can occur. We let $t = 0,1,2,\ldots$ be the index of these time slots. The situation is depicted in Figure 2.1 below.

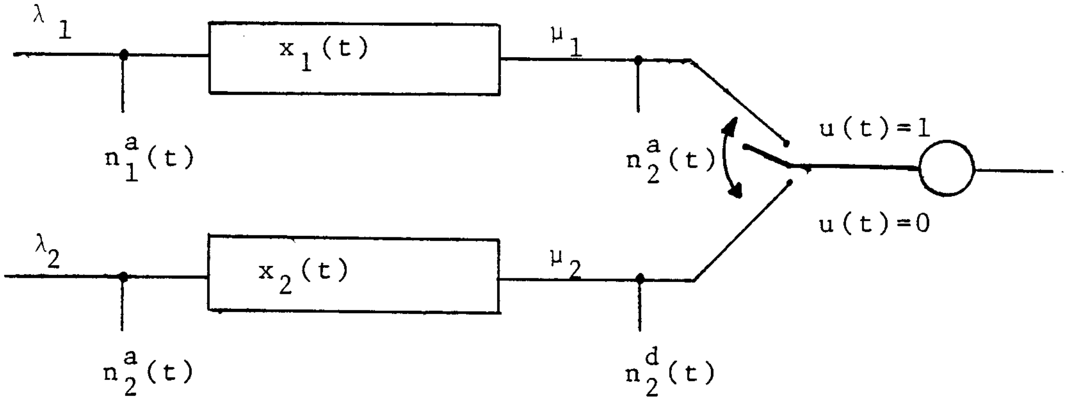


Figure 2.1. The server time allocation problem.

Customers arrive into queues 1 and 2 according to two independent Bernoulli streams with constant rates λ_1, λ_2 respectively. Thus if we let $\{n_i^a(\cdot);\ i = 1,2\}$ denote the two arrival processes, it is clear that they are discrete time 0-1 point processes:

$$n_i^a(t) = \begin{cases} 1, & \text{if an arrival occurs in the } t^{th} \text{ time slot of queue } i \\ 0, & \text{otherwise} \end{cases} \tag{2.2.1}$$

Our convention is that the t^{th} time slot is the half open interval $[t-1, t)$, where the length of each slot is assumed to be unity. In standard nomenclature [18], our assumptions imply that the arrival rates are

$$\lambda_i = \Pr\{ n_i^a(t) = 1 \mid \begin{array}{l}\text{past histories of}\\ \text{all processes}\end{array} \} \qquad ; i = 1,2$$

$$= \Pr\{n_i^a(t) = 1\} \qquad ; \ i = 1,2, \ . \qquad (2.2.2)$$

The two queues compete for the services of a single server. When the server serves queue i, i = 1,2, service completions follow a Bernoulli stream with constant rate $\{\mu_i; \ i = 1,2\}$. These assumptions imply that during each time slot at most one arrival and one service can occur, when each queue operates alone.

Let $x_i(t)$ be the number of customers in queue i(i = 1,2) at the end of the t^{th} time slot, the customer in service (if any) included. The control is used to allocate server time to queue 1 or to queue 2. Namely 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. If we let $\{n_i^d(\cdot); \ i = 1,2\}$ denote the two departure processes, their rates are given by (see [18] for some standard definitions)

$$\Pr\{n_i^d(t) = 1 \mid \begin{array}{l}\text{past histories of } x_1, \ x_2, \ n_1^a, \ n_2^a,\\ n_1^d, \ n_2^d, \text{ up to time } (t-1), \text{ and the}\\ \text{past history of } u, \text{ up to time } t\}\end{array}$$

$$= \Pr\{n_i^d(t) = 1 \mid x_i(t-1) = k, \ u(t)=v\} = \mu_i(t,k,v) \qquad ; \qquad i = 1,2 \ . \qquad (2.2.3)$$

Under our assumptions we have

$$\mu_1(t,k,v) = \begin{cases} \mu_1 \, v \,, & \text{if } k \neq 0 \\ 0 \,, & \text{if } k=0 \end{cases} \tag{2.2.4}$$

$$\mu_2(t,k,v) = \begin{cases} \mu_2 \, (1-v) \,, & \text{if } k \neq 0 \\ 0 \,, & \text{if } k=0. \end{cases} \tag{2.2.5}$$

We assume that both queues can grow without bound. This allows analytical treatment of the problem. When the queues are bounded, e.g. due to finite buffer size in computer/ communication systems, the methods used here lead to numerical treatment; analytical solutions have not been obtained to date. In the latter case if $\{N_i, i = 1,2\}$ are the maximum queue sizes for each queue, we have additional contraints on the arrival rates

$$\lambda_i(t,k,v) = \begin{cases} \lambda_i \,, & \text{if } k \neq 0, \text{ all } t, v, \\ 0 \,, & \text{if } k=N_i, \text{ all } t, v, \end{cases} \quad \text{for } i = 1,2 \tag{2.2.6}$$

The transition probabilities for each queue modelled as a Markov chain with countable state space over the set of nonnegative intergers have the form:

$$\left.\begin{aligned} P^i_{j,j}(v) &= \lambda_i \, \mu_i(j,v) + (1-\lambda_i)(1-\mu_i(j,v)) \\ P^i_{j,j+1}(v) &= \lambda_i \, (1-\mu_i(j,v)) \\ P^i_{j,j-1}(v) &= (1-\lambda_i) \, \mu_i(j,v) \\ P^i_{j,k} &= 0 \,, \text{ elsewhere} \qquad \text{for } i = 1,2 \end{aligned}\right\} \tag{2.2.7}$$

where we have suppressed the time argument, since it does not enter explicitly. In view of (2.2.4) (2.2.5), letting

$$\begin{aligned} b_1 &= \lambda_1\,(1-\mu_1)\ ,\quad b_2 = (1-\mu_2)\,\lambda_2 \\ d_1 &= \mu_1\,(1-\lambda_1)\ ,\quad d_2 = \mu_2\,(1-\lambda_2) \end{aligned} \qquad (2.2.8)$$

then (2.2.7) becomes in matrix form:

$$P^1(1) = \begin{bmatrix} 1-\lambda_1 & \lambda_1 & 0 & & 0 \\ d_1 & 1-b_1-d_1 & b_1 & & 0 \\ 0 & d_1 & 1-b_1-d_1 & b_1 & \\ & \ddots & \ddots & \ddots & \ddots \end{bmatrix} \equiv G_1 \qquad (2.2.9a)$$

$$P^1(0) = \begin{bmatrix} 1-\lambda_1 & \lambda_1 & 0 & 0 & 0 \\ 0 & 1-\lambda_1 & \lambda_1 & & 0 \\ 0 & 0 & 1-\lambda_1 & \lambda_1 & \\ & & \ddots & \ddots & \end{bmatrix} \equiv R_1 \qquad (2.2.9b)$$

$$P^2(1) = \begin{bmatrix} 1-\lambda_2 & \lambda_2 & 0 & 0 & 0 \\ 0 & 1-\lambda_2 & \lambda_2 & 0 & 0 \\ 0 & 0 & 1-\lambda_2 & \lambda_2 & \\ & \ddots & \ddots & \ddots & \ddots \end{bmatrix} \equiv R_2 \qquad (2.2.10a)$$

$$P^2(0) = \begin{bmatrix} 1-\lambda_2 & \lambda_2 & 0 & & 0 \\ d_2 & 1-b_2-d_2 & b_2 & 0 & 0 \\ & & & & \\ 0 & d_2 & 1-b_2-d_2 & b_2 & \\ & \ddots & \ddots & \ddots & \end{bmatrix} \equiv G_2 \tag{2.2.10b}$$

The transition probability matrix for the Markov chain representing both queues is given by

$$P(v) = P^1(v) \otimes P^2(v), \text{ for all } v \tag{2.2.11}$$

where ⊗ indicates matrix tensor product. It is straightforward to establish that for any value of the control variable v (i.e. 0 or 1), P(v) will not be a block diagonal matrix and therefore any state will communicate with any other. In other words, P(v) is irreducible [19, p. 232] for each value of v. We also observe that for each value of v, there are no absorbing states.

The controller decides the value of $u(\cdot)$ for the t^{th} slot at the end of the $(t-1)^{th}$ slot. The decision is based on past histories of control values, departure and arrival data up to the decision time (time slot by time slot). Therefore the controller knows the queue sizes at decision times. We shall assume that the parameters $\{\lambda_i, \mu_i; i = 1,2\}$ are known constants. Thus, we have a completely observed stochastic control problem. In subsequent chapters, we shall consider the case when the controller only observes the arrival process (Chapter 3) and when the parameters are unknown (Chapter 4).

At each decision time the controller must assign the value 1 or 0 to the control variable u(t) based on the following information:

$$\begin{aligned}&\{n_i^a(s)\ ;\ s = 0, 1, 2, \ldots, t-1\} \quad \text{for } i = 1,2\\ &\{n_i^d(s)\ ;\ s = 0, 1, 2, \ldots, t-1\} \quad \text{for } i = 1,2\\ &\{u(s)\ ;\ s = 0, 1, 2, \ldots, t-1\}\end{aligned} \tag{2.2.12}$$

Let

$$y^t = \{n_i^a(s),\ n_i^d(s);\ s = 0, 1, \ldots, t;\ i = 1,2\} \tag{2.2.13}$$

$$u^t = \{u(s);\ s = 1, 2, \ldots, t\}.$$

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

$$\gamma = (g_1, g_2, \ldots), \tag{2.2.14}$$

where

$$u(t) = g_t(y^{t-1}, u^{t-1}) \tag{2.2.15}$$

and each g_t takes values in {0, 1}. Note that at all times the controller knows the queue sizes, since

$$x_i(t) = n_i^a(t) - n_i^d(t) + x_i(t-1) \qquad ;\ i = 1, 2. \tag{2.2.16}$$

Service is assumed to be non-preemptive and server idling is not allowed; specifically

$$u(t) = \begin{cases} 1\ , & \text{if } x_1(t-1) \neq 0,\ x_2(t-1) = 0 \\ 0\ , & \text{if } x_1(t-1) = 0,\ x_2(t-1) \neq 1. \end{cases} \tag{2.2.17}$$

If both queues are empty at a decision time then either decision is acceptable.

In terms of this model, the performance criteria of interest are the following:

(A) Finite Horizon Problem with Expected Total Cost

$$J_f^\gamma = E[\sum_{t=0}^{T} c(x(t), u(t))] \tag{2.2.18}$$

where $x(t) = (x_1(t), x_2(t))$ and T denotes the finite time horizon. In the present discussion, the instantaneous cost, $c(x(\cdot), u(\cdot))$ is linear in the state, $x(\cdot)$ and has the form:

$$c(x(t), u(t)) = c_1x_1(t) + c_2x_2(t) \tag{2.2.19}$$

where c_1, c_2 are positive constants modelling the relative weight the controller attaches to delays in queue 1 versus those occurring in queue 2. Indeed c_i can be interpreted as the cost for a customer waiting for the duration of one time slot in queue i.

(B) Infinite Horizon Problem with Discounted Cost

$$J_{d,\beta}^\gamma = E[\sum_{t=0}^{\infty} \beta^t c(x(t), u(t))] \tag{2.2.20}$$

where $\beta \varepsilon [0,1)$ is the discount factor and the instantaneous cost is given in (2.2.19).

(C) Expected Long-Run Average Cost

$$J_a^\gamma = \lim_{T\to\infty} \inf \frac{1}{T} E[\sum_{t=0}^{T-1} c(x(t), u(t))] \tag{2.2.21}$$

Our objective is to derive the optimal strategies minimizing criteria (A), (B) and (C) for the queueing model introduced in (2.2.1) - (2.2.17). The superscript, γ in (2.2.18), (2.2.20) and (2.2.21) refers to the control strategy as defined in (2.2.14). For the cost considered here (2.2.19), the optimal strategy under both infinite horizon criteria turn out to be stationary, i.e., $\gamma = \{g, g, \ldots , g\}$, where g depends on the parameter values and the queue size. Clearly $J^{\gamma}_{d,\beta}$ is related to long term discounted average aggregate delay, while J^{γ}_{a} to long term average cost per unit time aggregate delay. Note in the case of unbounded queues, the instantaneous cost of (2.2.19) is unbounded.

2.3 Finite Horizon Stochastic Optimal Control General Results

In this section, the finite horizon average aggregate delay (2.2.18), (2.2.19) problem for the queueing system (2.2.1) - (2.2.17) is considered. This priority assignment problem is formulated as a stochastic control problem with complete observations. First, we review briefly the general dynamic programming theorems for the completely observed, finite horizon stochastic control problem. For discrete-time systems, these results can be proved by relatively straightforward arguments. In general, the optimal control policy depends parametrically on the dynamical system. Second, we apply the particular results for the two competing queue problem. We obtain explicit solutions for the finite time expected aggregate delay problem for bounded and unbounded queues. The implications of these results for practical applications are discussed.

Let the state space be an n-dimensional, Euclidean vector space, χ with state dynamics satisfying:

$$x(t+1) = \varphi_t(x(t),\ u(t),\ w(t+1));\ x(0) = x_0 \qquad (2.3.1)$$

$$\text{for } t = 0,1,2,\ \ldots\ ,\ T$$

where T is the finite time horizon, $u(t)\varepsilon U$ are the control values and $w(t)\varepsilon D$ are independent random variables with a known distribution. The function $\varphi_t(\cdot,\cdot,\cdot)$ is assumed to be known. The random disturbances $\{w(t)\}$ are characterized by a probability measure $p_t(\cdot|x(t),\ u(t))$ defined on a collection of events in D. This probability measure may depend explicitly on x(t) and u(t), but not on values of prior disturbances. An underlying probability triple (Ω,F,P) which carries x_0 and the $\{w(t)\}$ processes is assumed to be given. Furthermore, we shall assume, as is standard [20], that the disturbance space D is a countable set. The control space U is a convex, compact subset of R^m. The state space for the complete observation problem is a countable subset of R^n; for bounded queues χ is finite while for unbounded queues, χ is infinite. We note in passing that the partial observation problem (c.f. Chapter 3) can be reduced to the complete observation problem by an appropriate redefinition of χ.

For a control policy, $\gamma\varepsilon\Gamma$ the finite horizon performance criterion is denoted by

$$J_f^\gamma(x_0) = E[\sum_{t=0}^{T-1} c(t,x^\gamma(t),\ u^\gamma(t)) + c(T,\ x^\gamma(T))] \qquad (2.3.2)$$

where c(t,x,u) and c(T,x) denote respectively the instantaneous and terminal costs. The expectation above is, of course, taken with respect to the given probability distribution, $P(\cdot|x,u)$ which depends on x, u. The superscript γ in x, u indicates the state and control trajectories induced by the policy γ. The set of admissible control policies Γ is defined in (2.2.12) - (2.2.15), (2.2.17). The problem is to find $\gamma^{*}\varepsilon\Gamma$ such that

$$J_f^{\gamma^*}(x_0) = \inf \{J_f^{\gamma}(x_0):\gamma\varepsilon\Gamma\} \qquad \text{for all } x_0\varepsilon\chi \qquad (2.3.3)$$

The corresponding policy, γ^{*} is called optimal. It will be verified later that J_f^{γ} is well-defined. For the finite horizon problem, it is well-known that the optimal policy may not be stationary [20].

To solve the optimization problem (2.3.3), we resort to the well-known imbedding procedure of dynamic programming. Let $V_k(x,\gamma)$ denote the expected cost to go from t = k to T, given x(k) = x when the control law $\gamma\varepsilon\Gamma$ is followed; specifically

$$V_k(x,\gamma) = E[\sum_{t=k}^{T} c(t,x^{\gamma}(t), u^{\gamma}(t)) + c(T,x^{\gamma}(T))|x^{\gamma}(k)=x] \qquad (2.3.4)$$

$$\text{for } k = T-1, T-2, \ldots , 0$$

with terminal condition

$$V_T(x,\gamma) = c(T,x^{\gamma}(T)) \qquad (2.3.5)$$

The problem then is to select a control law for which $V_0(x,\gamma)$ is a minimum. Since for any control law, $V_k(x,\gamma)$ satisfies (2.3.4), (2.3.5) it is natural to ask whether one can compute a control law which is optimal. We have the following sufficient condition for optimality [20, p. 50]:

Theorem 2.3.1. If there exists a control law $\gamma^* \varepsilon \Gamma$ such that

(a) $V_k(x,\gamma^*)$ satisfies (2.3.4) and (2.3.5) for $\gamma^* \varepsilon \Gamma$, and

(b) For all t,x in the domain of interest and for all $\gamma \varepsilon \Gamma$

$$V_k(x,\gamma^*)$$

$$= \inf_{u \varepsilon U} E[c(k,x^{\gamma^*}(k),u^{\gamma^*}(k))+V_{k+1}(\varphi_k(x^{\gamma^*}(k),u^{\gamma^*}(k),w^{\gamma^*}(k+1)),\gamma^*)]$$

$$\leq E[c(k,x^{\gamma}(k),u^{\gamma}(k))+V_{k+1}(\varphi_k(x^{\gamma}(k),u^{\gamma}(k),w^{\gamma}(k+1)),\gamma)] \quad (2.3.6)$$

for all $k = 0,1,2,\ldots,T-1$

then γ^* is an optimal control law. Furthermore,

$$J_f^{\gamma^*}(x) = V_0(x,\gamma^*) \leq V_0(x,\gamma) \quad (2.3.7)$$

for all $x \varepsilon X$ and $\gamma \varepsilon \Gamma$.

The dynamic programming technique decomposes the problem (2.3.3) into a sequence of simpler minimization problems (2.3.6) that are carried out over the control space U rather than over a space of functions. The value functions in (2.3.4) minimize the "cost to go" from time k to T and are computed recursively, backwards in time starting, at time T and ending at time 0. Although Theorem 2.3.1 characterizes the optimal policy, explicit solutions of $V_k(\cdot,\cdot,\cdot)$ are generally not possible due to the "curse of dimensionality."

Application to the Two Competing Queue Problem

The finite horizon formulation is now applied to the two competing queues problem of Section 2.2. The state dynamics and instantaneous cost are given, respectively, by (2.2.16) and (2.2.19)

$$x_i(t+1) = x_i(t) + n_i^a(t+1) - n_i^d(t+1) \quad ; \quad i = 1,2 \qquad (2.3.8)$$

$$c(t,x(t),u(t)) = c_1x_1(t) + c_2x_2(t) = c^Tx(t) \qquad (2.3.9)$$

where

$$c^T = (c_1,c_2) \text{ and } x(t) = (x_1(t),x_2(t))^T$$

More precisely for $x(t) = (i_1,i_2)$, (2.3.8) has the form:

$$x(t+1) = \left\{ \begin{array}{lll} & \underline{u(t) = 1} & \underline{u(t) = 0} \\ A_1x(t) = (i_1+1,i_2+1) & \text{Prob } b_1\lambda_2 & b_2\lambda_1 \\ A_2x(t) = (i_1+1,i_2) & b_1(1-\lambda_2) & (1-d_2-b_2)\lambda_1 \\ A_3x(t) = (i_1,i_2+1) & (1-d_1-b_1)\lambda_2 & b_2(1-\lambda_1) \\ D_1x(t) = (i_1+1,i_2-1)^+ & 0 & d_2\lambda_1 \\ D_2x(t) = (i_1,i_2-1)^+ & 0 & d_2(1-\lambda_1) \\ D_3x(t) = (i_1-1,i_2+1)^+ & d_1\lambda_2 & 0 \\ D_4x(t) = (i_1-1,i_2)^+ & d_1(1-\lambda_2) & 0 \\ x(t) = (i_1,i_2) & (1-d_1-b_1)(1-\lambda_2) & (1-d_2-b_2)(1-\lambda_1) \end{array} \right. \qquad (2.3.10)$$

where b_i,d_i; $i = 1,2$ are defined in (2.2.8),

$$(i_1,i_2-1)^+ = (i_1,i_2-1 \vee 0) \text{ and } (i_1-1,i_2)^+ = (i_1-1 \vee 0,\ i_2)$$

For a policy $\gamma \varepsilon \Gamma$, the cost is the average aggregate delay

$$J_f^\gamma(x_0) = E[\sum_{t=0}^{T} c^Tx^\gamma(t)] \qquad . \qquad (2.3.11)$$

The control space for the single server queue is $U = \{0,1\}$. The state space for the unbounded queueing system is the Cartesian product, $X = ZxZ$ where Z is the set of positive integers. For a bounded queueing system, the arrival operators A_1, A_2 and A_3 occur with probability zero at the boundary states (see equation (2.2.6)) and the state space, $\mathcal{X} = Z_1 x Z_2$ where

$$Z_i = \{0,1,2, \ldots , N_i\} \qquad \text{for } i = 1,2.$$

To simplify notation, we adopt the convention, for the remainder of this section, to denote the optimal value functions of Theorem 2.3.1 as follows:

$$V_k(x) = V_{T-k}(x,\gamma^*) \quad \text{for } k = 0,1,2, \ldots , T \tag{2.3.12}$$

In particular, we have dropped the designation of the optimal policy $\gamma^* \varepsilon \Gamma$ and have reversed the indexing argument. By combining (2.3.8) - (2.3.12), the dynamic programming recursion of Theorem 2.3.1 reduces to

$$V_0(x) = c^T x$$

$$V_{k+1}(x) = c^T x + \{T^0 V_k(x) \wedge T^1 V_k(x)\} \tag{2.3.13}$$

$$\text{for all } x \varepsilon \mathcal{X},\ k = 0,1,2, \ldots , T$$

where

$$T^0 V(x) = b_2\lambda_1 \ V(A_1x) + b_2(1-\lambda_1) \ V(A_3x) + (1-b_2-d_2)\lambda_1 \ V(A_2x)$$
$$+ d_2\lambda_1 \ V(D_1x) + d_2(1-\lambda_1) \ V(D_2x) + (1-b_2-d_2)(1-\lambda_1) \ V(x) \tag{2.3.14}$$

and

$$T^1 V(x) = b_1\lambda_2 \ V(A_1x) + b_1(1-\lambda_2) \ V(A_2x) + (1-b_1-d_1)\lambda_2 \ V(A_3x)$$
$$+ d_1\lambda_2 \ V(D_3x) + d_1(1-\lambda_2) \ V(D_4x) + (1-b_1-d_1)(1-\lambda_2) V(x) \tag{2.3.15}$$

Remark 2.3.1. In (2.3.13), we have assumed the existence of the optimal policy, $\gamma^* \varepsilon \Gamma$ so that the infimum of (2.3.6) is replaced by the minimum [20].

Remark 2.3.2. For $x = (0,0)$ in (2.3.13), it follows

$$V_0(0,0) = 0$$

$$V_{k+1}(0,0) = \{T^0V_k(0,0) \wedge T^1V_k(0,0)\}$$

where in this case $b_i = \lambda_i$, $d_i = 0$; $i = 1,2$ in (2.3.14) and (2.3.15). In other words

$$\begin{aligned} T^0V_k(0,0) &= \lambda_1\lambda_2V_k(A_1x) + \lambda_1(1-\lambda_2)V_k(A_3x) + (1-\lambda_2)\lambda_1V(A_2x) \\ &= T^1V_k(0,0) \qquad (2.3.16) \end{aligned}$$

Consequently for $x = (0,0)$, the control value is arbitrary.

The finite horizon problem is considered for both bounded and unbounded queues. For an unbounded system, we prove that the optimal policy is the μc-rule, i.e. for $\mu_2c_2 > \mu_1c_1$

$$u(i_1,i_2) = \begin{cases} 1 & \text{if } i_2 = 0,\ i_1 \neq 0 \\ 0 & \text{if } i_2 \neq 0 \end{cases} \qquad (2.3.17)$$

and for $\mu_1c_1 > \mu_2c_2$

$$u(i_1,i_2) = \begin{cases} 1 & \text{if } i_1 \neq 0 \\ 0 & \text{if } i_1 = 0,\ i_2 \neq 0 \end{cases} \qquad (2.3.18)$$

When the queues are bounded, the methods used here lead to a numerical treatment; analytical solutions have not been obtained to date.

To prove (2.3.17) (2.3.18), we proceed with the following sequences of lemmas:

Lemma 2.3.2. For all $x = (i_1, i_2) \varepsilon \chi$, and arbitrary functions $f(\cdot), g(\cdot)$ such that $f(x) > g(x)$ then

$$T^i f(x) > T^i g(x) \quad : \quad i = 0,1 \tag*{(2.3.19(}$$

Proof: (Without loss of generality, assume i = 1). Suppose the converse, i.e. $T^1 g(x) \le T^1 f(x)$. Then by (2.3.15) we have

$$\begin{aligned} T^1 g(x) - T^1 f(x) = {} & b_1 \lambda_2 [g(A_1 x) - f(A_2 x)] \\ & + b_1 (1-\lambda_2)[g(A_2 x) - f(A_2 x)] \\ & + (1-b_1-d_1)\lambda_2 [g(A_3 x) - f(A_3 x)] \\ & + d_1 \lambda_2 [g(D_3 x) - f(D_3 x)] \\ & + d_1 (1-\lambda_2)[g(D_4 x) - f(D_4 x)] \\ & + (1-b_1-d_1)(1-\lambda_2)[g(x) - f(x)] < 0 \end{aligned}$$

since each term involving λ_i, b_i and d_i's are non-negative and by hypothesis

$$f(x) > g(x) \quad \text{for all } x \varepsilon \chi$$

Contradiction. Therefore $T^1 f(x) > T^1 g(x)$ QED.

Lemma 2.3.3. For all $x = (i_1, i_2) \varepsilon \chi$ such that $(i_1, i_2) \neq (0,0)$

$$T^0 T^1 V_k(x) = T^1 T^0 V_k(x) \quad \text{for all } k = 0,1,2, \ldots , T \tag*{(2.3.20)}$$

Proof: We consider (2.3.13) in vector form by defining

$$\underline{V}_k = \{V_k(i_1, i_2)\} \tag*{(2.3.21)}$$

$$e = (1,1,1, \ldots , 1, \ldots)^T \tag*{(2.3.22)}$$

$$\nu = (0,1,2, \ldots , n, \ldots)^T \tag*{(2.3.23)}$$

The optimality condition then becomes

$$\underline{V}_0 = c_1(\nu\otimes e) + c_2(e\otimes\nu) \tag{2.3.24}$$

$$\underline{V}_{k+1} = c_1(\nu\otimes e) + c_2(e\otimes\nu) + \{P(0)\underline{V}_k \wedge P(1)\underline{V}_k\} \tag{2.3.25}$$

$$\text{for } k = 0,1,2, \ldots, T-1$$

where $P(v); v = 0,1$ is defined in (2.2.8) - (2.2.11) and $\otimes$ indicates matrix tensor product. Note that we have ordered the vector value functions $\{\underline{V}_k\}$ according to the sequence 00,01,02, ... , 10,11,12, ... , 20,21,22, ...

To show $T^0T^1V_k(x) = T^1T^0V_k(x)$ we need to show equivalently

$$P(0)P(1)\underline{V}_k = P(1)P(0)\underline{V}_k \tag{2.3.26}$$

By (2.2.9) - (2.2.11), we have

$$\begin{aligned} P(0)P(1) &= (R_1\otimes G_2) \cdot (G_1\otimes R_2) \\ &= (R_1G_1) \otimes (G_2R_2) \end{aligned}$$

where the second equality follows from properties of tensor products [21, p. 228]. Similarly

$$\begin{aligned} P(1)P(0) &= (G_1\otimes R_2) \cdot (R_1\otimes G_2) \\ &= (G_1R_1) \otimes (R_2G_2) \end{aligned}$$

Therefore to show (2.3.26), one need only show that except for the first row (recall $(i_1,i_2)\neq(0,0)$).

$$R_iG_i = G_iR_i \qquad ; i = 1,2 \tag{2.3.27}$$

By simple matrix multiplication, it is obvious that (2.3.27) holds. In particular from (2.2.9) we have

$$R_1G_1 = \begin{bmatrix} 1-\lambda_1 & \lambda_1 & 0 & 0 \\ 0 & 1-\lambda_1 & \lambda_1 & 0 \\ 0 & 0 & 1-\lambda_1 & \lambda_1 \\ & & \ddots & \ddots \end{bmatrix} \begin{bmatrix} 1-\lambda_1 & \lambda_1 & 0 & 0 \\ d_1 & 1-b_1-d_1 & b_1 & 0 \\ 0 & d_1 & 1-b_1-d_1 & b_1 \\ & \ddots & \ddots & \ddots \end{bmatrix}$$

$$= \begin{bmatrix} (1-\lambda_1)^2+d_1\lambda_1 & [\lambda_1(1-\lambda_1)+\lambda_1(1-d_1-b_1)] & b_1\lambda_1 & 0 \\ d_1(1-\lambda_1) & [\lambda_1 d_1+(1-\lambda_1)(1-b_1-d_1)] & [b_1(1-\lambda_1)+\lambda_1(1-b_1-d_1)] & b_1\lambda_1 \\ 0 & d_1(1-\lambda_1) & [\lambda_1 d_1+(1-\lambda_1)(1-b_1-d_1)] & \ddots \\ \vdots & 0 & \ddots & \ddots \\ & \vdots & & \end{bmatrix}$$

Also

$$G_1R_1 = \begin{bmatrix} 1-\lambda_1 & \lambda_1 & 0 & 0 \\ d_1 & 1-b_1-d_1 & b_1 & 0 \\ 0 & d_1 & 1-b_1-d_1 & b_1 \\ & \ddots & \ddots & \ddots \end{bmatrix} \begin{bmatrix} 1-\lambda_1 & \lambda_1 & 0 & 0 \\ 0 & 1-\lambda_1 & \lambda_1 & 0 \\ 0 & 0 & 1-\lambda_1 & \lambda_1 \\ & & \ddots & \ddots \end{bmatrix}$$

$$= \begin{bmatrix} (1-\lambda_1)^2 & 2\lambda_1(1-\lambda_1) & \lambda_1^2 & \\ d_1(1-\lambda_1) & [\lambda_1 d_1+(1-\lambda_1)(1-b_1-d_1)] & [b_1(1-\lambda_1)+\lambda_1(1-b_1-d_1)] & b_1\lambda_1 \\ 0 & d_1(1-\lambda_1) & [\lambda_1 d_1+(1-\lambda_1)(1-b_1-d_1)] & \\ \vdots & 0 & \ddots & \ddots \\ & \vdots & & \end{bmatrix}$$

so that it follows that except for the first row

$$R_1G_1 = G_1R_1$$

Similarly one can show that except for the first row

$$R_2G_2 = G_2R_2 \qquad \text{QED}$$

Lemma 2.3.4. For each $k = 0,1,2,\ldots,T$ if $\mu_2c_2>\mu_1c_1$ and $c_1,c_2>0$ then

(a) $V_k(A_2x)>V_k(x)$, $V_k(A_3x)>V_k(x)$ for all $x\varepsilon\chi$ (2.3.28)

(b) $T^1V_k(x)>T^0V_k(x)$ for all $x = (i_1,i_2)\varepsilon\chi$; $i_2\neq 0$ (2.3.29)

(c) $T^0V_k(x)>T^1V_k(x)$ for all $x = (i_1,0)$; $i_1\neq 0$ (2.3.30)

Proof: By mathematical induction. By (2.3.13) - (2.3.15) for k=0, we have

$$V_0(x) = c^Tx \quad , \quad V_0(A_2x) = c^Tx + c_1, \; V_0(A_3x) = c^Tx+c_2$$

$$T^1V_0(x) = \begin{cases} c^Tx+c_1\lambda_1+c_2\lambda_2 - \mu_1c_1 & ; \quad i_1\neq 0 \\ c^Tx+c_1\lambda_1+c_2\lambda_2 & ; \quad i_1=0 \end{cases} \tag{2.3.31}$$

$$T^0V_0(x) = \begin{cases} c^Tx+c_1\lambda_1+c_2\lambda_2-\mu_2c_2 & ; \quad i_2\neq 0 \\ c^Tx+c_1\lambda_1+c_2\lambda_2 & ; \quad i_2=0 \end{cases} \tag{2.3.32}$$

Clearly (a) - (c) hold for k = 0. Suppose (a) - (c) hold for k. We shall show they hold for k + 1.

Part (a)

For $x = (i_1,i_2)\neq(0,0)$, we have by (2.3.13) and (2.3.29)

$$V_{k+1}(A_2x) = c^T(A_2x) + T^0V_k(A_2x)$$

$$V_{k+1}(x) = c^Tx + T^0V_k(x)$$

By Lemma 2.3.2 and (2.3.28)

$$T^0V_k(A_2x)>T^0V_k(x) \tag{2.3.33}$$

so that it follows

$$V_{k+1}(A_2x)>V_{k+1}(x) \qquad \text{for } x = (i_1,i_2)\neq(0,0)$$

For $x = (0,i_2)$; $i_2\neq 0$ we have by (2.3.13), (2.3.29) and (2.3.30)

$$V_{k+1}(A_2x) = c^T(A_2x) + T^0V_k(A_2x)$$

$$V_{k+1}(x) = c^Tx + T^0V_k(x)$$

Again by Lemma 2.3.2 and (2.3.28), equation (2.3.33) holds so that

$$V_{k+1}(A_2x)>V_{k+1}(x) \qquad \text{for } x = (0,i_2);\ i_2\neq 0$$

For $x = (i_1,0)$; $i_1\neq 0$ we have by (2.3.13), (2.3.29) and (2.3.30)

$$V_{k+1}(A_2x) = c^T(A_2x) + T^0V_k(A_2x)$$

$$V_{k+1}(x) = c^Tx + T^1V_k(x)<c^Tx + T^0V_k(x)$$

where the inequality follows from (2.3.30). Again equation (2.3.33) holds so that

$$V_{k+1}(A_2x)>V_{k+1}(x) \text{ for } x = (i_1,0);\ i_1\neq 0$$

For $x = (0,0)$, we have by (2.3.16)

$$V_{k+1}(A_2x) - V_{k+1}(x) = c^T(A_2x) - c^Tx>0$$

Hence we have shown

$$V_{k+1}(A_2x)>V_{k+1}(x) \qquad \text{for all } x\varepsilon X$$

By similar arguments, one can show

$$V_{k+1}(A_3x)>V_{k+1}(x) \qquad \text{for all } x\varepsilon X$$

Part (b)

For $x\neq(0,0)$, we have by (2.3.13) and (2.3.29)

$$\begin{aligned} V_{k+1}(x) &= c^Tx + \{T^0V_k(x)\wedge T^1V_k(x)\} \\ &= c^Tx + T^0V_k(x) \end{aligned} \tag{2.3.34}$$

Hence,

$$\begin{aligned} T^1V_{k+1}(x) &= T^1(c^Tx) + T^1T^0V_k(x) \\ T^0V_{k+1}(x) &= T^0(c^Tx) + T^0T^0V_k(x) \\ &\le T^0(c^Tx) + T^0T^1V_k(x) \qquad \text{(by (2.3.34))} \\ &< T^1(c^Tx) + T^0T^1V_k(x) \qquad \text{(by (2.3.31), (2.3.32))} \\ &= T^1(c^Tx) + T^1T^0V_k(x) \qquad \text{(by Lemma 2.3.3)} \\ &= T^1V_{k+1}(x) \end{aligned}$$

so that it follows

$$T^1V_{k+1}(x)>T^0V_{k+1}(x) \qquad \text{for all } x = (i_1,i_2)\neq(0,0)$$

For $x = (0,i_2)$; $i_2\neq0$ we have by (2.3.13) - (2.3.15) with $b_1=\lambda_1$ and $d_1=0$

$$\begin{aligned} T^1V_{k+1}(x) &= \lambda_1\lambda_2[V_k(A_1x) - V_k(x)] + \lambda_1(1-\lambda_2)[V_k(A_2x)-V(x)] \\ &\quad +(1-\lambda_1)\lambda_2[V(A_3x) - V_k(x)] \\ T^0V_{k+1}(x) &= b_2\lambda_1[V_k(A_1x)-V_k(x)] + b_2(1-\lambda_1)[V_k(A_3x)-V_k(x)] \\ &\quad + (1-b_2-d_2)\lambda_1[V_k(A_2x)-V_k(x)] + d_2\lambda_1[V_k(D_1x)-V_k(x)] \\ &\quad + d_2(1-\lambda_1)[V_k(D_2x) - V_k(x)] \end{aligned}$$

By combining and simplifying, we have

$$T^1V_{k+1}(x) - T^0V_{k+1}(x)$$

$$= \lambda_1(\lambda_2-b_2)[V_k(A_1x)-V_k(x)]+\lambda_1[(1-\lambda_2)-(1-b_2)][V_k(A_2x)-V_k(x)]$$

$$+ (1-\lambda_1)(\lambda_2-b_2)[V_k(A_3x)-V_k(x)]$$

$$- d_2\lambda_1[V_k(D_1x)-V_k(A_2x)]-d_2(1-\lambda_1)[V_k(D_2x)-V_k(x)]$$

$$= \lambda_1(\lambda_2-b_2)[V_k(A_1x)-V_k(A_2x)]$$

$$+ (1-\lambda_1)(\lambda_2-b_2)[V_k(A_3x)-V_k(x)]$$

$$- d_2\lambda_1[V_k(D_1x)-V_k(A_2x)]-d_2(1-\lambda_1)[V_k(D_2x)-V_k(x)]$$

But λ_1, $(1-\lambda_1)$, $d_2>0$ and $\lambda_2>b_2 = \lambda_2(1-\mu_2)$. Also by (2.3.11) and (2.3.28)

$$[V_k(A_1x) - V_k(A_2x)]>0 \quad [V_k(A_3x) - V_k(x)]>0$$

$$[V_k(A_2x) - V_k(D_1x)]>0 \quad [V_k(D_2x) - V_k(x)]\leq 0$$

so that

$$T^1V_{k+1}(x)>T^0V_{k+1}(x) \qquad \text{for all } x = (0,i_2) \ ; \ i_2\neq 0$$

Hence, we have shown

$$T^1V_{k+1}(x)>T^0V_{k+1}(x) \qquad \text{for all } x = (i_1,i_2) \ ; \ i_2\neq 0$$

Part (c)

For $x = (i_1,0)$; $i_1\neq 0$ we have by (2.3.13) - (2.3.15) with $b_2 = \lambda_2$ and $d_2 = 0$

$$T^1V_{k+1}(x) = b_1\lambda_2[V_k(A_1x)-V_k(x)]+b_1(1-\lambda_2)[V_k(A_2x)-V_k(x)]$$

$$+ (1-b_1-d_1)\lambda_2[V_k(A_3x)-V_k(x)]+d_1\lambda_2[V_k(D_1x)-V_k(x)]$$

$$+ d_1(1-\lambda_2)[V_k(D_2x) - V_k(x)]$$

$$T^0V_{k+1}(x)=\lambda_1\lambda_2[V_k(A_1x)-V_k(x)]+\lambda_2(1-\lambda_1)[V_k(A_3x)-V_k(x)]$$

$$+ (1-\lambda_2)\lambda_1[V_k(A_2x) - V_k(x)]$$

By combining and simplifying, we have

$$T^0V_{k+1}(x) - T^1V_{k+1}(x)$$

$$= \lambda_2(\lambda_1-b_1)[V_k(A_1x)-V_k(x)]+\lambda_2[(1-\lambda_1)-(1-b_1)][V_k(A_3x)-V_k(x)]$$

$$+ (1 -\lambda_2)(\lambda_1-b_1)[V_k(A_2x) - V_k(x)]$$

$$- d_1\lambda_2[V_k(D_3x)-V_k(A_3x)] - d_1(1-\lambda_2)[V_k(D_4x)-V_k(x)]$$

$$= \lambda_2(\lambda_1-b_1)[V_k(A_1x) - V_k(A_3x)]$$

$$+ (1-\lambda_1)(\lambda_2-b_2)[V_k(A_2x) - V_k(x)]$$

$$- d_1\lambda_2[V_k(D_3x) - V_k(A_3x)] - d_1(1-\lambda_2)[V_k(D_4x) - V_k(x)]$$

Again λ_2, $(1-\lambda_2)$, $d_2>0$ and $\lambda_1>b_1 = \lambda_1(1-\mu_1)$. Also by (2.3.11) and (2.3.38)

$$[V_k(A_1x) - V_k(A_3x)]>0 \qquad [V_k(A_2x) - V_k(x)]>0$$

$$[V_k(A_3x) - V_k(D_3x)]>0 \qquad [V_k(D_4x) - V_k(x)]\leq 0$$

so that

$$T^0V_{k+1}(x)>T^1V_{k+1}(x) \text{ for all } x = (i_1,0) \; ; \; i_1\neq 0$$

The lemma now follows by induction. QED.

By the nature of the queueing system (2.2.1) - (2.2.17), we have the symmetric result:

<u>Lemma 2.3.5</u>. For each $k = 0,1,2,\ldots,T$ if $\mu_1c_1>\mu_2c_2$ and c_1, $c_2>0$ then

(a) $V_k(A_2x)>V(x)$, $V_k(A_3x)>V_k(x)$ for all $x\in\mathcal{X}$ (2.3.35)

(b) $T^0V_k(x)>T^1V_k(x)$ for all $x = (i_1,i_2)\in\mathcal{X}$; $i_1\neq 0$ (2.3.36)

(c) $T^1V_k(x)>T^0V_k(x)$ for all $x = (0,i_2)$; $i_2\neq 0$ (2.3.37)

Proof: Follows along the same lines as Lemma 2.3.4.

In addition to the above properties, since $c^Tx\geq 0$ we have

$$V_{k+1}(x)\geq V_k(x) \quad \text{for all } x\varepsilon X \qquad (2.3.38)$$

Moreover, the optimal cost is well-defined.

Lemma 2.3.6. For $x\varepsilon X$ and for the optimal policy $\gamma^*\varepsilon\Gamma$, $J_f^{\gamma^*}(x)<\infty$

Proof: Let $|x| = |i_1| + |i_2|$ and $|c| = \max(c_1,c_2) = c_1\wedge c_2$ For any initial state x and any policy, the state, x(t) at time t must satisfy

$$|x(t)|\leq|x| + t$$

Moreover

$$V_k(x) \leq \sum_{t=k}^{T} |c|(|x|+t)$$

$$\leq |c||x|(T+1-k) + T(T+1)/2 < \infty$$

Hence it follows from (2.3.12) and Theorem 2.3.1

$$J_f^{\gamma^*}(x) = V_0(x) <\infty$$ QED.

Theorem 2.3.7. There is an optimal stationary policy, $\gamma_f^*\varepsilon\Gamma$ for the unbounded queueing system (2.3.8), (2.3.10) under the finite horizon average aggregate delay criterion (2.3.9), (2.3.11). The optimal policy and cost, V_f^* are determined from the optimality equation (2.3.13). Specifically, let $g_f^*(.,.)$: ZxZ→U define the optimal policy, i.e.

$$\gamma_f^* = (g_f^*, g_f^*, g_f^*, \ldots g_f^*)$$

then $g_f^*(.,.)$ is the μc-rule given in (2.3.17), (2.3.18).

Proof: Without loss of generality assume $\mu_2 c_2 > \mu_1 c_1$. By Theorem 2.3.1, the optimal policy $\gamma^* \varepsilon \Gamma$ is that strategy which minimizes (2.3.6); more precisely the bracketed quantity in (2.3.13). By Lemma 2.3.4, we have

$$T^1 V_k(x) > T^0 V_k(x) \quad \text{for all } x = (i_1, i_2) \ ; \ i_2 \neq 0$$

$$T^0 V_k(x) > T^1 V_k(x) \quad \text{for all } x = (i_1, 0) \ ; \ i_1 \neq 0$$

and by (2.3.16)

$$T^1 V_k(0,0) = T^1 V_k(0,0)$$

for all k = 0,1,2,...,T. Hence

$$g_f^*(i_1, i_2) = \begin{cases} 1 & \text{if } i_1 \neq 0, \ i_2 = 0 \\ 0 & \text{if } i_2 \neq 0 \\ \text{arbitrary} & \text{if } i_1 = i_2 = 0 \end{cases} \qquad (2.3.40)$$

By Lemma 2.3.6, $V_0(x, \gamma^*)$ is well-defined. The stationary of the optimal policy follows from (2.3.13) and a consequence of Lemma 2.3.3. QED.

Remark 2.3.3. One can observe that the optimal strategy is almost open loop and independent of the arrival rates, λ_1 and λ_2. For the case of more than two classes, this latter property is no longer valid [12]. In the two class system, it easily follows that the μc-rule is optimal for time-varying and state dependent arrival rates. Consider the recursive equations (2.3.13) - (2.3.15) for k = 0 and

$x = (i_1,\ i_2) \neq (0,0)$

$$V_0(x) = c^T x$$

$$\begin{aligned} T^0 V_0(x) &= b_2(0,i_2)\lambda_1(0,i_1)[V_0(A_1x) - V_0(x)] \\ &+ b_2(0,i_2)\lambda_1(0,i_1)[V_0A_3x) - V_0(x)] \\ &+ (1-b_2(0,i_2)-d_2(0,i_2))\lambda_1(0,i_1)[V_0(A_2x)-V_0(x)] \\ &+ d_2(0,i_2)\lambda_1(0,i_1)[V_0(D_1x)-V_0(x)] \\ &= c^T x + c_1\lambda_1(0,i_1)+c_2\lambda_2(0,i_2)-c_1\mu_1 \end{aligned} \tag{2.3.41}$$

where from (2.2.2), (2.2.8)

$$\lambda_i(0,j_i) = \Pr\{n_i^a(0) = 1 | x_i(0)=j_i\} \quad ; \ i=1,2 \tag{2.3.42}$$

$$\begin{aligned} b_2(0,i_2) &= \lambda_2(0,i_2)(1-\mu_2) \\ d_2(0,i_2) &= \mu_2(1-\lambda_2(0,i_2)) \end{aligned} \tag{2.3.43}$$

Similarly it follows

$$T^0 V_0(0,i_2) = c^T x + c_1\lambda_1(0,0) + c_2\lambda_2(0,i_2) \tag{2.3.44}$$

$$T^1 V_0(i_1,i_2) = c^T x + c_1\lambda_1(0,i_1) + c_2\lambda_2(0,i_2) - c_2\mu_2 \tag{2.3.45}$$

$$T^1 V_0(i_1,0) = c^T x + c_1\lambda_1(0,i_1) + c_2\lambda_2(0,0) \tag{2.3.46}$$

Consequently, the properties (a) - (c) in Lemma 2.3.4 hold for the more general recursion (2.3.41) - (2.3.46) when $k = 0$. The arguments in Lemma 2.3.4 follow along the same lines for the rates given in (2.3.42), (2.3.43).

The discussion up to this point has dealt with the unbounded queueing system. We shall discuss for the remainder of this section the bounded system. In particular, the arrival rates satisfy (2.2.6) where N_i

equals the maximum queue size for the i^{th} queue. Moreover, the state space becomes $X = Z_1 \times Z_2$ where $Z_i = \{0,1,2,...,N_i\}$ for $i = 1,2$. When the queues are bounded, the optimality of the μc rule does not follow. First, consider the results of Lemma 2.3.3. For a bounded queueing system,

$$T^0 T^1 V_k(x) = T^1 T^0 V_k(x) \text{ for all } k=0,1,2,...,T \qquad (2.3.47)$$

for all $x = (i_1, i_2)$ in the interior of X; specifically

$$x \varepsilon \{1,2,...,N_1-1\} \times \{1,2,...,N_2-1\} \qquad (2.3.48)$$

In other words, the product matrices

$$R_i G_i = G_i R_i \qquad ; i = 1,2$$

are identical except for the first and last rows and the last columns. Second, these additional conditions on Lemma 2.3.3 imply a feedback on the control value. This feedback control law, with the methods presented here leads to a numerical treatment; analytical solutions for this case have not been obtained. Due to the linearity of the instantaneous cost (2.3.9) and binary control space $U = \{0,1\}$, the computation of the optimal strategy is quite simple. Furthermore, these off-line computations can be stored in an elementary way to facilitate the on-line implementation of the strategy. The optimal average delay (2.3.11) is shown to be piecewise linear in the state.

We proceed from Theorem 2.3.1 or more precisely (2.3.13) - (2.3.15) to have

$$V_0(i,j) = e_i^1 \, d_0^1 + e_j^2 \, d_0^2 \qquad (2.3.49)$$

where

$$d_0^i = c_i \nu_i \qquad ; \ i = 1,2 \tag{2.3.50}$$

$$\nu_i = (0,1,2,\ldots,N_i)^T \varepsilon R^{N_i+1} \qquad ; \ i = 1,2 \tag{2.3.51}$$

$$e_j^i = (0,0,\ldots,0,\underset{\substack{\uparrow \\ j^{th} \text{ position}}}{1},0,\ldots) \ \varepsilon R^{N_{i+1}} \qquad ; \ i = 1,2 \tag{2.3.52}$$

Next at k = 1, (2.3.13), (2.3.21) - (2.3.25) imply

$$V_1(i,j) = e_i^1 \, d_0^1 + e_j^2 \, d_0^2 + \min_{u \varepsilon U} \{(P^1(u) \otimes P^2(u))\underline{V}_0\}_{i,j}$$

$$= \min_{u \varepsilon U} \{e_i^1[I_1 + P^1(u)]d_0^1 + e_j^2[I_2 + P^2(u)]d_0^2\} \tag{2.3.53}$$

where I_i is an identity matrix of dimension $\{N_i; \ i = 1,2\}$ and $\{\cdot\}_{i,j}$ denotes the (i,j) element of the vector in brackets. In other words, the optimal control is a function of the state, $x = (i,j) \varepsilon \chi$. It can be described as follows. The set $\chi = Z_1 \times Z_2$ is separated into two disjoint subsets

$$\chi_1 = \{(i,j) \varepsilon \chi \text{ such that}$$

$$e_i^1[P^1(0) - P^1(1)]d_0^1 \geq e_j^2[P^2(1) - P^2(0)]d_0^2\} \tag{2.3.54}$$

$$\chi_0 = \text{complement of } \chi_1$$

We associate the index 1 with χ_1, the index 0 with χ_0 so that

$$u_1^* = \begin{cases} 1 & \text{on } \chi_1 \\ 0 & \text{on } \chi_0 \end{cases} \tag{2.3.55}$$

Let $a_1(\cdot,\cdot)$ be the function

$$a_1(i,j) = \begin{cases} 1 & \text{if } (i,j)\varepsilon X_1 \\ 0 & \text{if } (i,j)\varepsilon X_0 \end{cases} \tag{2.3.56}$$

and

$$d_1^1(i,j) = [I_1 + P^1(a_1(i,j))]\ d_0^1$$

$$(2.3.57)$$

$$d_2^1(i,j) = [I_2 + P^2(a_1(i,j))]\ d_0^2$$

It is now clear that

$$V_1(i,j) = e_i^1\ d_1(i,j) + e_j^2\ d_2(i,j) \tag{2.3.58}$$

and therefore, $V_1(\cdot,\cdot)$ is piecewise linear also. The general computation follows from the following lemma.

Lemma 2.3.8. Define the binary-valued functions $\{a_\ell;\ \ell=1,2,\ldots,T\}$ on $X= Z_1 \times Z_2$ and the column vectors $\{d_\ell^i;\ i = 1,2; \ell = 0,1,2,\ldots,T\}$ by the forward recursions

$$d_0^i = c_i \nu_i$$

$$d_\ell^i = d_0^i + P^i(a_\ell)\ d_{\ell-1}^i(a_{\ell-1}, a_{\ell-2},\ldots,a_1)$$

$$a_\ell(i,j) = \begin{cases} 1 & \text{if } e_i^1[P^1(0)-P^1(1)]d_{\ell-1}^1(a_{\ell-1},a_{\ell-2},\ldots,a_1) \\ & \quad \geq e_j^2[P^2(1)-P^2(0)]d_{\ell-1}^2(a_{\ell-1},a_{\ell-2},\ldots,a_1) \\ 0 & \text{otherwise} \end{cases}$$

for $i = 1,2$; $\ell = 1,2,\ldots,T$

Then for $k = 1,2,\ldots,T$ and $x = (i,j)\varepsilon X$

$$V_k(i,j) = e_i^1\ d_k^1(a_k,a_{k-1},\ldots,a_1) + e_j^2 d_k^2(a_k,a_{k-1},\ldots,a_1) \tag{2.3.59}$$

In other words, $V_k(\cdot,\cdot)$ is piecewise linear for each k.

Proof: By mathematical induction. It follows from (2.3.53) - (2.3.58) that (2.3.59) holds for k = 1. Let us assume the result holds for k. Then from (2.3.13) (2.3.21) - (2.3.25) after computations identical to (2.3.53), we have

$$V_{k+1}(i,j) = \min_{u\varepsilon U} \{e_i^1 d_0^1 + e_j^2 d_0^2$$

$$+ e_i^1 P^1(u) d_k^1 + e_j^2 P^2(u) d_k^2\}$$

It follows now by the definition of $a_{k+1}(\cdot,\cdot)$ and $d_{k+1}^1(\cdot,\cdot)$, $d_{k+1}^2(\cdot,\cdot)$ that (2.3.59) holds for k + 1.

QED.

Remark 2.3.4. The recursive forward computations proceed diagrammatically as follows:

$$\begin{bmatrix} d_0^1 \\ d_0^2 \\ a_1 \end{bmatrix} \longrightarrow \begin{bmatrix} d_1^2 \\ d_1^2 \\ a_2 \end{bmatrix} \longrightarrow \cdots \longrightarrow \begin{bmatrix} d_k^1 \\ d_k^2 \\ a_{k+1} \end{bmatrix} \longrightarrow \begin{bmatrix} d_T^1 \\ d_T^2 \\ - \end{bmatrix}$$

We also have established the corollary.

Corollary 2.3.9. The optimal control policy in feedback form, as a function of $x\varepsilon X$, is given by

$$g_f^*(k;i_1,i_2) = a_k(i_1,i_2) \qquad k = 1,2,\ldots,T$$

Combining now the results of Lemma 2.3.8, Corollary 2.3.9 and (2.3.7) of Theorem 2.3.1, we have established the following result for the bounded queueing system (2.3.8), (2.3.10).

Theorem 2.3.10. The optimal server time allocation strategy and expected aggregate delay for the finite horizon criterion (2.3.9), (2.3.11) are determined as follows. First the vectors $\{d_\ell^i;\ i=1,2;\ell=0,1,\ldots,T\}$ and binary-valued functions $\{a_\ell;\ell=1,2,\ldots,T\}$ are computed off-line and stored from Lemma 2.3.8. The queueing system propagates forward as described by (2.3.8), (2.3.10). The optimal strategy at time k is given by

$$g_f^*(k;i_1,i_2) = a_k(i_1,i_2) \ ;\ k = 1,2,\ldots,T \qquad (2.3.60)$$

The optimal average aggregate delay has the form

$$V_T(i_1,i_2) = e_i^1\ d_T^1(i_1,i_2) + e_j^2\ d_T^2(i_1,i_2) \qquad (2.3.61)$$

Note: Recall the notational convention introduced in (2.3.12). In particular, the recursion in Lemma 2.3.8 is more correctly a backward recursion and the time argument in (2.3.61) is strictly speaking 0.

The discussion here has dealt with the bounded queueing system. The results of Lemma 2.3.8, Corollary 2.3.9 and Theorem 2.3.10 hold equally well for the unbounded queueing system. One consequence of Lemma 2.3.8 is an alternative proof of Theorem 2.3.7. We proceed by the following sequence of lemmas:

Lemma 2.3.11. For arbitrary vectors f,g such that

$$<e_i^1,\ f-g> = f(i) - g(i) \begin{cases} >0 & \text{if } i\neq 0 \\ =0 & \text{if } i=0 \end{cases} \qquad (2.3.63)$$

then

(a) $$\langle e_i^1, f-g\rangle_{G_1} \equiv e_i^1 \, G_1(f-g)>0 \qquad \text{for all } i \tag{2.3.64}$$

and

(b) $$\langle e_i^1, f-g\rangle_{R_1} \equiv e_i^1 \, R_1(f-g)>0 \qquad \text{for all } i \tag{2.3.65}$$

where G_1, R_1 are defined in (2.3.9).

<u>Proof</u>: For (a), we have for $i>1$

$$\begin{aligned}\langle e_i^1, f-g\rangle_{G_1} &= d_1[f(i-1) - g(i-1)] \\ &\quad + (1-b_1-d_1)[f(i) - g(i)] \\ &\quad + d_1[f(i+1) - g(i+1)]>0 \end{aligned} \tag{2.3.66}$$

where each term in brackets is positive by (2.3.63). For $i=1$, by (2.3.63) the first term in (2.3.66) is zero but the remaining terms are positive; hence (2.3.64) holds for $i=1$. Finally for $i=0$

$$\begin{aligned}\langle e_1^1, f-g\rangle_{G_1} &= (1-\lambda_1)[f(0) - g(0)] \\ &\quad + \lambda_1[f(1) - g(1)]>0\end{aligned}$$

since the first term is zero and the second term is positive. The proof of (b) follows along the same lines. QED.

<u>Lemma 2.3.12</u>. Define the column vectors $\{d_k^i : i=1,2,;k=0,1,2,\dots,T\}$ as specified in Lemma 2.3.8. For $k = 0,1,2,\dots,T$ if $\mu_1 c_1 > \mu_2 c_2$ then

$$a_k(i,j) = \begin{cases} 1 & \text{if} \quad i\neq 0 \\ 0 & \text{if} \quad i=0 \end{cases} \tag{2.3.67}$$

or equivalently for $i\neq 0$

$$e_i^1[P^1(0) - P^1(1)]d_{k-1}^1 > e_j^2[P^2(1) - P^2(0)]d_{k-1}^2 \qquad (2.3.68)$$

and the reverse inequality for i=0.

Proof: By mathematical induction. For k=1, (2.3.68) becomes

$$e_i^1[P^1(0) - P^1(1)]d_0^1 = e_i^1[R_1 - G_1](c_1\nu_1)$$

$$= e_i^1[c_1\mu_1 \ \tilde{1}] \qquad (2.3.69)$$

where the second equality follows from (2.2.9) and

$$\tilde{1} = [0,1,1,\ldots,1,\ldots]^T \qquad (2.3.70)$$

Similarly by (2.2.10)

$$e_j^2[P^2(1) - P^2(0)]d_0^2 = e_j^2[R_2 - G_2](c_2\nu_2)$$

$$= e_j^2[c_2\mu_2 \ \tilde{1}] \qquad (2.3.71)$$

By combining (2.3.52), (2.3.69) - (2.3.71), then for k=1 (2.3.68) holds. Suppose (2.3.68) holds for k; specifically

$$e_i^1 \ G_1 \ d_k^1 + e_j^2 \ R_2 \ d_k^2 < e_i^1 \ R_1 \ d_k^1 + e_j^2 \ G_2 \ d_k^2 \qquad (2.3.72)$$

$$d_{k+1}^1 = d_0^1 + G_1 \ d_k^1 \text{ and } d_{k+1}^2 = d_0^2 + R_2 \ d_k^2 \qquad (2.3.73)$$

We shall show (2.3.68) holds for k+1. First, we have by (2.3.73)

$$e_i^1 \ G_1 \ d_{k+1}^1 + e_j^2 \ R_2 \ d_{k+1}^2$$

$$= e_i^1 \ G_1 \ [d_0^1 + G_1 \ d_k^1] + e_j^2 \ R_2[d_0^2 + R_2 \ d_k^2]$$

$$< e_i^1 \ G_1 \ [d_0^1 + R_1 \ d_k^1] + e_j^2 \ R_2 \ [d_0^2 + G_2 \ d_k^2] \qquad (2.3.74)$$

where the last inequality follows from (2.3.72) and Lemma 2.3.11.

For (i,j)≠(0,0) by Lemma 2.3.3, equation (2.3.37) we have

$$e_i^1(R_1G_1) = e_i^1(G_1R_1)$$
$$e_j^2(R_2G_2) = e_j^2(G_2R_2) \qquad (2.3.75)$$

so that combined with (2.3.74)

$$\begin{aligned} & e_i^1 \; G_1 \; d_{k+1}^1 + e_j^2 \; R_2 \; d_{k+1}^2 \\ & < e_i^1[G_1d_0^1 + R_1G_1d_k^1] + e_j^2[R_2d_0^2 + G_2R_2d_k^2] \\ & = e_i^1 \; R_1[d_0^1 + G_1d_k^1] + e_i^1(G_1 - R_1)d_0^1 \\ & + e_j^2G_2[d_0^2 + R_2d_k^2] + e_j^2(R_2 - G_2)d_0^2 \\ & < e_i^1 \; R_1 \; d_{k+1}^1 + e_j^2 \; G_2 \; d_{k+1}^2 \end{aligned} \qquad (2.3.76)$$

where the last equality follows from (2.3.69) - (2.3.71) and (2.3.73).

For i=0, j≠0 it follows from (2.2.9), (2.3.52) and (2.3.68) that

$$e_0^1[P^1(0) - P^1(1)]d_{k+1}^1 = e_0^1(R_1 - G_1)d_{k+1}^1 = 0 \qquad (2.3.77)$$

since the first rows of R_1, G_1 are identical. Combining (2.3.76), (2.3.77) the result (2.3.68) holds for k+1. QED.

<u>Lemma 2.3.13</u>. Define the column vectors $\{d_k^i:i=1,2;k=0,1,2,...,T\}$ as specified in Lemma 2.3.8. For k = 1,2,...,T if $\mu_2c_2>\mu_1c_1$ then

$$a_k(i,j) = \begin{cases} 0 & \text{if } j\neq 0 \\ 1 & \text{if } j=0 \end{cases} \qquad (2.3.78)$$

or equivalently for j≠0

$$e_i^1[P^1(0) - P^1(1)]d_{k-1}^1<e_j^2[P^2(1)-P^2(0)]d_{k-1}^2 \qquad (2.3.79)$$

and the reverse inequality for $j=0$.

Proof: Follows along the same lines as in Lemma 2.3.12.

Given the above Lemmas 2.3.12, 2.3.13 for an unbounded queueing system, we have the following:

Alternative proof of Theorem 2.3.7: Without loss of generality assume $\mu_2 c_2 > \mu_1 c_1$. By Theorem 2.3.10, the optimal strategy at time k is given by

$$g_f^*(k;i_1,i_2) = a_k(i_1,i_2) = \begin{cases} 1 & \text{if } i_1 \neq 0,\ i_2=0 \\ 0 & \text{if } i_2 \neq 0 \\ \text{arbitrary} & i_1=i_2=0 \end{cases} \tag{2.3.80}$$

where the second equality follows from (2.3.78) for $i_1 \neq 0$ and (2.3.77) for $i_1=0$. Clearly, (2.3.80) is identical to (2.3.40) and the remainder of the proof follows as before. QED.

Remark 2.3.5. The discussion of both the bounded and unbounded queueing system (2.3.8), (2.3.10) assumed that $\{\lambda_i, \mu_i;\ i=1,2\}$ are constant. The above algorithm, specified by Lemma 2.3.8, Corollary 2.3.9 and Theorem 2.3.10, can be adapted to handle a time-varying, state dependent system by replacing

$$\lambda_i \to \lambda_i(k,j)$$

$$\mu_i \to \mu_i(k,j)$$

$$P^i(u) \to P^i(u,k) \quad \text{(appropriately modified)}$$

for all $k = 0,1,2,\ldots T$; $j \varepsilon Z_i$; $i=1,2$.

Remark 2.3.6. The implementation of the optimal strategy is quite simple. The decision space $X = Z_1 \times Z_2$ is divided at the k^{th} step into at most 2^k subsets which are characterized by the binary numbers with k binary digits i.e., $a_k\ a_{k-1}\ a_{k-2} \ldots a_0$. The first binary digit of the number is associated with the control as provided in Corollary 2.3.9. These observations are quite useful when implementing these strategies in a microprocessor. The only on-line computation needed is the propagation of the queue sizes (2.3.8) and the selection of the pre-stored arrays $\{a_k\ ;\ k = 1,2,\ldots,T\}$.

2.4 Infinite Horizon Discounted Stochastic Optimal Control General Results

Motivated by the results of the previous section, we are interested now in the existence of a stationary optimal policy, under the infinite horizon discounted criterion (2.2.20). In the present and subsequent section, we shall assume: (i) the number of stages are infinite, (ii) the system dynamics are stationary, (iii) the queueing system (2.2.1) - (2.2.17) has unbounded capacity. These assumptions constitute a reasonable and analytically convenient approximate for problems involving a very large, but finite number of stages. The stationarity and unboundedness assumptions provide mathematical and conceptual elegance not found in the general finite horizon problem. Moreover, the infinite horizon problem introduces certain analytical tools needed in analyzing the limiting behavior of the system. These tools are exploited in the adaptive control problem of Chapter 4.

As in the finite horizon problem, the state space is an n-dimensional, Euclidean vector space, X with state dynamics satisfying:

$$x(t+1) = \varphi(x(t), u(t), w(t+1)) \qquad (2.4.1)$$

$$\text{for } t = 0,1,2,\ldots$$

where $u(t)\varepsilon U$ are the control values and $w(t)\varepsilon D$ are independent random variables with a known distribution. The disturbance space D is assumed countable. The general problem description is as before (see the beginning of Section 2.3) given the above assumptions (i) - (iii). For a control policy, $\gamma\varepsilon\Gamma$ the infinite horizon performance criterion is denoted by

$$J^{\gamma}_{d,\beta}(x) = E[\sum_{t=0}^{\infty} \beta^{t} c(x^{\gamma}(t), u^{\gamma}(t))] \qquad (2.4.2)$$

where $\beta\varepsilon[0,1)$ is the discount factor and $c(x,u)$ denotes the instantaneous cost. The expectation above is taken with respect to a given probability distribution, $p(\cdot|x,u)$ which depends on x, u (see discussion between (2.3.1) and (2.3.3)). The superscript γ in x,u indicates the state and control trajectories induced by the policy γ. The set of admissible control policies Γ is as defined in (2.2.12) - (2.2.15). The problem is to find $\gamma^{*}\varepsilon\Gamma$ such that

$$J^{\gamma *}_{d,\beta}(x) = \inf\{J_{d,\beta}(x):\gamma\varepsilon\Gamma\} \qquad (2.4.3)$$

The corresponding policy, γ^{*} is called <u>optimal</u>. A class of admissible policies of interest is the class of <u>stationary admissible policies</u>, $\Gamma_{s}\subset\Gamma$ of the form:

$$\gamma = (g,g,\ldots) \qquad (2.4.4)$$

where

$$u(t) = g(y^{t-1}, u^{t-1}) \quad ; \quad g\varepsilon U \qquad (2.4.5)$$

For such policies, the rule for control selection is the same for each stage.

In the case of uniformly bounded instantaneous cost

$$|c(x,u)|<M<\infty \quad \text{for all } x\varepsilon X,\ u\varepsilon U \qquad (2.4.6)$$

the following approach is standard [20], [22].

First, one defines from (2.4.1) - (2.4.3) an operator, $H(\cdot)$ such that

$$H(J)(x) = \inf_{u\varepsilon U} E[c(x,u) + J(\varphi(x,u,w))] \qquad (2.4.7)$$

The operator $H(\cdot)$ is defined on $B(X)$, the set of all bounded real-valued functions on X. With every function $J:X\rightarrow R$ that belongs to $B(X)$, we associate the metric

$$\|J\| = \sup_{x\varepsilon X} |J(x)| \qquad (2.4.8)$$

Because the discount factor $\beta\varepsilon\{0,1)$, the operator $H(\cdot)$ is a contraction mapping; specifically

<u>Definition 2.4.1</u>. A mapping $H:B(X)\rightarrow B(X)$ is a contraction mapping, if there exists a scalar $\rho<1$ such that

$$\|H(J) - H(J')\|\leq\rho\|J-J'\| \quad \text{for all } J,J'\varepsilon B(X)$$

where $\|\cdot\|$ is defined in (2.4.8).

Second by the contraction mapping property, we have the following:

<u>Contraction Mapping Fixed Point Theorem 2.4.2</u>. If $H:B(X)\rightarrow$

B(X) is a contraction mapping, then there exists a unique point of H, i.e. there exists a unique function $J^* \varepsilon B(X)$ such that

$$H(J^*) = J^* \qquad (2.4.9)$$

Furthermore, if J is any function in B(X) and H^k denotes the composition of H with itself k times, then

$$\lim_{k\to\infty} \|H^k(J) - J^*\| = 0 \qquad (2.4.10)$$

The fixed point theorem leads to a necessary and sufficient optimality condition for the problem (2.4.3) [20, Proposition 2, p. 229].

Theorem 2.4.3. For a uniformly bounded instantaneous cost (2.4.6), the optimal value function satisfies

$$J^{\gamma^*}_{d,\beta}(x) = \inf_{u\varepsilon U} E[c(x,u) + \beta J^{\gamma^*}_{d,\beta}(\varphi(x,u,w))] \qquad (2.4.11)$$

$$\text{for all } x\varepsilon X$$

or equivalently

$$J^{\gamma^*}_{d,\beta}(x) = H(J^{\gamma^*}_{d,\beta})(x) \equiv J^*$$

Furthermore, J^* is the unique bounded solution of (2.4.11). In addition if $g^*:X\to U$ is a function such that $g^*(x)$ attains the infimum in the right-hand side of (2.4.11) for each $x\varepsilon X$, then the stationary policy $\gamma^* \varepsilon \Gamma_s \subset \Gamma$, $\gamma^* = (g^*, g^*, \ldots)$ is optimal. Conversely if $\gamma^* = (g^*, g^*, \ldots)$ is an optimal stationary policy, then $g^*(x)$ attains the infimum in the right-hand side of (2.4.11) for all $x\varepsilon X$.

Third, the optimal value function $J^{\gamma^*}_{d,\beta}(\equiv J^*)$ and the optimal stationary policy γ^* of Theorem 2.4.3 can be obtained by dynamic programming. Let $J_0 = 0$ on X and

$$J^*_{k+1}(x) = H(J^*_k)(x) \qquad \text{for all } x\varepsilon X \qquad (2.4.12)$$
$$k = 0,1,2,...$$

Then it follows from (2.4.6), (2.4.10) and (2.4.11)

$$J^*(x) = J^*_\infty(x) = \lim_{k\to\infty} J^*_k(x)<\infty \qquad (2.4.13)$$

Recall from Section 2.3 that (2.4.13) was the starting point of our analysis for the finite horizon problem.

To obtain the optimal policy, we have to overcome a slight technical problem when the instantaneous cost does not satisfy (2.4.6). The value function, $J^\gamma_{d,\beta}(x_0)$ in (2.4.2) for some initial state x_0 and some admissible policy $\gamma\varepsilon\Gamma$ may be infinite. Thus, we shall conduct our analysis with the understanding that the value function is an extended real-valued function. Within the context of the optimization problem (2.4.3), we shall follow the results of Lippman [16] and allow a polynomial growth (in the state) of the instantaneous cost. Our approach is to define an equivalent operator as in (2.4.7) under a suitable metric. The contraction mapping and fixed point properties then follow. The results to be presented provide a characterization of the optimal value function $J^{\gamma*}_{d,\beta}$ as well as the optimal stationary policy, γ^*. In particular, the successive approximation method is applied to obtain the limiting optimal value (2.4.13). Rather than presenting a general result (the reader is referred to [19]), we shall concentrate our development on the intended application. Our methodology follows the standard approach mentioned above.

Application to the Two Competing Queue Problem

For the two competing queue system of Section 2.2, the infinite horizon formulation is as follows. The state dynamics and instantaneous cost are given respectively by (2.2.16) and (2.2.19)

$$x_i(t+1) = x_i(t) + n_i^a(t+1) - n_i^d(t+1) \qquad ; i = 1,2 \tag{2.4.14}$$

$$c(x(t), u(t)) = c_1 x_1(t) + c_2 x_2(t) = c^T x(t) \tag{2.4.15}$$

where

$$c^T = (c_1, c_2) \text{ and } x(t) = (x_1(t), x_2(t))^T$$

These dynamics (2.4.14) can be expanded as in (2.3.10). For a policy $\gamma \varepsilon \Gamma$, the cost is the discounted average aggregate delay

$$J_{d,\beta}^{\gamma}(x_0) = E[\sum_{t=0}^{\infty} \beta^t \cdot c^T x^{\gamma}(t)] \tag{2.4.16}$$

where the discount factor $\beta \varepsilon (0,1)$. The control space for the single server queue is $U = \{0,1\}$. The queueing system is assumed to have unlimited capacity. Consequently, the state space is the Cartesian product, $X = Z \times Z$ where Z is the set of positive integers.

The existence result of Lippman [16] requires the following assumptions:

(A1) Only a finite number of states are accessible from each state in one transition

(A2) There are constants K and m such that for each t

$$\max_{t'} \sup_{v \varepsilon U} E[c(x(t'), u(t'))|x(t) = x, u(t)=v] \le K(tV1)^m \tag{2.4.17}$$

(A3) With probability one, only a finite number of transitions are made in a finite amount of time.

Assumptions (A1) and (A3) are satisfied trivially for the queueing system (2.2.1) - (2.2.17) since only one arrival and one departure can occur in each queue during each time slot (see (2.3.10)). Assumption (A2) holds for the cost (2.4.15) when m = 1. A slight modification of the arguments in [16, Theorem 1] for discrete time, establishes Denardo's [23] N-stage contraction assumption in an appropriate metric space with weighted sup metric. These arguments provide the proof of the following result, since the control space, U = {0,1} is finite.

Theorem 2.4.4. If Assumptions (A1) - (A3) hold, then for $\beta \varepsilon [0,1)$ there is an optimal stationary policy γ^* for the infinite horizon discount problem (2.4.3), (2.4.16). The optimal policy and optimal cost, $V_d^*(x)$ are determined from the stationary Bellman functional equation:

$$V_d^*(i_1,i_2) = c_1 i_1 + c_2 i_2 + \beta \min_{v \varepsilon \{0,1\}} [\sum_{x \varepsilon X} p_{i_1 i_2;x}(v) V_d^*(x)] \tag{2.4.18}$$

for all $(i_1,i_2) \varepsilon X$

where $P(v) = \{p_{i_1 i_2; j_1 j_2}(v)\}$ is the transition probability matrix given in (2.2.8) - (2.2.11). Moreover, V_d^* is the unique solution of (2.4.18).

Proof. Basically we follow [16], [23] where (A1) - (A3) are respectively (1'), Assumptions 1 and 2, of [16, pp. 718-720]. The proof is given here for completeness. Let (β,ρ) be the complete metric space of doubly indexed sequences.

$$\zeta(\cdot,\cdot) : X \to R$$

where

$$B = \{\zeta(\cdot,\cdot) : \sup_{i_1,i_2} \frac{|\zeta(i_1,i_2)|}{((i_1+i_2) \vee 1)} < \infty\} \tag{2.4.19}$$

and

$$\rho(\zeta,\zeta') = \sup_{i_1 i_2} \frac{|\zeta(i_1,i_2) - \zeta'(i_1,i_2)|}{((i_1+i_2) \vee 1)} , \quad \zeta,\zeta' \varepsilon \beta(X) \tag{2.4.20}$$

For each control function, g defined in (2.2.15), we define the operator $H_g : B(X) \to B(X)$ by

$$(H_g \zeta)(i_1,i_2) = c_1 i_1 + c_2 i_2 + \beta \sum_{x \varepsilon X} p_{i_1 i_2;x}(g) \zeta(x)$$

To see that $H_g \zeta \varepsilon B(X)$ for $\zeta \varepsilon B(X)$, let $M = \rho(\zeta,0)$ and observe

$$\begin{aligned}(H_g\zeta)(i_1,i_2) &\le (c_1 \vee c_2)\cdot(i_1+i_2)\\ &\quad + \beta \sum_{x\varepsilon X} p_{i_1 i_2;x}(g)\cdot M\cdot((j_1+j_2) \vee 1)\\ &\le (c_1 \vee c_2)(i_1+i_2) + \beta M\cdot(i_1+i_2+2)\\ &\le [(i_1+i_2) \vee 1]\cdot[(c_1 \vee c_2) + \beta M\cdot\frac{(i_1+i_2+2)}{(i_1+i_2) \vee 1}]\\ &\le [(i_1+i_2) \vee 1][(c_1 \vee c_2) + 3 \beta M]\end{aligned} \tag{2.4.22}$$

where $x = (j_1,j_2) \varepsilon X$. Next we show that H_g^k is a contraction mapping for some k^*. From (2.4.18), we see that

$$|(H_g\zeta - H_g\zeta')(i_1,i_2)| = \beta \sum_{x\varepsilon X} p_{i_1 i_2;x}(g)\ [\zeta(x) - \zeta'(x)]|$$

$$\leq \beta\rho(\zeta\ \zeta')\cdot(i_1+i_2+2) \qquad (2.4.23)$$

By induction

$$|(H_g^{k+1}\zeta - H_g^{k+1}\zeta')(i_1,i_2)| \leq \beta[\sum_{x\varepsilon X} p_{i_1 i_2;x}(g)\,|(H_g^*\zeta - H_g^k\zeta')(x)|]$$

$$\leq \beta \sum_{x\varepsilon X} \beta^k p_{i_1 i_2;x}(g)\rho(\zeta,\zeta')\cdot(i_1+i_2+2k)$$

$$\leq \beta^{k+1}\rho(\zeta,\zeta')\cdot(i_1+i_2+2k) \qquad (2.4.24)$$

Therefore

$$\rho(H_g^k\zeta,\ H_g^k\zeta') \leq \beta^k\rho(\zeta,\zeta')\cdot\sup_{i_1,i_2} \frac{(i_1+i_2+2k)}{((i_1+i_2)\ \vee\ 1)}$$

$$= \beta^k\rho(\zeta,\zeta')\ (1+2k) \qquad (2.4.25)$$

Choose k^* such that $\beta^{k^*}(1+2k^*)\leq\beta$ and the result follows. Note that the bounds in (2.4.22) - (2.4.25) are independent of the choice of g and therefore for any finite sequence $g_1,\ g_2,\ldots,g_k$ of control functions

$$H_{g_1} \circ H_{g_2} \circ \cdots \circ H_{g_k} \qquad (2.4.26)$$

is a contraction on $B(X)$. By defining the operator, H_g^* : $B(X)\to B(X)$ such that

$$(H_g^*\zeta)(i_1,i_2) = \inf\{(H_g\zeta)(i_1,i_2)\ :\ \zeta\varepsilon B(X)\}$$

then it follows from the above (2.4.22) - (2.4.25) that $(H_g^*)^{k*}$ is a contraction mapping on $B(X)$. Hence by Theorem 2.4.2, H_g^* is a unique fixed point. The existence of an optimal stationary policy follows from Denardo [23, Corollary 2]. QED.

Similar to the uniformly bounded case (see Theorem 2.4.3 and (2.4.13)), Theorem 2.4.4 provides a means of characterizing the optimal value function (2.4.3), (2.4.16) and optimal stationary policy, $\gamma^* \varepsilon \Gamma_s \subset \Gamma$. In particular, by the uniqueness of the solution in (2.4.18), we define the following dynamic programming recursion:

$$V_0(x) = c^T x$$

$$V_{k+1}(x) = c^T x + \beta\{T^0 V_k(x) \wedge T^1 V_k(x)\} \tag{2.4.27}$$

where for notational convenience

$$V_{k+1}(x) = (H_g^* V_k)(x) = (H_g^* V_0)^k(x) \tag{2.4.28}$$

$$\text{for all } x \varepsilon X,\ k = 0,1,2,\ldots$$

and $T^0 V(\cdot)$ and $T^1 V(\cdot)$ are defined in (2.3.14), (2.3.15) respectively. By combining (2.3.10), (2.4.14) - (2.4.16) with (2.4.18), the resulting recursion (2.4.27) follows. As in the finite horizon problem we have the following sequence of lemmas.

Lemma 2.4.5. For each $k = 0,1,2,\ldots$ if $\mu_2 c_2 < \mu_1 c_1$ and $c_1, c_2 > 0$ then

(a) $V_k(A_2 x) > V_k(x)$, $V_k(A_3 x) > V_k(x)$ for all $x \varepsilon X$ (2.4.29)

(b) $T^1 V_k(x) > T^0 V_k(x)$ for all $x = (i_1, i_2) \varepsilon X$; $i_2 \neq 0$ (2.4.30)

(c) $T^0 V_k(x) > T^1 V_k(x)$ for all $x = (i_1, 0)$; $i_1 \neq 0$ (2.4.31)

Proof: By mathematical induction similar to the arguments in Lemma 2.3.4, the result follows. QED.

Lemma 2.4.6. For each $k = 0,1,2,\ldots$ if $\mu_1 c_1 > \mu_2 c_2$ and $c_1, c_2 > 0$ then

$$\text{(a)} \quad V_k(A_2 x) > V(x), \quad V_k(A_3 x) > V_k(x) \quad \text{for all } x \varepsilon \mathcal{X} \tag{2.4.32}$$

$$\text{(b)} \quad T^0 V_k(x) > T^1 V_k(x) \quad \text{for all } x = (i_1, i_2) \varepsilon \mathcal{X};\ i_1 \neq 0 \tag{2.4.33}$$

$$\text{(c)} \quad T^1 V_k(x) > T^0 V_k(x) \quad \text{for all } x = (0, i_2);\ i_2 \neq 0 \tag{2.4.34}$$

Clearly $c^T x \geq 0$, therefore $V_k(x) \leq V_{k+1}(x)$ and so this limit exists:

$$V_\infty(x) = \lim_{k \to \infty} V_k(x) \qquad \text{for all } x \varepsilon \mathcal{X} \tag{2.4.35}$$

Moreover, the optimal value function is well-defined.

Lemma 2.4.7. For $x \varepsilon \mathcal{X}$ and for the optimal policy $\gamma^* \varepsilon \Gamma_s \subset \Gamma$, $J^{\gamma^*}_{d,\beta}(x) < \infty$.

Proof. [8, p. 602] Let $|x| = i_1 + i_2$, $|c| = \max(c_1, c_2) = c_1 \wedge c_2$. For any initial state and any policy, the state $x(t)$ at time t must satisfy

$$|x(t)| \leq |x| + t$$

Moreover

$$V_k(x) \leq \sum_{t=0}^{k-1} \beta^t |c| [\, |x| + k]$$

$$< \frac{|c|\,|x|}{(1-\beta)} + \frac{|c|\ \beta}{(1-\beta)^2} < \infty \quad \text{for } \beta \varepsilon [0,1)$$

Hence it follows from Theorem 2.4.4,

$$J^{\gamma^*}_{d,\beta}(x) = V_\infty(x) < \infty \qquad \text{for all } x \varepsilon \mathcal{X} \qquad \text{QED.}$$

Theorem 2.4.8. There is an optimal stationary policy $\gamma^*_d \varepsilon \Gamma_s \subset \Gamma$ for the unbounded queueing system (2.4.14), (2.3.10) under the infinite horizon discounted average aggregate delay criterion (2.4.15), (2.4.16). The optimal policy and cost V_d^* are determined from the optimality equation (2.4.18), (2.4.27). Specifically, let $g_g^*(\cdot,\cdot) : Z x Z \to U$ define the optimal policy, i.e.

$$\gamma_d^* = (g_d^*, g_d^*, \ldots, g_d^*)$$

then $g_d^*(\cdot,\cdot)$ is the "μc rule" given in (2.3.17), (2.3.18).

Proof. By Theorem 2.4.4 and Lemmas 2.4.5 - 2.4.7, the result follows along the same lines as in Theorem 2.3.7.

QED.

2.5 Expected Long-Run Average Cost Stochastic Optimal Control

In this section, the expected long-run average cost (2.2.21) problem for the unbounded queueing system (2.2.1) - (2.2.17) is considered. Because of the special structure of the optimal discounted control strategy, one might expect the same structure for the optimal average cost policy. This indeed is the case. In general, the average cost problem is treated as the limit of the discounted problem [16], [20], [22], [31]. However in our problem, the optimal μc-rule strategy is insensitive to the discount factor, β or the number of stages (see the results in Theorems 2.4.8 and 2.3.7, respectively). Consequently, we exploit these properties by following the unusual procedure of moving to the average cost as the limit of the finite horizon problem. This approach was followed by Rosberg et al [8] in a tandem queue problem. Since this latter approach is problem dependent, only the intended application is presented; no general formulation is discussed. Thus after certain preliminaries, the average cost two competing queue result is stated.

For the long-run average cost problem, let χ denote the state space and U the finite control space such that for each $(i_1,i_2)\varepsilon\chi$, $u\varepsilon U$ there corresponds a set of transition probabilities $\{p_{i_1i_2;j_1j_2}(u):(j_1,j_2)\varepsilon\chi\}$ where

$$p_{i_1i_2;j_1j_2}(v) = \Pr\{x(t+1)=(j_1,j_2)\,|\,x(t) = (i_1,i_2),u(t) = v\} \qquad (2.5.1)$$

For a control policy, $\gamma\varepsilon\Gamma$ and finite horizon T, the expected finite horizon cost criterion is denoted by

$$J^{\gamma}_{f,T}(x) = E[\sum_{t=0}^{T-1} c(x^{\gamma}(t),u^{\gamma}(t))] \qquad (2.5.2)$$

and the expected average cost criterion is denoted by

$$J^{\gamma}_{a}(x) = \liminf_{T\to\infty} \frac{1}{T} J^{\gamma}_{f,T}(x)$$

$$= \liminf_{T\to\infty} \frac{1}{T} E[\sum_{t=0}^{T-1} c(x^{\gamma}(t),u^{\gamma}(t))] \qquad (2.5.3)$$

where c(x,u) denotes the instantaneous cost. For the two competing queue problem, the transition probabilities are given in (2.2.8) - (2.2.11) and the instantaneous cost is of the form:

$$c(x^{\gamma}(t),u^{\gamma}(t)) = c_1x_1(t) + c_2x_2(t) \qquad (2.5.4)$$

The expectation above is taken with respect to the probability distributions in (2.5.1) which depend on x,u.

The superscript γ in x,u indicates the state and control trajectories induced by the policy γ. The set of admissible control policies Γ is as defined in (2.2.12) - (2.2.15), (2.2.17). The average cost per unit time problem is to find $\gamma^* \varepsilon \Gamma$ such that

$$J_a^{\gamma^*}(x) = \min\{J_a^{\gamma}(x) : \gamma \varepsilon \Gamma\} \qquad \text{for all } x \varepsilon \chi \qquad (2.5.5)$$

The corresponding policy, γ^* is called average cost optimal. A class of admissible policies of interest is the class of stationary admissible policies, $\Gamma_s \subset \Gamma$ of the form

$$\gamma = (g,g,\ldots) \qquad (2.5.6)$$

where

$$u(t) = g(y^{t-1}, u^{t-1}) \; ; \; g \varepsilon U \qquad (2.5.7)$$

For such policies, the rule for control selection is the same for each state. As in Section 2.4, we shall assume: (i) the system dynamics are stationary and (ii) the queueing system (2.2.1) - (2.2.17) has unbounded capacity.

Given these preliminaries, we have the following result:

Theorem 2.5.1. For the unbounded, stationary queueing system (2.3.8), (2.3.10) under the expected long-run average cost criterion (2.5.3) - (2.5.5), the μc-rule is the optimal stationary policy.

Proof: Let $\bar{\gamma} = (\bar{g},\bar{g},\ldots)\varepsilon\Gamma_s$ where $\bar{g}(x), x\varepsilon\chi$ denotes the μc-rule. By Theorem 2.3.7 for each finite T, we have

$$J^{\bar\gamma}_{f,T}(x)\le J^{\gamma}_{f,T}(x) \qquad \text{for all } x\varepsilon\chi,\ \gamma\varepsilon\Gamma \tag{2.5.8}$$

Now consider

$$J^{\bar\gamma}_{a}(x) = \liminf_{T\to\infty} \frac{1}{T} J^{\bar\gamma}_{f,T}(x) \tag{2.5.9}$$

First, suppose that $J^{\bar\gamma}_{a}(\cdot)$ is infinite. Then given an M and T', there is a $t\ge T'$ such that

$$0<M<\frac{1}{t} J^{\bar\gamma}_{f,t}(x)\le\frac{1}{t} J^{\gamma}_{f,t}(x) \qquad \text{for all } \gamma\varepsilon\Gamma \tag{2.5.10}$$

where the second inequality follows from (2.5.8). Hence for each $\gamma\varepsilon\Gamma$, $J^{\gamma}_{a}(\cdot)$ is unbounded so any policy is average cost optimal; in particular the μc-rule is optimal. Conversely, suppose that $J^{\bar\gamma}_{a}(\cdot)$ is finite, i.e.

$$J^{\bar\gamma}_{a}(x) = \Lambda_x < \infty \qquad \text{for all } x\varepsilon\chi \tag{2.5.11}$$

Now to show that $\bar\gamma\varepsilon\Gamma_s$ is average cost optimal, let us suppose there exists a policy $\gamma'\varepsilon\Gamma_s$ such that

$$J^{\gamma'}_{a}(x) < \Lambda \qquad \text{for some specific } x\varepsilon\chi \tag{2.5.12}$$

Then

$$J^{\gamma'}_{a}(x) = \Lambda_x - \varepsilon_1 \qquad \text{for some } \varepsilon_1>0 \tag{2.3.13}$$

By the definition of lim inf [64], there exists a $T'(\varepsilon_1)$ such that

$$\frac{1}{t} J^{\bar\gamma}_{f,t}(x)>\Lambda_x - \frac{\varepsilon_1}{2} \qquad \text{for all } t>T'(\varepsilon_1) \tag{2.5.14}$$

Moreover, given ε_1 and $T'(\varepsilon_1)$, there exists a $t'\ge T'(\varepsilon_1)$ such that

$$J^{\gamma'}_{a}(x) + \frac{\varepsilon_1}{2} > \frac{1}{t'} J^{\gamma'}_{f,t'} \tag{2.5.15}$$

By combining (2.5.13) - (2.5.15), it follows

$$\frac{1}{t'} J^{\bar\gamma}_{f,t'}(x)>\Lambda_x - \frac{\varepsilon_1}{2} = J^{\gamma'}_{a}(x) + \frac{\varepsilon_1}{2} > \frac{1}{t'} J^{\gamma'}_{f,t}(x)$$

which contradicts (2.5.8). Hence for $J^{\bar\gamma}_{a}(\cdot)$ finite, the μc-rule is average cost optimal. QED.

2.6 Evaluation - Finite Queue System

In this section, the infinite horizon discounted average aggregate delay (2.2.20) and the expected long-run average cost aggregate delay (2.2.21) problems for the bounded queueing system (2.2.1) - (2.2.17) are considered. For the finite capacity system, condition (2.2.6) is satisfied. The methods presented here lead to a numerical treatment; analytical solutions have not been obtained. We intend to demonstrate via numerical examples the additional complexity in determining the optimal control strategy. While in the unbounded system, our major effort involved handling the cases of zero queues, here one must handle in addition the upper boundary states. The sensitivity of the optimal policy to the rate parameters $\{\lambda_i, \mu_i;\ i=1,2\}$, to the queue size and the corresponding performance criterion is presented.

Before proceeding, we shall first discuss the computational algorithms used in this evaluation. For the discounted performance criterion, we have selected the successive approximation methodology [20, p. 237]. Specifically from (2.3.13), we have the following recursion:

$$V_0(i,j) = c_1 i + c_2 j$$

$$V_{k+1}(i,j) = c_1 i + c_2 j + \beta \min_{u\varepsilon U} \{ \sum_{n,m=0}^{N} p^1_{i,m}(u) \cdot V_k(m,n) \cdot p^2_{j,n}(u) \} \qquad (2.6.1)$$

where $\beta\varepsilon(0,1)$ is the discount factor, $N=N_1=N_2$ is the maximum queue size (see (2.26)) and $\{p^k_{i,j}(u): k=1,2\}$ are the transition probabilities given in (2.2.4) - (2.2.7). Note, the term in brackets is easily implemented as a weighted inner product norm with respect to the rows of $P^1(u)$, $P^2(u)$ respectively, i.e.

$$<p_i^1(u),\ p_j^2(u)>_{V_k}$$

For the average cost performance objective, three different implementations were investigated; successive approximations, linear programming and policy iteration. The successive approximation approach involved simply using β equal to unity in (2.6.1). However, as will be apparent later, the solution is not numerically stable for large β. For β small, the contraction property holds with the algorithm (2.6.1) converging in 10-20 iterations. For β large, the algorithm typically failed to converge in 40 iterations. The convergence criterion was the following:

$$|V_{k+1}(i,j) - V_k(i,j)|<0.01 \text{ for all } i,j\varepsilon\chi \qquad (2.6.2)$$

Alternatively, the linear programming approach of Wolfe and Dantzig [67] was developed. The average cost problem reduces to the following linear programming (LP) formulation:

$$\min \sum_{u=0}^{1} \sum_{i,j=0}^{N} (c_1 i+c_2 j)\cdot y(i,j;u) \qquad (2.6.3)$$

subject to

$$\sum_{u=0}^{1} \sum_{i,j=0}^{N} y(i,j;u) = 1 \quad ; \quad y(i,j;u)\geq 0$$

and

$$\sum_u \sum_{m,n} p^1_{i,m}(u) y(m,n;u) p^2_{j,n}(u) - \sum_u y(i,j;u) = 0 \tag{2.6.4}$$

By letting d(i,j;u) be the probability of choosing action u given the state (i,j), then it follows

$$\begin{aligned} d(i,j;v) &= \Pr\{u(i,j) = v \mid x = (i,j)\} \\ &= \frac{y(i,j;v)}{\sum_v y(i,j;v)} \end{aligned} \tag{2.6.5}$$

The implementation of (2.6.3) - (2.6.5) introduced a $2N^2$-dimensional vector, $Y = \{y(i,j;u)\}_{i,j,u}$ and an augmented transition probability matrix such that the equality constraint (2.6.4) becomes

$$[P(0) \mid P(1)]Y - [I_N \mid I_N]Y = 0 \tag{2.6.6}$$

where P(v) is given in (2.2.11). The minimization in (2.6.3) invoked the simplex method of linear programming. For the cases considered, only in the trivial situations did the LP solution compare with the corresponding discounted cost strategy (β small). In several situations, idling-type policies (i.e. u(0,j) = 1 or u(i,0) = 0) were generated. Consequently, this approach was not felt appropriate and thus was abandoned.

The policy iteration method of Howard [68] was adopted for solving the average cost problem. Given a stationary policy $\gamma \varepsilon \Gamma_s$, $\gamma = (g,g,\ldots)$, the algorithm obtains an improved policy by means of a minimization process on the transition probabilities, until no further improvement is possible. Because of finite state and control spaces, the number of iteration is finite. The basic iteration consists of two steps: value determination and policy improvement. Given a particular policy the value determination step solves

$$h + f(i,j) = c_1 i + c_2 j + \sum_{m,n} p^1_{i,m}(u(i,j)) \cdot f(m,n) \cdot p^2_{j,n}(u(i,j)) \qquad \text{for } i,j = 1,2,\ldots,N \tag{2.6.7}$$

for the relative values $\{f(i,j)\}$ and h by setting f(0,0) to zero. The policy improvement step updates the control values $\{u(i,j)\}$ such that

$$u'(i,j) = \arg\min\{ \sum_{m,n} p^1_{i,m}(u(i,j)) \cdot f(m,n) \cdot p^2_{j,n}(u(i,j))\} \tag{2.6.8}$$

$$\text{for all } i,j = 1,2,\ldots,N$$

The values $\{u'(i,j)\}$ become the new control actions in the next value determination step (2.6.7). The algorithm converges when the set $\{u(i,j)\}$ equals $\{u'(i,j)\}$ in the policy improvement step. To improve computational efficiency, the policy iteration algorithm was initialized with either the equivalent discounted cost policy or the μc rule strategy. On average, the policy iteration algorithm converged more rapidly than the successive approximation algorithm (2.6.1), but the later requires less computations per iteration verses the former.

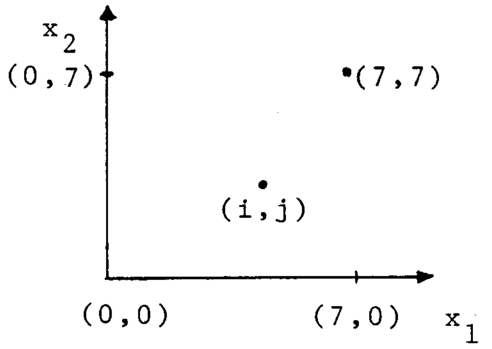
Given these preliminaries, the cases studied are shown in Table 2.1. The size of each queue were identical with

$N = N_1 = N_2 = 7$. Also, the parameters were selected such

Table 2.1 - Finite Queue System (N=7)

Figure	λ_1	c_1	λ_2	c_2	μ_1	μ_1	Optimal Cost
2.2	0.3	1.0	0.3	1.0	0.5	0.5	6.44
2.3	0.3	1.0	0.3	1.0	0.6	0.2	7.32
2.4	0.4	1.0	0.2	2.0	0.6	0.2	13.88
2.5	0.45	1.0	0.2	2.0	0.6	0.2	13.94
2.6	0.5	1.0	0.2	2.0	0.6	0.2	13.99
2.7	0.4	1.0	0.3	2.0	0.6	0.2	14.73
2.8	0.4	1.0	0.2	2.0	0.6	0.15	13.88
2.9	0.4	1.0	0.2	2.0	0.6	0.1	14.84

that $\mu_1 c_1 \geq \mu_2 c_2$. Because of symmetry, there is no loss of generality. The optimal policy of each case is shown in Figures 2.2 - 2.9 with the following notational convention:



For each entry

$$u(i,j) = \begin{cases} 1 & \text{service } 1 \\ 0 & \text{service } 2 \\ 2 & \text{arbitrary} \end{cases}$$

In each figure, the optimal strategies for β = 0.3, 0.5, 0.9 and 1.0 are shown respectively in (a) - (d) with the understanding that for (d) policy iteration is used. In Figure 2.2, the parameters were selected for a completely symmetric system. This case insures that each algorithm handles the trivial case. Observe for small β, the control value is arbitrary in the lower-valued queue states while as β increases, only the diagonal entries retain this property. In Figure 2.3, the parameters were selected to establish that the μc rule holds in the bounded queue system. Here the policy is independent of the discount factor with

$$\mu_1 c_1 = 0.6 > 0.2 = \mu_2 c_2 \text{ and } \lambda_1 c_1 = \lambda_2 c_2 = 0.3$$

One may conjecture that even in the finite case, the μc rule is optimal.

The conjecture is shown false given the results in Figures 2.4 - 2.6. In these cases, the boundary states and the discount parameter influence the optimal policy. First, consider Figure 2.4(a), (b) (β small) which can be explained as follows: For control values $\{u(7,j):j=1,2,...,6\}$,

$$\text{(a)}\quad \begin{matrix} 0&1&1&1&1&1&1&2\\ 0&1&1&1&1&1&2&0\\ 0&1&1&1&1&2&0&0\\ 0&2&2&2&2&0&0&0\\ 0&2&2&2&2&0&0&0\\ 0&2&2&2&2&0&0&0\\ 0&2&2&2&2&0&0&0\\ 2&1&1&1&1&1&1&1 \end{matrix} \qquad\qquad \begin{matrix} 0&1&1&1&1&1&1&2\\ 0&1&1&1&1&1&2&0\\ 0&1&1&1&1&2&0&0\\ 0&1&1&1&2&0&0&0\\ 0&1&1&2&0&0&0&0\\ 0&1&2&0&0&0&0&0\\ 0&2&0&0&0&0&0&0\\ 2&1&1&1&1&1&1&1 \end{matrix}\quad \text{(c)}$$

$$\text{(b)}\quad \begin{matrix} 0&1&1&1&1&1&1&2\\ 0&1&1&1&1&1&2&0\\ 0&1&1&1&1&2&0&0\\ 0&1&1&1&2&0&0&0\\ 0&2&2&2&0&0&0&0\\ 0&2&2&2&0&0&0&0\\ 0&2&2&2&0&0&0&0\\ 2&1&1&1&1&1&1&1 \end{matrix} \qquad\qquad \begin{matrix} 0&1&1&1&1&1&1&2\\ 0&1&1&1&1&1&2&0\\ 0&1&1&1&1&2&0&0\\ 0&1&1&1&2&0&0&0\\ 0&1&1&2&0&0&0&0\\ 0&1&2&0&0&0&0&0\\ 0&2&0&0&0&0&0&0\\ 2&1&1&1&1&1&1&1 \end{matrix}\quad \text{(d)}$$

Figure 2.2 $\lambda_1=0.3$ $c_1=1.0$ $\mu_1=0.5$
$\lambda_2=0.3$ $c_2=1.0$ $\mu_2=0.5$

```
         0 1 1 1 1 1 1 1        0 1 1 1 1 1 1 1
         0 1 1 1 1 1 1 1        0 1 1 1 1 1 1 1
         0 1 1 1 1 1 1 1        0 1 1 1 1 1 1 1
    (a)  0 1 1 1 1 1 1 1        0 1 1 1 1 1 1 1    (c)
         0 1 1 1 1 1 1 1        0 1 1 1 1 1 1 1
         0 1 1 1 1 1 1 1        0 1 1 1 1 1 1 1
         0 1 1 1 1 1 1 1        0 1 1 1 1 1 1 1
         2 1 1 1 1 1 1 1        2 1 1 1 1 1 1 1

         0 1 1 1 1 1 1 1        0 1 1 1 1 1 1 1
         0 1 1 1 1 1 1 1        0 1 1 1 1 1 1 1
         0 1 1 1 1 1 1 1        0 1 1 1 1 1 1 1
    (b)  0 1 1 1 1 1 1 1        0 1 1 1 1 1 1 1    (d)
         0 1 1 1 1 1 1 1        0 1 1 1 1 1 1 1
         0 1 1 1 1 1 1 1        0 1 1 1 1 1 1 1
         0 1 1 1 1 1 1 1        0 1 1 1 1 1 1 1
         2 1 1 1 1 1 1 1        2 1 1 1 1 1 1 1
```

Figure 2.3 $\lambda_1=0.3$ $c_1=1.0$ $\mu_1=0.6$
$\lambda_2=0.3$ $c_2=1.0$ $\mu_2=0.2$

```
         0 1 1 1 1 1 1 1        0 1 1 1 1 1 1 1
         0 1 1 1 1 1 1 0        0 1 1 1 1 1 1 0
         0 1 1 1 1 1 1 0        0 1 1 1 1 1 1 0
    (a)  0 1 1 1 1 1 1 0        0 1 1 1 1 1 0 0    (c)
         0 1 1 1 1 1 1 0        0 1 1 1 1 1 0 0
         0 1 1 1 1 1 1 0        0 1 1 1 1 1 0 0
         0 1 1 1 1 1 1 0        0 1 1 1 1 1 0 0
         2 1 1 1 1 1 1 1        2 1 1 1 1 1 1 1

         0 1 1 1 1 1 1 1        0 1 1 1 1 1 1 0
         0 1 1 1 1 1 1 0        0 1 1 1 1 0 0 0
         0 1 1 1 1 1 1 0        0 0 0 0 0 0 0 0
    (b)  0 1 1 1 1 1 1 0        0 0 0 0 0 0 0 0    (d)
         0 1 1 1 1 1 1 0        0 0 0 0 0 0 0 0
         0 1 1 1 1 1 1 0        0 0 0 0 0 0 0 0
         0 1 1 1 1 1 1 0        0 0 0 0 0 0 0 0
         2 1 1 1 1 1 1 1        2 0 1 1 1 1 1 1
```

Figure 2.4 $\lambda_1=0.4$ $c_1=1.0$ $\mu_1=0.6$
$\lambda_2=0.2$ $c_2=2.0$ $\mu_2=0.2$

```
    0 1 1 1 1 1 1 1        0 1 1 1 1 1 1 1
    0 1 1 1 1 1 1 0        0 1 1 1 1 1 1 0
    0 1 1 1 1 1 1 0        0 1 1 1 1 1 0 0
(a) 0 1 1 1 1 1 1 0        0 1 1 1 1 1 0 0   (c)
    0 1 1 1 1 1 1 0        0 1 1 1 1 0 0 0
    0 1 1 1 1 1 1 0        0 1 1 1 1 0 0 0
    0 1 1 1 1 1 1 0        0 1 1 1 1 1 0 0
    2 1 1 1 1 1 1 1        2 1 1 1 1 1 1 1

    0 1 1 1 1 1 1 1        0 1 1 1 1 0 0 0
    0 1 1 1 1 1 1 0        0 0 0 0 0 0 0 0
    0 1 1 1 1 1 1 0        0 0 0 0 0 0 0 0
(b) 0 1 1 1 1 1 1 0        0 0 0 0 0 0 0 0
    0 1 1 1 1 1 1 0        0 0 0 0 0 0 0 0   (d)
    0 1 1 1 1 1 1 0        0 0 0 0 0 0 0 0
    0 1 1 1 1 1 1 0        0 0 0 0 0 0 0 0
    2 1 1 1 1 1 1 1        2 0 1 1 1 1 1 1
```

Figure 2.5 $\lambda_1=0.45$ $c_1=1.0$ $\mu_1=0.6$
$\lambda_2=0.2$ $c_2=2.0$ $\mu_2=0.2$

```
    0 1 1 1 1 1 1 0        0 1 1 1 1 1 1 0
    0 1 1 1 1 1 1 0        0 1 1 1 1 1 1 0
    0 1 1 1 1 1 1 0        0 1 1 1 1 1 0 0
(a) 0 1 1 1 1 1 1 0        0 1 1 1 1 0 0 0   (c)
    0 1 1 1 1 1 1 0        0 1 1 1 1 0 0 0
    0 1 1 1 1 1 1 0        0 1 1 1 1 0 0 0
    0 1 1 1 1 1 1 0        0 1 1 1 1 0 0 0
    2 1 1 1 1 1 1 1        2 1 1 1 1 1 1 1

    0 1 1 1 1 1 1 0        0 1 1 0 0 0 0 0
    0 1 1 1 1 1 1 0        0 0 0 0 0 0 0 0
    0 1 1 1 1 1 1 0        0 0 0 0 0 0 0 0
(b) 0 1 1 1 1 1 1 0        0 0 0 0 0 0 0 0
    0 1 1 1 1 1 1 0        0 0 0 0 0 0 0 0   (d)
    0 1 1 1 1 1 1 0        0 0 0 0 0 0 0 0
    0 1 1 1 1 1 1 0        0 0 0 0 0 0 0 0
    2 1 1 1 1 1 1 1        2 0 1 1 1 1 1 1
```

Figure 2.6 $\lambda_1=0.5$ $c_1=1.0$ $\mu_1=0.6$
$\lambda_2=0.2$ $c_2=2.0$ $\mu_2=0.2$

```
        0 1 1 1 1 1 1 1          0 1 1 1 1 1 1 1
        0 1 1 1 1 1 1 0          0 1 1 1 1 1 1 1
        0 1 1 1 1 1 1 0          0 1 1 1 1 1 1 0
(a)     0 1 1 1 1 1 1 0          0 1 1 1 1 1 1 0
        0 1 1 1 1 1 1 0          0 1 1 1 1 1 1 0     (c)
        0 1 1 1 1 1 1 0          0 1 1 1 1 1 0 0
        0 1 1 1 1 1 1 0          0 1 1 1 1 1 0 0
        2 1 1 1 1 1 1 1          2 1 1 1 1 1 1 1


        0 1 1 1 1 1 1 1          0 1 1 1 1 1 1 1
        0 1 1 1 1 1 1 0          0 1 1 1 1 1 1 1
        0 1 1 1 1 1 1 0          0 1 1 1 1 1 1 1
(b)     0 1 1 1 1 1 1 0          0 1 1 1 1 1 1 1
        0 1 1 1 1 1 1 0          0 1 1 1 1 1 1 0     (d)
        0 1 1 1 1 1 1 0          0 1 1 1 1 1 1 0
        0 1 1 1 1 1 1 0          0 1 1 1 1 1 1 0
        2 1 1 1 1 1 1 1          2 1 1 1 1 1 1 1
```

Figure 2.7 $\lambda_1=0.4$ $c_1=1.0$ $\mu_1=0.6$
$\lambda_2=0.3$ $c_2=2.0$ $\mu_2=0.2$

```
        0 1 1 1 1 1 1 1          0 1 1 1 1 1 1 1
        0 1 1 1 1 1 1 1          0 1 1 1 1 1 1 1
        0 1 1 1 1 1 1 1          0 1 1 1 1 1 1 1
        0 1 1 1 1 1 1 1          0 1 1 1 1 1 1 0
(a)     0 1 1 1 1 1 1 1          0 1 1 1 1 1 1 0     (c)
        0 1 1 1 1 1 1 1          0 1 1 1 1 1 1 0
        0 1 1 1 1 1 1 1          0 1 1 1 1 1 1 0
        2 1 1 1 1 1 1 1          2 1 1 1 1 1 1 1


        0 1 1 1 1 1 1 1          0 1 1 1 1 1 1 1
        0 1 1 1 1 1 1 1          0 1 1 1 1 1 1 1
        0 1 1 1 1 1 1 1          0 1 1 1 1 1 1 1
(b)     0 1 1 1 1 1 1 1          0 1 1 1 1 1 1 0     (d)
        0 1 1 1 1 1 1 1          0 1 1 1 1 1 0 0
        0 1 1 1 1 1 1 1          0 1 1 1 1 0 0 0
        0 1 1 1 1 1 1 1          0 1 1 1 1 0 0 0
        2 1 1 1 1 1 1 1          2 0 1 1 1 1 1 1
```

Figure 2.8 $\lambda_1=0.4$ $c_1=1.0$ $\mu_1=0.6$
$\lambda_2=0.2$ $c_2=2.0$ $\mu_2=0.15$

(a)

0	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1

(c)

0	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1

(b)

0	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1

(d)

0	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1
0	1	1	1	1	1	1	0
2	0	1	1	1	1	1	1

Figure 2.9 $\lambda_1=0.4$ $c_1=1.0$ $\mu_1=0.6$
$\lambda_2=0.2$ $c_2=2.0$ $\mu_2=0.1$

the controller performsa dual function; minimize overall cost while regulating overall system capacity. Since queue 2 incurs a higher cost ($c_2>c_1$) then queue 1 and since it's customer is already in the system, it is more advantageous to restrict queue 1's arrivals and service queue 2. At the boundary (7,7), both arrivals can be restricted. Hence since $c_1<c_2$, it is better to serve queue 1. Non-idling policies explain the boundary state (7,0). This same pattern holds true in Figures 2.5(a), (b) and 2.6(a) (b) where the parameters remain the same except the arrival rate, λ_1 is increased.

As the discount parameter β increases, the above situation becomes more pronounced. In Figures 2.4(c) - 2.6(c), the immediate rewards are now competing with the long term benefits. The additional service provided to queue 2 for the states $\{(i,j):i=5,6;j=1,2,...,6\}$ suggests that it is more desirable to restrict system capacity. Note that in Figure 2.6(c), it is better to service queue 2 for state (7,7) than permit another arrival into queue 1. This phenomenon is natural within a traffic flow context. With increased demand on the system, it is better to avoid overall system congestion by restricting entry at the nodes of the network.

For the average cost performance criterion ($\beta=1$), the resulting strategies in Figures 2.4(d) - 2.6(d) are unexplainable. As remarked earlier, the successive approximation algorithm (2.6.1) for $\beta=1$ is not numerical stable. Even for the cases in Figures 2.4(c) - 2.6(c) with $\beta=0.9$, the algorithm did not converge under the given criterion (2.6.2); the maximum iteration count equalled 40. To insure the policy iteration algorithm (2.6.7), (2.6.8) was not yielding misleading results, the initial policy was chosen as the corresponding ($\beta=0.9$) discount cost strategies. Hence, there is a high confidence that the results are numerically correct.

In Figures 2.7 - 2.9, the discounted cost strategies follow the same pattern as noted earlier. Conversely, the average cost policies are unrelated to these results. In several situations, idling-type policies are selected (see e.g. Figure 2.9(d)). Moreover, the switch curve is no longer a connected region. This is an area for future research.

3. STOCHASTIC CONTROL OF TWO PARTIALLY OBSERVED COMPETING QUEUES

3.1 Introduction

While the analysis of the priority assignment strategies for queueing systems with complete observations has received moderate attention [1] - [8], the study of dynamic strategies with partial observations has barely begun [24] - [29]. By dynamic strategies, we mean again a policy which at each time t utilizes the information available up to time t. In Chapter 2, the discussion dealt with a completely observable queueing system. Here, our discussion deals with a partially observed system; specifically only the arrival process to each queue is observed. Consequently, the queue sizes are unobservable to the controller because the departures are not observed. The theory of optimal control of perfectly observed Markov processes is by now well-understood and quite general optimality conditions are available [20], [22], [30] - [32]. When the observations are noisy, the analysis becomes more difficult. In this chapter, we analyze a simple stochastic control problem for two competing queues using noisy observations. Estimation schemes for such a queueing system have been provided previously [33]; here we shall incorporate these estimates in the optimal control scheme.

The problem formulation is similar to the previously discussed complete observations problem. Two parrallel queues are served by a single server with the control selected so that the finite horizon aggregate delay is minimized. The problem is formulated in discrete time with the arrival and departure processes modelled by Bernoulli streams. The arrival and service rates at each queueing station are allowed to depend on the queue size and control values. At each service completion time, a control is selected to decide which queue to service next. 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 each queue. Thus, we have a finite horizon, partially observed stochastic control problem.

The framework in which the problem is formulated is that of a controlled, partially observed Markov process [34] - [38]. The problem of optimal control of a Markov process with incomplete state information can be transformed into a problem of optimal control with complete information [34, Theorem 3]. However the original state space is transformed in the latter problem to be the space of probability distributions. It is of interest though that the information set, upon which the transformed problem is formulated, can be reduced to a "lower dimensional" set without loss of information content. The search for such a sufficient statistic is primarily a problem related to data reduction [36]. While it is possible to show that various functions of the data constitute a sufficient statistic, our attention is focused on a particular one; the sufficient statistic chosen is the conditional probability measure of the state of the Markov chain given the past histories of arrivals and controls.

It is difficult to treat the partially observed control problem unless the control aspects and the statistical estimation aspects can be somewhat separated. The best known example of such a separation occurs in the partially observed, linear regulator problem [37]. In that case the separation principle [39, p. 339] states that the optimal feedback law is a function of the past observations only through the expected value of the state; not a function of any other higher order moments. For the controlled, partially observed Markov process, it can be shown that the conditional distribution of the state given the past histories depends on the control policy only through the most recent value of the control [36, Theorem 1]. In other words, the optimal control law need only depend on the information set via the sufficient statistic and as consequence, the optimal control problem is simplified.

By transforming the partially observed problem into a completely observed problem, the theory of optimal control of Markov processes is readily applicable. In particular, the standard arguments of dynamic programming follow [20], [22]. A particular feature of the transformed partial observed problem is that the "cost to go" function is convex in the state statistic [35, p. 406]. This convexity of the value function is exploited in the work of Smallwood et al [24], [26], Segall[27] and Baras et al [29].

The two competing queuesproblem is analyzed within the aforementioned framework. Our starting point is the joint statistics (assumed known) of the observed arrival processes and of the transitions of the chain. By this approach, the observations, modelled as discrete time 0-1 point processes, have rates [18] that are influenced by the chain states. An identical formulation in continuous time was applied by Segall[27] for a dynamic file assignment problem. The dependency of arrival and departure rates on queue sizes was first considered in the queueing system context by Jackson [40]. By our joint statistic approach, a slight modification is required to the existing framework of a controlled, partially observed Markov process. The modification is necessary due to the special relationship between the observations and state transitions in such queueing systems.

The classical tools of Bayes rule and dynamic programming suffice for our analysis [24], [41]. We show that the "one-step" predicted density of the state, given the point process observations, is a sufficient statistic for the control. Thus, the optimal server allocation strategy depends on the observed arrivals through this statistic which is computed recursively via the filter-predictor equations [33]. It is shown that the optimal strategy in addition has the form of the separation principle.

The dynamic programming methodolgy is simplified as a consequence of this separation property. In particular, all the necessary computations needed to implement the optimal strategy can be performed off-line. Due to the linearity of the instantaneous cost, the optimal value function is piecewise linear in the state statistic. The piecewise linearity of the value function was first observed by Smallwood and Sondik [24] in a machine replacement problem. The resulting off-line computations can be stored in an elementary way in order to facilitate the on-line implementation of the strategy. The results are similar to those of Chapter 2 (see Section 2.3, Theorem 2.3.10 and Remark 2.3.6). The methods presented here lead to a numerical treatment; analytical solutions for this problem have not been obtained. Our results serve as a basis for further analysis of the two competing queue problem: evaluation of suboptimal policies and alternative performance objectives such as infinite horizon discount and average cost per unit time aggregate delays.

This chapter is organized as follows. In Section 3.2, we formulate a simple two competing queue problem. Extensions to more general models of such queueing systems are then presented. The basic questions to be studied in this chapter are discussed. The data reduction algorithm to obtain a sufficient statistic is developed in Section 3.3. The general filtering and prediction results for the class of stochastic systems whose observations are influenced by state transitions are presented. We then apply these results to the queueing problem under discussion. In Section 3.4, the optimality equations characterizing the solution of the finite horizon, partial observed problems are presented. Again, the extension of these methods to the multi-class case of [9] - [13] is theoretically straightforward, but computationally burdensome. Our development simplifies the on-line solution of the optimal policy. In Section 3.5, we present computations and evaluations of the strategies obtained, as the theory is applied to a simple problem.

3.2 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 Figure 3.1 below. Time is divided into uniform time slots; that is,we adopt a discrete time formulation. Customers arrive into stations 1 and 2 according to two independent Bernoulli streams with constant rates λ_1, λ_2 respectively.

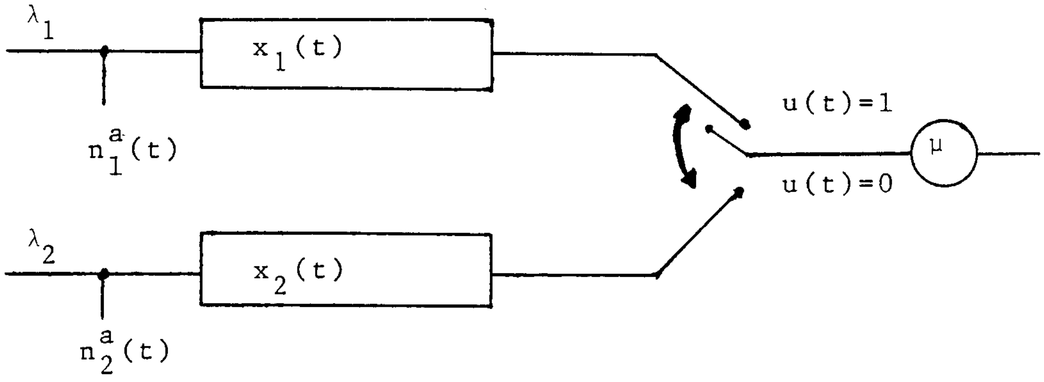


Figure 3.1. The partial observation control problem

If we let $\{n_i^a(\cdot);\ i = 1,2\}$ denote the two arrival processes, it is clear that they are discrete time 0-1 point processes:

$$n_i^a(t) = \begin{cases} 1, & \text{if an arrival occurs in the } t^{th} \text{ time slot of queue } i \\ 0, & \text{otherwise} \end{cases} \qquad (3.2.1)$$

Our convention is that the t^{th} time slot is the half open interval $[t,t+1)$. The rates of the arrival processes are given by [18]:

$$\lambda_i = \Pr\{n_i^a(t)=1\} \qquad , i=1,2 \quad . \qquad (3.2.2)$$

The two queues compete for the service of a server whose service completions follow a Bernoulli stream with constant rate μ. If we let $\{n^d(t)\}$ be the service process, whenever the server is connected to one of the two queues (when it is nonempty) then

$$\mu = \Pr\{n^d(t) = 1\} \qquad (3.2.3)$$

Let $x_i(t)$ denote the number of customers in the i^{th} queue during the time slot t, with the customer in service included. The control to be selected is 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 queueing 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 instantaneous cost at time t with queues $x_1(t)$, $x_2(t)$ is given by:

$$c_1 x_1(t) + c_2 x_2(t). \tag{3.2.4}$$

For a finite horizon of length T, we wish to minimize the <u>average aggregate delay</u>

$$J_f^{\gamma} = E[\sum_{t=0}^{T} (c_1 x_1(t) + c_2 x_2(t))] \tag{3.2.5}$$

The queue sizes are not observable, since only the arrival processes $\{n_i^a(t);\ i=1,2\}$ are observable to the controller. The server time allocation strategy is to be selected, so that (3.2.5) is minimized, and is allowed to be a function of the past histories of the arrival processes and the past history of the control.

This simple priority assignment problem is motivated by the related work in urban traffic control problems [42] - [44], [33]. This problem is the simplest in a sequence of such problemsin the urban traffic context. The controller represents the traffic light, the two queues correspond to the two approaches at a traffic intersection and the arrival processes observed are the outputs of street loop detectors. Therefore the problem we have described above, models in an elementary way the critical intersection traffic control problem [45]. Traffic activated control laws lead to the dependency of the control value $u(\cdot)$ on the past histories of arrival and control processes. The filtering and prediction results of queue sizes in [33], [42] are incorporated into the results of this chapter. The interested reader is directed to [33], [42] for further details on the development of queue models appropriate to urban traffic problems. Similar priority assignment problems appear in computer networks, where one allocates files according to demand [27] or in satellite communication networks [28] where one controls retransmission laws according to traffic load of the network.

The priority problem is easier to analyze for an unbounded queueing system. The results obtained can be applied to more general queueing models than the one described above. In particular, to represent effects of congestion, one can let the arrival and service rates depend on the queue size. Consequently, we shall consider the queueing system under the following assumptions:

$$\begin{aligned}&\Pr\{n_i^a(t)=1 \mid \text{past histories of } x_1,\ x_2,\ n_1^a, n_2^a, \text{ and } u, \text{ up to time } t\}\\ &=\Pr\{n_i^a(t)=1 \mid x_i(t)=k,\ u(t)=v\} = \lambda_i(t,k,v) \qquad ;\ i=1,2\end{aligned} \tag{3.2.6}$$

and

$$\begin{aligned}&\Pr\{n_i^d(t)=1 \mid \text{past histories of } x_1, x_2, n_1^a, n_2^a \text{ and } u, \text{ up to time } t\}\\ &=\Pr\{n^d(t)=1 \mid x_i(t)=k, u(t)=v\} = \mu_i(t,k,v) \qquad ;\ i=1,2\end{aligned} \tag{3.2.7}$$

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

$$\mu_i(t,0,v)=0 \qquad \text{for all } t,v\ ;\ i=1,2\ . \tag{3.2.8}$$

For the simple problem described in Figure 3.1,

$$\lambda_i(t,k,v) = \lambda_i \qquad \text{for all } t,k,v \quad ; \; i=1,2 \qquad (3.2.9)$$

and

$$\begin{aligned} \mu_1(t,k,v) &= \mu \; v && \text{for all } t,v \quad ; \; k\neq 0 \\ \mu_2(t,k,v) &= \mu \; (1-v) && \text{for all } t,v \; ; \; k\neq 0 \end{aligned} \qquad (3.2.10)$$

In most applications the queues are bounded in size. If we let $\{N_i;i=1,2\}$ denote the maximum queue size for each queue, then in addition to (3.2.8) we must have

$$\lambda_i(t,N_i,v)=0 \qquad \text{for all } t,v \; ; \; i=1,2 \qquad (3.2.11)$$

For the simple problem of Figure 3.1, (3.2.11) implies that

$$\lambda_i(t,k,v) = \begin{cases} \lambda_i, & k\neq 0, \;\; \text{for all } t,v \quad ; \; i=1,2 \\ 0 \;, & k=N_i, \; \text{for all } t,v \; ; \; i=1,2 \end{cases} \qquad (3.2.12)$$

In queueing models such as the preceding, there is a direct 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. In other words, 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:

$$S_{ij}(t,v,\psi) = \Pr\{x(t+1)=j,y(t)=\psi \,|\, x(t)=i,u(t)=v\} \qquad (3.2.13)$$
$$\text{for } i,j\varepsilon\chi = Z=\{0,1,2,\ldots\}, \psi\varepsilon Y=\{0,1\}, v\varepsilon U =\{0,1\}$$

This is the starting point of our development of filtering and prediction results in Section 3.3. Moreover, the two queues are <u>linked only through the current control value</u>;

a fact that is made precise in the subsequent section.

Finally in modelling the admissible control strategies, one may wish to allow the control during the t^{th} time slot to depend on the observations during the t^{th} time slot or not. This is clearly a modelling question. Realistically, it is better to allow u(t) to depend on the observations only up to time t-1. However in some applications, it may be important to know the tradeoff in complexity and performance, when the former condition holds. For completeness we shall consider both cases here.

It is important to emphasize 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 as was apparent in Chapter 2.

3.3 Filtering and Prediction

General Results

In this section, we briefly review the filtering and prediction formulas for the state of a partially observed, controlled Markov chain that influencesthe observations [24], [34], [36], [46]. Since our problem is formulated in discrete time, these results can be derived in an elementary fashion using Bayes rule [24]. These general results are applied to the case of primary interest: discrete time, 0-1 point process observations influenced by the state transitions of the chain. The explicit filtering and and prediction formulas for the queueing problem under study are equivalent to the ones in [33] obtained by martingale techniques. For an unbounded system, the Markov chain state space, χ is countable while for a bounded system the state space is finite.

We assume the joint statistics of the observations and state transitions are known and are given by:

$$S_{ij}(t,v,\psi)=\Pr\{x(t+1)=j,y(t)=\psi|x(t)=i,u(t)=v\}$$
$$\text{for } i,j\in\chi,\ \psi\in Y,\ v\in U \qquad (3.3.1)$$

The inclusion of controls in the description of these statistics is meant to emphasize that the filtering and prediction formulas derived here are used in subsequent sections for the solution of the stochastic control problem with partial observations. For our purposes, it suffices to assume that the output process, $\{y(\cdot)\}$and the control process, $\{u(\cdot)\}$take values in finite sets, denoted Y and U, respectively. For the queueing system introduced above (3.2.1) - (3.2.12), the output set, Y = {0,1} while the control set, U = {0,1} for the single server problem or U = $\{0,1\}^m$ (m^{th}-fold Cartesian product) in the case of m servers. The state space χ in (3.3.1) for the unbounded queue is the set of positive integers, Z = {0,1,2,...} while for the bounded queue, it has the form:

$$\chi = Z = \{0,1,2,...,N\} \quad ; N = \text{maximum queue size} \qquad (3.3.2)$$

For the queueing problems of interest, the fundamental modelling assumption is that the joint statistics of observations and state transitions are influenced <u>only</u> by the current state and control values. Specifically, let

x^t, y^{t-1}, u^t denote respectively the past histories of the state, observation and control processes up to time t

$$x^t = \{x(s);\ s=1,2,\ldots,t\}$$

$$y^{t-1} = \{n_i^a(s);\ s=0,1,\ldots,t-1;\ i=1,2\} \qquad (3.3.3)$$

$$u^t = \{u(s);\ s=1,2,\ldots,t\}.$$

Then we assume that the following "semi-Markovian" assumption holds:

$$\text{(SM)}\quad \Pr\{x(t+1)=j, y(t)=\psi \mid x^t, y^{t-1}, u^t\}$$

$$= \Pr\{(x(t+1)=j, y(t)=\psi \mid x(t)=i,\ u(t)=v\} \qquad (3.3.4)$$

$$\text{for all } i,j \varepsilon X,\ \psi \varepsilon Y,\ v \varepsilon U$$

This assumption is consistent with a "stochastic dynamical system" model of queues, which as discussed in [33] (see equation (2.11), p. 13) is valid under very general circumstances. It is easy to see that on the basis of (3.3.4), one can describe the partially observed queue as a probabilistic automaton [47].

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

$$z^t = (y^t, u^{t-1}) \qquad (3.3.5)$$

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

$$\gamma = (g_0, g_1, \ldots, g_t, \ldots), \qquad (3.3.6)$$

where each g_t is a function

$$g_t : Y_t x U_t \to U$$

$$z^t \to u(t) = g_t(z^t) \qquad (3.3.7)$$

The policies in Γ are called nonanticipative, following standard terminology [31]. Recall in Section 3.2, we noted the difficulties associated with whether or not to allow the control at time t, u(t) to depend on the observation at time t, y(t). This problem is intrinsic to the influence of state transitions on the observations, which in the case of a queueing system are arrivals or departures and therefore convey information about the queue size (the state). One thus has the choice, in the specific problem of two competing queues described in the preceding section, either to utilize in the decision process at time t, the arrival observation at time t, or not. This is a matter of choice and we wish to develop the methodology keeping both options. Thus we shall consider a subclass of Γ as admissible policies. Namely, let Γ_0 be the subset of Γ consisting of policies γ of the form (3.3.6), where each g_t is a function of the form:

$$g_t : Y_t \text{ x } U_t \to U$$

$$\zeta^{t-1} \to u(t) = g_t(\zeta^{t-1}) = g_t(z^{t-1}, u(t-1)). \qquad (3.3.8)$$

and

$$\zeta^{t-1} = (y^{t-1}, u^{t-1}) \qquad (3.3.9)$$

We call the policies in Γ_0, strictly nonanticipative. Wherever our results need modifications as a consequence of restricting admissible policies to be strictly nonanticipative, we shall indicate so explicitly and give the necessary modifications. Following standard usage [36], we call z^t or ζ^{t-1} the <u>information vector available at time t</u>.

From the joint statistics (3.3.1), it follows that the transition probabilities of the chain are given by:

$$p_{ij}(t,v) \equiv \Pr\{x(t+1)=j \mid x(t)=i, u(t)=v\} = \sum_{\psi \varepsilon Y} S_{ij}(t,v,\psi) \ . \tag{3.3.10}$$

Similarly the output statistics given the state are given by:

$$\lambda_i(\psi,t,v) \equiv \Pr\{y(t)=\psi \mid x(t)=i, u(t)=v\} = \sum_{j \varepsilon X} S_{ij}(t,v,\psi) \ . \tag{3.3.11}$$

There is a consistency requirement on the statistics described by the matrix S. This requirement is due to the nature of the link between observations y(·) and state transitions that was discussed earlier. Namely from (3.3.1), it appears that the value of the control at time t influences the statistics of the observations y(t) at time t.
On the other hand, since we wish to analyze nonanticipative policies as well, we allow the value of the control u(t) at time t to depend on the value of the observation y(t) at time t. To avoid the apparent difficulty with existence of a causal relationship between u(t) and y(t) (which is necessary in studying nonanticipative control policies) we require that S satisfies the following consistency condition:

Consistency condition: The output statistics at time t (3.3.11), induced by the joint statistics of output and state transitions (3.3.1) do not depend on the value of the control at time t; specifically

$$\sum_{j \varepsilon \chi} S_{ij}(t,v,\psi) = \text{independent of } v \text{ for all } i,\psi,t. \tag{3.3.12}$$

For queueing systems with server control, the consistency condition (3.3.12) is always satisfied as will be demonstrated later in this section. As a consequence of (3.3.12), the control argument, v is dropped in the notation of $\lambda_i(\cdot)$ as defined in (3.3.11).

Given a control policy $\gamma \varepsilon \Gamma$, the conditional probabilities of interest in the control problem (3.2.1) - (3.2.12) are the probabilities of the state given the information vector; specifically we define the row vector probabilities (possibly infinite dimensional) for all $i \varepsilon \chi$ to be

$$\Pi^{\gamma}_{t+1|t}(i) = \Pr\{x(t+1) = i|\zeta^t\}$$

and

$$\Pi^{\gamma}_{t|t}(i) = \Pr\{x(t) = i|\zeta^t\} \tag{3.3.13}$$

The supercript γ in (3.3.13) indicates the state and control trajectories induced by the policy γ. For notational convenience, we introduce the additional probabilities:

$$p^{\gamma}_{t+1|t}(j|i,\zeta^t) = \Pr\{x(t+1) = j|x(t) = i, \zeta^t\}$$

(3.3.14)

$$p^{\gamma}(\psi|i,v,\zeta^{t-1}) = \Pr\{y(t) = \psi|x(t) = i,u(t) = v,\zeta^{t-1}\}$$

(3.3.15)

Then to establish the sufficient statistic for the control problem (3.2.1) - (3.2.12), we proceed with the following sequence of lemmas:

Lemma 3.3.1. For all $\gamma\varepsilon\Gamma$, $\zeta^t\varepsilon Y_t xU_t$ and $i,j\varepsilon X$ we have

(a) $$p^{\gamma}_{t+1|t}(j|i,\zeta^t) = \frac{S_{ij}(t,u(t),y(t))}{\sum_{j\varepsilon X} S_{ij}(t,u(t),y(t)} \qquad (3.3.16)$$

provided the denominator in (3.3.16) is positive.

(b) $$p^{\gamma}(\psi|i,v,\zeta^{t-1}) = \lambda_i(\psi,t) \qquad (3.3.17)$$

where $\lambda_i(\cdot)$ is defined in (3.3.11)

Proof: For (a), by Bayes rule

$$p^{\gamma}_{t+1|t}(j|i,\zeta^t) = \frac{p^{\gamma}_{t+1|t}(j,y(t)|i, y^{t-1},u^t)}{p^{\gamma}(y(t)|i,y^{t-1},u^t)}$$

$$= \frac{S_{ij}(t,u(t),y(t))}{\sum_{j\varepsilon X} S_{ij}(t,u(t),y(t))} \qquad (3.3.18)$$

provided $p^{\gamma}(y(t)|i,y^{t-1},u^t)>0$. The second equality in (3.3.18) follows from (3.3.1) and (3.3.4).

For (b), we observe from (3.3.4) for any policy $\gamma\varepsilon\Gamma$

$$p^{\gamma}(\psi|i,v,\zeta^{t-1}) = p^{\gamma}(\psi|i,v)$$

and hence the result (3.3.17) follows from (3.3.11) and (3.3.12). QED.

Remark 3.3.1. Observe in (a) since $S_{ij}(\cdot)\geq 0$ when $p^{\gamma}(y(t)|i,y^{t-1},u^{t})$ is zero for some $i\varepsilon X, y(t)\varepsilon Y$ (recall that due to the consistency condition (3.3.12), $p^{\gamma}(y(t)|i,y^{t-1},u^{t})$ does not depend on u(t)), then

$$S_{ij}(t,u(t),y(t))=0 \quad \text{for all } j\varepsilon X \tag{3.3.19}$$

In this case for consistency, we let $p^{\gamma}_{t+1|t}(j|i,\zeta^{t})=\delta_{ij}$ in (3.3.16). To simplify later computations, we introduce the following matrix:

$$M_{ij}(t,v,\psi) = \begin{cases} \dfrac{S_{ij}(t,v,\psi)}{\sum\limits_{j\varepsilon X} S_{ij}(t,v,\psi)} & , \text{ if } \sum\limits_{j\varepsilon X} S_{ij}(t,v,\psi)>0 \\ \delta_{ij} & , \text{ otherwise} \quad \text{for } i,j\varepsilon X \end{cases} \tag{3.3.20}$$

Then Lemma 3. 3.1(a) can be restated as

$$p^{\gamma}_{t+1|t}(j|i,\zeta^{t}) = M_{ij}(t,u(t),y(t)) \tag{3.3.21}$$

Remark 3.3.2. By Lemma 3.3.1, the filtering and prediction formulas for the conditional probabilities of interest (3.3.13) can be derived. First,

$$\begin{aligned}\Pi^{\gamma}_{t+1|t}(j) &= p^{\gamma}_{t+1|t}(j|\zeta^{t}) = \sum_{i\varepsilon X} p^{\gamma}_{t+1|t}(j,x(t)=i|\zeta^{t}) \\ &= \sum_{i\varepsilon X} p^{\gamma}_{t+1|t}(j|i,\zeta^{t}) \cdot p^{\gamma}_{t|t}(i|\zeta^{t}) \quad \text{(by Bayes rule)} \\ &= \sum_{i\varepsilon X} M_{ij}(t,u(t),y(t))\cdot\Pi^{\gamma}_{t|t}(i) \quad \text{(by (3.3.21) and (3.3.13))}.\end{aligned}$$

or in matrix notation:

$$\Pi^{\gamma}_{t+1|t} = \Pi^{\gamma}_{t|t}\, M(t,u(t),y(t)) \tag{3.3.22}$$

Similarly,

$$\Pi^{\gamma}_{t+1|t+1}(i) = p^{\gamma}_{t+1|t+1}(i|\zeta^{t+1}) = p^{\gamma}_{t+1|t+1}(i|z^{t+1})$$

(by (3.3.7)

$$= \frac{p_{t+1|t+1}(i,z^{t+1})}{\sum\limits_{i\varepsilon X} p^{\gamma}_{t+1|t+1}(i,z^{t+1})} \quad \text{(by Bayes rule)} \tag{3.3.23}$$

By Lemma 3.3.1(b), we have

$$p^{\gamma}_{t+1|t+1}(i,z^{t+1}) = \lambda_i(y(t+1),t+1)\cdot p^{\gamma}_{t+1|t}(i|\zeta^{t})\cdot p^{\gamma}(\zeta^{t}) \tag{3.3.24}$$

By combining (3.3.23), (3.3.24) and simplifying, we have

$$\Pi^{\gamma}_{t+1|t+1}(i) = \frac{\lambda_i(y(t+1),t+1)\cdot\Pi^{\gamma}_{t+1|t}(i)}{\prod\limits_{i\varepsilon X} \lambda_i(y(t+1),t+1)\cdot\Pi^{\gamma}_{t+1|t}(i)} \tag{3.3.25}$$

or in matrix notation:

$$\Pi^{\gamma}_{t+1|t+1} = \frac{\Pi^{\gamma}_{t+1|t} D(t,y(t+1))}{\Pi^{\gamma}_{t+1|t} D(t,y(t+1))e} \tag{3.3.26}$$

where the diagonal matrix

$$D(t,\psi) = \mathrm{diag}\{\lambda_i(\psi,t)\} \qquad \text{for all } i\varepsilon X \tag{3.3.27}$$

and the column vector

$$e = [1,1,\ldots,1,\ldots]^T \tag{3.3.28}$$

The computations (3.3.22), (3.3.26) - (3.3.28) are slight modifications of existing results (e.g. see [19]). Indeed, $S_{ij}(t,v,\psi)$ in (3.3.1) has the form $p_{ij}(v)\cdot r^{v}_{j\psi}$ in the notation of [19]. Although this product form for $S(\cdot)$ was considered in [19], the arguments of [19] do <u>not</u> make use of this fact. The filtering and prediction formulas, (3.3.22) and (3.3.26)-(3.3.28) respectively lead to the result [36, Theorem 1]:

<u>Lemma 3.3.2</u>. The conditional probability vector $\Pi^{\gamma}_{t+1|t+1}$, $\Pi^{\gamma}_{t+1|t}$, of (3.3.13) does <u>not</u> depend on $\gamma\epsilon\Gamma$; specifically it depends only on the values of the control not on the control policy. Furthermore, the conditional probabilities are computed recursively by (3.3.22), (3.3.26).

<u>Proof</u>: The recursive computation of $\Pi^{\gamma}_{t+1|t+1}$, $\Pi^{\gamma}_{t+1|t}$ follow from Remark 3.3.2. For the independence with respect to the control policy, let Π_0 denote the row vector of the initial probabilities of the state x(0). Then by (3.3.26) for t=0, we have

$$\Pi_{0|0} = \frac{\Pi_0 D(0,y(0))}{\Pi_0 D(0,y(0))e} \qquad (3.3.29)$$

It is now obvious from (3.3.22) (3.3.26) that $\Pi^{\gamma}_{t+1|t+1}$,$\Pi^{\gamma}_{t+1|t}$ depend <u>only</u> on the values of the controls. QED.

As a consequence of Lemma 3.3.2, we shall drop the superscript γ from (3.3.13), (3.3.22) and (3.3.26), for the remainder of our discussion. Following [36], [41] we consider the concept of information states; specifically [36, pp. 583-585].

<u>Definition 3.3.1</u>. Let τ^t denote the information vector available at time t. Then a vector $\Phi(t)$ is called an

information state at time t for the controlled stochastic system described by (3.3.1) if

(a) $\Phi(t)$ can be evaluated from τ^t

(b) There exists a function $T_t(\cdot)$ such that

$$\Phi(t+1) = T_t(\Phi(t),\ \tau^{t+1}\backslash\tau^t) \tag{3.3.30}$$

where $\tau^{t+1}\backslash\tau^t$ denotes the new information generated at time t+1.

We can now state the following theorem, which is a slight modification of well-known results [24], [34], [36], [41].

Theorem 3.3.3. For the system described by (3.3.1), (3.3.4)

(a) if $\tau^t=z^t$ as defined in (3.3.5), then $(\Pi_{t|t},y(t))$ is an information state at time t.

(b) if $\tau^t=\zeta^{t-1}$ as defined in (3.3.9), then $\Pi_{t|t-1}$ is an information state at time t.

Proof: For (a), $\tau^{t+1}\backslash\tau^t=(y(t+1),u(t))$ so that combined with (3.3.22), (3.3.26)

$$\Pi_{t+1|t+1} = \frac{\Pi_{t|t}\cdot M(t,u(t),y(t))\cdot D(t+1,y(t+1))}{\Pi_{t|t}\cdot M(t,u(t),y(t))\cdot D(t+1,y(t+1))e} \tag{3.3.31}$$

the result follows.

For (b), $\tau^{t+1}\backslash\tau^t = (y(t),u(t))$ so that combined with (3.3.22), (3.3.26)

$$\Pi_{t+1|t} = \frac{\Pi_{t|t-1}\cdot D(t,y(t))\cdot M(t,y(t),y(t))}{\Pi_{t|t-1}\cdot D(t,y(t))e} \tag{3.3.32}$$

the result follows QED.

Remark 3.3.3. From (3.3.31) (3.3.32), the difference equations for the "unnormalized" conditional probabilities are easily derived; specifically they have the form:

$$\rho_{t+1|t+1} = \rho_{t|t}\cdot M(t,u(t),y(t))\cdot D(t+1,y(t+1)) \qquad (3.3.33)$$

$$\rho_{t+1|t} = \rho_{t|t-1}\cdot S(t,u(t),y(t)) \qquad (3.3.34)$$

where

$$\rho_{0|0} = \Pi_{0|0} \text{ and } \rho_{0|-1} = \Pi_0 \qquad (3.3.35)$$

Then

$$\Pi_{t+1|t+1} = \frac{\rho_{t+1|t+1}}{\rho_{t+1|t+1}e} \text{ and } \Pi_{t|t} = \frac{\rho_{t|t}}{\rho_{t|t}e} \qquad (3.3.36)$$

To obtain (3.3.34), observe that from (3.3.11), (3.3.27)

$$D(t,y(t))e = S(t,u(t),y(t))e \qquad (3.3.37)$$

and from (3.3.11), (3.3.20)

$$D(t,y(t))\cdot M(t,u(t),y(t)) = S(t,u(t),y(t)) \qquad (3.3.38)$$

As a result of Theorem 3.3.3 and (3.3.33) - (3.3.35), we have established the following corollary.

Corollary 3.3.4. For the system described by (3.3.1), (3.3.4)

(a) if $\tau^t=z^t$ as defined in (3.3.5), then $(\rho_{t|t},y(t)$ is an information state at time t.

(b) if $\tau^t = \zeta^{t-1}$ as defined in (3.3.9), then $\rho_{t|t-1}$ is an information state at time t.

The importance of considering the "unnormalized" versions (3.3.33), (3.3.34) rest primarily on their linearity in state dynamics (as compared to the nonlinear equations (3.3.22), (3.3.26)) as it has been recently emphasized in nonlinear filtering studies [48], [49]. The recursions (3.3.33), (3.3.34) are a slight modification of the results of Rudemo [46].

Application to Queueing Models in Discrete Time

The filtering and prediction results developed above are now applied to the general queueing system described by (3.2.6) - (3.2.13). Recall by (3.2.6), (3.2.7) that the arrival and departure rates are allowed to depend on the queue size. Using martingale techniques, similar results have been obtained by Baras et al [33], [42].

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

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

and (3.3.39)

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

We assume that time discretization is such that the probability of more than one arrival or departure in a single time slot is zero. The arrival and departure point processes $n^a(t)$, $n^d(t)$, respectively are defined as follows:

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

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

Consequently by (3.3.39), we have

$$\lambda(t,i,v) = \Pr\{n^a(t)=1|x(t)=i,u(t)=v\}.$$

$$\mu(t,i,v) = \Pr\{n^d(t)=1|x(t)=i,u(t)=v\} \qquad (3.3.40)$$

The queue size during the t^{th} time slot is denoted by x(t). Here we assume that the queue size is controlled by controlling the departure and (or) the arrival rates. The quantity $\mu(t,i,v)$ in (3.3.40) is also referred to as service rate. Finally, we make the usual assumption (see e.g. [33]) that the departure and arrival processes are independent of each other given the queue size and the control.

The problem of interest is the partially observed queue as described in Section 2; specifically the arrival process $\{n^a(t)\}$ is observed while the departure process $\{n^d(t)\}$ is not observed. Consequently, we have that the observation process is given by

$$y(t) = n^a(t), \qquad t=0,1,2,\ldots,T \qquad (3.3.41)$$

Following the framework of (3.3.1) (3.3.4), we need to specify the joint statistics of the observations and state transitions:

$$\begin{aligned} S_{ij}(t,v,1) &= \Pr\{x(t+1)=j,n^a(t)=1|x(t)=i,u(t)=v\} \\ &= \Pr\{x(t+1)=j|n^a(t)=1,x(t)=i,u(t)=v\} \\ &\cdot \Pr\{n^a(t)=1|x(t)=i,u(t)=v\} \qquad \text{for all } i,v \end{aligned}$$

Therefore

$$S_{ii}(t,v,1) = \lambda(t,i,v)\ \mu(t,i,v)$$

$$S_{i,i+1}(t,v,1) = \lambda(t,i,v)\ (1-\mu(t,i,v)) \qquad (3.3.42)$$

$$S_{ij}(t,v,1) = 0 \quad , \text{ elsewhere} \quad \text{for all } i,j\varepsilon X;v\varepsilon U$$

Similarly,

$$S_{ij}(t,v,0) = \Pr\{x(t+1)=j,n^a(t)=0|x(t)=i,u(t)=v\}$$

$$= \Pr\{x(t+1)=j|n^a(t)=0,x(t)=i,u(t)=v\}$$

$$\cdot \Pr\{n^a(t)=0|x(t)=i,u(t)=v\} \qquad \text{for all } i,v$$

and therefore

$$S_{ii}(t,v,0) = (1-\lambda(t,i,v,))\ (1-\mu(t,i,v))$$

$$S_{i,i-1}(t,v,0) = (1-\lambda(t,i,v))\ \mu(t,i,v) \qquad (3.3.43)$$

$$S_{ij}(t,v,0) = 0 \quad , \text{ elsewhere} \quad \text{for all } i,j\varepsilon X;v\varepsilon U$$

Observe that the special link between the observations and the state transitions as discussed in Section 2 is captured in the description of S in (3.3.42), (3.3.43). In the case of a queue evolving without bounds, the only constraint imposed on $\lambda(t,i,v)$, $\mu(t,i,v)$ is that

$$\mu(t,0,v) = 0 \quad , \quad \text{for all } t,\ v\varepsilon U. \qquad (3.3.44)$$

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 we require

$$\lambda(t,N,v) = 0 \quad , \quad \text{for all } t,\ v\varepsilon U. \qquad (3.3.45)$$

The matrix of state transition probabilities, according to (3.3.10), are computed from the description of S (3.3.42), (3.3.43); in particular they coincide with our earlier model (2.2.7) for state independent rates. The point to be noted here is that the description of S is more appropriate when studying partially observed queueing systems.

In our discussion of nonanticipative and strictly nonanticipative policies (3.3.5) - (3.3.9), we stated that, for queueing system which are controlled by controlling the departure (or service) rate, the consistency condition (3.3.12) holds. Clearly, the arrival rate $\lambda(\cdot)$ in (3.3.40) is independent of v and from (3.3.42), (3.3.43) it follows:

$$\begin{aligned} &\sum_{j\varepsilon X} S_{ij}(t,v,1) = \lambda(t,i) \\ &\sum_{j\varepsilon X} S_{ij}(t,v,0) = (1-\lambda(t,i)) \qquad \text{for all } i\varepsilon X \end{aligned} \qquad (3.3.46)$$

Hence the consistency condition (3.3.12) is satisfied for the system under discussion. In (3.3.46) and for the remainder of our discussion, we shall drop the argument v from the arrival rate, $\lambda(\cdot)$ in (3.3.40).

To complete the characterization of the filtering and prediction results (3.3.31), (3.3.32) from (3.3.20), (3.3.42), (3.3.43) it follows:

$$\begin{aligned} &M_{ii}(t,v,1) = \mu(t,i,v) \\ &M_{i,i+1}(t,v,1) = 1-\mu(t,i,v) \\ &M_{ij}(t,v,1) = 0 \qquad \text{, elsewhere} \quad \text{for all } i,j\varepsilon X; v\varepsilon U \end{aligned} \qquad (3.3.47)$$

and

$$
\begin{aligned}
&M_{ii}(t,v,0) = 1-\mu(t,i,v) \\
&M_{i,i-1}(t,v,0) = \mu(t,i,v) \qquad (3.3.48) \\
&M_{ij}(t,v,0) = 0 \quad , \text{ elsewhere} \quad \text{for all } i,j\varepsilon X; v\varepsilon U
\end{aligned}
$$

Furtheremore, the matrix D introduced in (3.3.27) becomes

$$
\left.\begin{aligned}
&D(t,1) = \mathrm{diag}\{\lambda(t,i)\} \\
&D(t,0) = \mathrm{diag}\{1-\lambda(t,i)\} \quad \text{for all } i\varepsilon X
\end{aligned}\right\} \qquad (3.3.49)
$$

By substituting (3.3.47) - (3.3.49) into (3.3.3)), (3.3.32) we obtain

$$
\Pi_{t|t}(i) = \begin{cases} \dfrac{\lambda(t,i)\ \Pi_{t|t-1}(i)}{\sum\limits_{j\varepsilon X} \lambda(t,j)\ \Pi_{t|t-1}(j)} & , \text{ if } n^a(t) = 1 \\[3ex] \dfrac{(1-\lambda(t,i))\ \Pi_{t|t-1}(i)}{\sum\limits_{j\varepsilon X} (1-\lambda(t,j))\ \Pi_{t|t-1}(j)} & , \text{ if } n^a(t) = 0 \end{cases}
$$

$$
\Pi_{t+1|t}(i) = \begin{cases} \mu(t,i,v)\ \Pi_{t|t}(i) \\ +\ (1-\mu(t,i-1,v))\ \Pi_{t|t}(i-1) & , \text{ if } n^a(t) = 1 \\[2ex] (1-\mu(t,i,v))\ \Pi_{t|t}(i) \\ +\ \mu(t,i+1,v))\ \Pi_{t|t}(i+1) & , \text{ if } n^a(t)=0 \end{cases} \qquad (3.3.51)
$$

By elementary techniques, we have obtained the filtering and prediction formulas for queue size previously reported in [33, equation (3.14), (3.15)]. Futhermore from (3.3.33), (3.3.34), we have established the unnormalized versions of these filtering and prediction formulas.

Now we shall apply the filtering and prediction results developed above to the two competing queue problem (3.2.1) - (3.2.13). Each queue is described as in (3.3.39) - (3.3.45) where for notational convenience the superscript or subscript 1 or 2 refers to the parameters of the respective queue. In particular, the matrices $\{S^i(\cdot), M^i(\cdot), D^i(\cdot); i=1,2\}$ are described respectively by (3.3.42) - (3.3.43), (3.3.47) - (3.3.48) and (3.3.49). The observation process for the combined two queue system is given by:

$$y(t) = (y_1(t), y_2(t)) = (n_1^a(t), n_2^a(t)) \quad \text{for each } t = 0,1,2,\ldots,T \tag{3.3.52}$$

Again, the control is applied through the departure rates of each queue. Hence, the consistency condition (3.3.12) holds as was demonstrated in (3.3.46). Although we can accommodate more general models, we shall assume the following independence condition to simplify the computations:

> Independence condition: Each queue's transitions and arrival process are conditionally independent given the current queue sizes and current control value.

In most practical applications (e.g. urban traffic control, computer or communication networks) this condition is usually satisfied. Basically it expresses the observed fact that for the combined two queue system with control provided via the service rate, the basic coupling between the two arrival processes is through the control and through each queue's evolution in response to the control.

For the combined queue system, the state process is given by:

$$x(t) = (x_1(t),\ x_2(t))$$

where the state space $X = ZxZ$ and Z denotes the set of positive integers. The combined joint statistics of the observations and state transitions is given by:

$$S_{i_1 i_2; j_1 j_2}(t,v,\psi) = \Pr\{x_1(t+1)=j_1, x_2(t+1)=j_2, n_1^a(t)=\psi_1,$$

$$n_2^a(t)=\psi_2 \,|\, x_1(t) = i_1, x_2(t)=i_2, u(t)=v\}$$

$$= S^1_{i_1 j_1}(t,v,\psi_1)\ S^2_{i_2 j_2}(t,v,\psi_2) \qquad (3.3.53)$$

$$\text{for all } i,j\varepsilon X; v\varepsilon U; \psi_1,\psi_2\varepsilon Y$$

or in matrix notation:

$$S(t,v,\psi) = S^1(t,v,\psi_1) \otimes S^2(t,v,\psi_2) \qquad (3.3.54)$$

where $\otimes$ denotes tensor product. The output observation probabilities for the combined system with $\psi=(\psi_1,\psi_2)$ are given by:

$$\Pr\{y(t)=\psi | x_1(t)=i_1, x_2(t)=i_2, u(t)=v\}$$

$$= \sum_{j_1,j_2\varepsilon I} S_{i_1 i_2; j_1 j_2}(t,v,\psi) = \sum_{j_1\varepsilon I} S^1_{i_1 j_1}(t,v,\psi_1) \sum_{j_2\varepsilon I} S^2_{i_2 j_2}(t,v,\psi_2)$$

$$= D^1_{i_1 i_1}(t,\psi_1)\ D^2_{i_2 i_2}(t,\psi_2)\ . \qquad (3.3.55)$$

where the first equality follows from (3.3.10) and the last equality follows from (3.3.37). Again in matrix notation, (3.3.55) becomes

$$D(t,v(t)) = D^1(t,n_a^1(t)) \otimes D^2(t,n_a^2(t)) \qquad (3.3.56)$$

Similarly, the M-matrix introduced in (3.3.20) for the combined system has the form:

$$M(t,v,y) = M^1(t,v,n_a^1(t)) \otimes M^2(t,v,n_a^2(t)) \qquad (3.3.57)$$

Assuming that the initial probability vectors Π_0^1, Π_0^2 for the two queue sizes are independent, implies that the initial probability vector for the combined state can be written as

$$\Pi_0 = \Pi_0^1 \otimes \Pi_0^2 \ . \qquad (3.3.58)$$

Consequently from (3.3.26), (3.3.56) it follows that the initial condition of the filtering-prediction recursive for the combined system satisfies:

$$\Pi_{0|0} = \frac{\Pi_0 \ D(0,y(0))}{\Pi_0 \ D(0,y(0))e} = \frac{\Pi_0^1 \ D^1(0,n_a^1(0)) \otimes \Pi_0^2 \ D^2(0,n_2^a(0))}{(\Pi_0^1 \ D^1(0,n_a^1(0))e)(\Pi_0^2 \ D^2(0,n_2^a(0)e)} \qquad (3.3.59)$$

$$= \Pi_{0|0}^1 \otimes \Pi_{0|0}^2 \ .$$

Moreover by (3.3.54) - (3.3.57), it follows inductively that (3.3.22), (3.3.26) have the form:

$$\Pi_{t|t} = \Pi_{t|t}^1 \otimes \Pi_{t|t}^2$$

$$\Pi_{t+1|t} = \Pi_{t+1|t}^1 \otimes \Pi_{t+1|t}^2 \qquad \text{for } t=0,1,2,\ldots,T \quad (3.3.60)$$

where $\{\Pi_{t|t}^i, \Pi_{t+1|t}^i; \ i=1,2\}$ are computed independently using (3.3.22), (3.3.26). Clearly similar tensor product expressions are valid for the unnormalized filtered and one-step predicted probability vectors of the combined two queue system.

To emphasize the significance of this "decoupling" nature of the filtering-prediction recursions for the stochastic control problem, we restate Theorem 3.3.3 and Corollary 3.3.4 for the combined queueing system.

Theorem 3.3.5. For the combined queue system described by (3.3.1), (3.3.4), (3.3.52) - (3.3.57):

(a) if nonanticipative control strategies as defined in (3.3.5) - (3.3.7) are used, then $\{\Pi^i_{t|t}, n^a_i(t); i=1,2\}$ (or $\{\rho^i_{t|t}, n^a_i(t); i=1,2\}$) is an information state at time t

(b) if strictly nonanticipative control strategies as defined in (3.3.6), (3.3.8) - (3.3.9) are used, then $\{\Pi^i_{t|t-1}; i=1,2\}$ (or $\{\rho^i_{t|t-1}; i=1,2\}$) is an information state at time t.

3.4 Finite Horizon Stochastic Optimal Control General Results

In this section, the finite horizon average aggregate delay (3.2.4), (3.2.5) problem for the queueing system (3.2.6) - (3.2.13) is considered. This priority assignment problem is formulated as a controlled, partially observed Markov process [34] - [38]. First, we review briefly the general dynamic programming results for a controlled, partially observed Markov process. By the introduction of the information state (3.3.30), this partially observed problem is transformed into a completely observed Markov decision problem. Consequently, the theory of Markov decision processes is applicable. Both anticipative and strictly non-anticipative control policies [31] are discussed. Second, we apply the particular results for the combined two competing queue problem. We obtain explicit solutions for the finite time expected aggregate delay problem for bounded and unbounded queues. These results are an extension of those discussed in Section 2.3. The implications of these results for practical applications are discussed.

Let the state and observation spaces be an n-dimensional and p-dimensional Euclidean vector spaces, denoted respectively as X and Y, with state dynamics and observations satisfying:

$$\begin{aligned} &(x(t+1) = \varphi_t(x(t),u(t),w(t+1)); \; x(0) = x_0 \\ &y(t) = \theta_t(x(t),v(t)) \quad \text{for } t=0,1,2,\ldots,T \end{aligned} \qquad (3.4.1)$$

where T is the finite time horizon, $u(t) \varepsilon U$ are the control values and $w(t) \varepsilon D$, $v(t) \varepsilon V$ are independent random variables with known distributions. The functions $\varphi_t(\cdot,\cdot,\cdot)$ and $\varphi_t(\cdot,\cdot)$ are assumed to be known. The random disturbances $\{w(t)\}$ are characterized by a probability measure $p_w(\cdot|x(t),u(t))$ defined on a collection of events in D. This probability measure may depend explicitly on x(t) and u(t), but not on values of prior state disturbances. The random disturbances $\{v(t)\}$ are characterized by a probability measure $p_v(\cdot|x(t))$ defined on a collection of events in V. This probability measure may depend explicitly on x(t), but not on prior observation disturbances v(0), v(1),...,v(t-1) or any of the state disturbances w(0), w(1),...,w(t). An underlying probability triple (Ω,F,P) which carries x_0 and the $\{w(t)\}$

and $\{v(t)\}$ processes is assumed to be given. Furthermore, we shall assume, as is standard [20], that the disturbance spaces D and V are countable sets. The control space U is a convex, compact subset of R^m. The state space is a countable subset of R^n; for bounded queues X is finite while for unbounded queues, X is infinite.

For a control policy, $\gamma\varepsilon\Gamma$ the finite horizon performance criterion is denoted by

$$J_f^\gamma(x_0) = E[\sum_{t=0}^{T-1} c(t,x^\gamma(t), u^\gamma(t)) + c(T,x^\gamma(T))] \quad (3.4.2)$$

where $c(t,x,u)$ and $c(T,x)$ denote respectively the instantaneous and terminal costs. The expectation above is, of course, taken with respect to the given probability distributions, $p_w(\cdot|x,u)$ which depends on x, u and $p_v(\cdot|x)$ which depends on x. The supercript γ in x, u indicates the state and control trajectories induced by the policy γ. The set of admissible nonanticipative control policies, Γ is defined in (3.3.5) - (3.3.7) while the set of admissible, strictly nonanticipative policies, Γ_0 is defined in (3.3.6), (3.3.8) - (3.3.9). The problem is to find $\gamma^*\varepsilon\Gamma$ (or $\gamma^*\varepsilon\Gamma_0$) such that

$$J_f^{\gamma^*}(x_0) = \inf\{J_f^\gamma(x_0): \gamma\varepsilon\Gamma\} \quad (3.4.3)$$

The corresponding policy, γ^* is called the <u>optimal nonanticipative</u> (or <u>strictly nonanticipative</u>) policy. It is well known [34], [36] that due to partial observations of the state, the optimal policy will not be Markovian [20], [31]. It has been shown in [24], [34], [36], [38], for various partially observed stochastic systems, that instead the

the optimal policy is a function only of the information state. Typically, the information state in the notation of Section 3.3 is the filtered probabilities $\Pi_{t|t}$ of (3.3.13). For the intended application, a slight modification is needed, to reflect the fact that the information state is not the usually prescribed one (see Theorem 3.3.5).

Before proceeding, we introduce the concept of a separated nonanticipative policies; in particular following [41]:

Definition 3.4.1. A policy $\gamma \varepsilon \Gamma$, $\gamma = \{g_0, g_1, \ldots, g_t \ldots\}$ is called a separated, nonanticipative policy if g_t depends on the available information, τ^t at time t, only through the information state, $\phi(t)$ as defined by (3.3.30); specifically

$$u(t) = g_t(\phi(t)) \qquad \text{for } t=0,1,2,\ldots,T \qquad (3.4.4)$$

Similarly, $\gamma \varepsilon \Gamma_0$ is called a separated, strictly nonanticipative policy if (3.4.4) holds.

Let $\Gamma_s \subset \Gamma$ denote the subset of separated, nonanticipative policies and $\Gamma_{0,s} \subset \Gamma_0$ denoted the subset of separated, strictly nonanticipative policies.

To solve the optimization problem (3.4.3), we resort to the well-known procedure of dynamic programming.

Let π be the set of probability vectors

$$\{\Pi(i); i \varepsilon X : \Pi(i) \geq 0, \sum_{i \varepsilon X} \Pi(i) = 1\} \qquad (3.4.5)$$

and let $\tau^{\gamma,t}$ denote the information vector, sample path generated while using policy γ. Let J_k^γ denote the expected value to go from t=k to T, given the information vector $\tau^{\gamma,k}$ and the control law γ is followed; specifically

$$J_k^\gamma = E[\sum_{t=k}^{T} c(t,x^\gamma(t),\ u^\gamma(t)) + c(T,x^\gamma(T))|\tau^{\gamma,k}]$$

$$\text{for } k = T-1,\ T-2,\ldots,0 \tag{3.4.6}$$

with terminal condition

$$J_T^\gamma = E[c(T,x^\gamma(T))|\tau^{\gamma,T}] \tag{3.4.7}$$

The problem then is to select a control law for which J_0^γ is a minimum. Since for any control law, J_k^γ satisfies (3.4.6), (3.4.7) it is natural to ask whether one can compute a control law which is optimal. We have the following sufficient condition for optimality in the case of nonanticipative policies:

Theorem 3.4.1. For $0 \leq k \leq T$, define the functions $V_k(\cdot,\cdot)$ on $\pi x Y$ such that

$$\text{(a)}\quad V_T(\Pi,y) = \sum_{i\varepsilon X} c(T,i)\ \Pi(i)$$

$$\text{(b)}\quad V_k(\Pi,y) = \inf_{u\varepsilon U} \{ \sum_{i\varepsilon I} c(k,i,u)\ \Pi(i) + \tag{3.4.8}$$

$$+ \sum_{\psi\varepsilon Y} V_{k+1}(\frac{\Pi\ M(k,u,y)\ D(k+1),\psi)}{\Pi\ M(k,u,y)\ D(k+1,\psi)e},\psi)\Pi\ M(k,u,y)\ D(k+1,\psi)e\}$$

Then for $\gamma\varepsilon\Gamma$

$$V_k(\Pi_{k|k}(z^{\gamma,k}),y(k)) \leq J_k^\gamma \qquad \text{for } k=0,1,2,\ldots T \tag{3.4.9}$$

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

$$V_k(\Pi_{k|k}(z^{\gamma^*,k}),y(k)) = J_k^{\gamma^*} \qquad \text{for } k=0,1,2,\ldots,T \tag{3.4.10}$$

Remark 3.4.1: Recall from Chapter 2 Theorem 2.3.1, the value functions were defined on the state space X, while here they are defined on the set of probability vectors, π. This is a consequence of transforming the partially observed problem to a completely observed problem. Also notice that the state dynamics in (2.3.6) and (3.4.8) differ. In the latter, the state dynamics of (3.4.1) are mapped by the filtering-prediction formulas (3.3.22), (3.3.26) to the operators shown. Specifically for the information vector, $\tau^{\gamma,k} = z^{\gamma,k}$ defined in (3.3.5), where $z^{\gamma,k}$ denotes the sample path generated information, the value of the filtered probability vector is denoted by $\Pi_{k|k}(z^{\gamma,k})$.

Proof: The basic steps are standard and are given here for the sake of completeness. By (3.3.23), (3.4.7)

$$J_T^{\gamma} = E\{c(T,x^{\gamma}(T))|x^{\gamma,T}\} = \sum_{i\varepsilon X} c(T,i)\,\Pi_{T|T}(z^{\gamma,T})$$

so that (3.4.9) holds with equality for k = T. Suppose (3.4.9) holds for k + 1, i.e.

$$V_{k+1}(\Pi_{k+1|k+1}(z^{\gamma,k+1}),y(k+1))\le J_{k+1}^{\gamma} \quad . \tag{3.4.11}$$

Then

$$J_k^{\gamma} = E\{c(k,x^{\gamma}(k),u^{\gamma}(k)) + E\{\sum_{t=k+1}^{T-1} c(t,x^{\gamma}(t),u^{\gamma}(t))+c(T,x^{\gamma}(T))|z^{\gamma,k+1}\}|z^{\gamma,k}\}$$

(by induction hypothesis (3.4.11)

$$\ge E\{c(k,x^{\gamma}(k),u^{\gamma}(k)) + V_{k+1}(\Pi_{k+1|k+1}(z^{\gamma,k+1}),y^{\gamma}(k+1))|z^{\gamma,k}\}$$

$$= E\{E\{c(k,x^{\gamma}(k),u^{\gamma}(k))+V_{k+1}(\Pi_{k+1|k+1}(z^{\gamma,k+1}),y^{\gamma}(k+1))|z^{\gamma,k},u^{\gamma}(k)\}|z^{\gamma,k}\}$$

(by (3.3.31))

$$= E\{ \sum_{i\in X} c(k,i,u^{\gamma}(k))\ \Pi_{k|k}(i,z^{\gamma,k})$$

$$+ V_{k+1}\Big(\frac{\Pi_{k|k}(z^{\gamma,k})M(k,u^{\gamma}(k),y^{\gamma}(k))D(k+1,y^{\gamma}(k+1))}{\Pi_{k|k}(z^{\gamma,k})M(k,u^{\gamma}(k),y^{\gamma}(k))D(k+1,y^{\gamma}(k+1))e}\ ,\ y^{\gamma}(k+1)\Big)|z^{\gamma,k}\}$$

(since $\gamma\in\Gamma$)

$$= \sum_{i\in X} c(k,i,u^{\gamma}(k))\ \Pi_{k|k}(i,z^{\gamma,k})+\sum_{y^{\gamma}(k+1)\in Y} V_{k+1}(\ldots)p^{\gamma}(y^{\gamma}(k+1)|z^{\gamma,k})$$

$$= \sum_{i\in X} c(k,i,u^{\gamma}(k))\ \Pi_{k|k}(i,z^{\gamma,k})$$

$$+ \sum_{\psi\in Y} V_{k+1}\Big(\frac{\Pi_{k|k}(z^{\gamma,k})M(k,u^{\gamma}(k),y^{\gamma}(k))D(k+1,\psi)}{\Pi_{k|k}(z^{\gamma,k})M(k,u^{\gamma}(k),y^{\gamma}(k))D(k+1,\psi)e}\ ,\psi\Big)\Pi_{k|k}(z^{\gamma,k})$$

$$\cdot M(k,y^{\gamma}(k),y^{\gamma}(k))D(k+1,\psi)e$$

(and by (b))

$$\geq V_k(\Pi_{k|k}(z^{\gamma,k}),y^{\gamma}(k)) \quad , \qquad (3.4.12)$$

and this completes the proof of (3.4.9). The last equality in (3.4.12) follows from

$$p^{\gamma}(\psi|z^{\gamma,k}) = \sum_{i\in X} p^{\gamma}(\psi|i,z^{\gamma,k})\cdot p^{\gamma}(i|z^{\gamma,k})$$

for $y^{\gamma}(k+1) = \psi,\ x^{\gamma}(k+1) = i$

(by (3.3.13) and Lemma 3.3.1)

$$= \sum_{i\in X} \lambda_i(\psi,k+1)\ \Pi_{k+1|k}(i,x^{\gamma,k})$$

(by (3.3.22), (3.3.27)

$$= \Pi_{k|k}(z^{\gamma,k})\ M(k,u^{\gamma}(k),y^{\gamma}(k))\ D(k+1,\psi)e \quad . \qquad (3.4.13)$$

To prove (3.4.10) observe that it holds for k = T by (3.4.6). Next assume (inductive hypothesis) that (3.4.10) holds for k + 1. In (3.4.12), the first inequality becomes now equality because of the inductive hypothesis. The last inequality in (3.4.12) becomes equality because now (b) holds with equality when $\gamma=\gamma^*$. Thus (3.4.10) holds for k. Finally for k=0 in (3.4.10)

$$V_0(\Pi_{0|0},y(0)) = J_0^{\gamma^*}$$

and

$$J(\gamma^*) = E\{J_0^{\gamma^*}\} = E\{V_0(\Pi_{0|0},y(0))\}. \qquad (3.4.14)$$

However for any $\gamma\varepsilon\Gamma$, from (3.4.9) with k=0

$$V_0(\Pi_{0|0},y(0))\leq J_0^{\gamma}$$

and

$$J(\gamma) = E\{J_0^{\gamma}\} \geq E\{V_0(\Pi_{0|0},y(0))\} \qquad (3.4.15)$$

and the proof of the theorem is complete. QED.

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

$$C_T(i) = c(T,i)$$
$$[C_k(u)](i) = c(k,i,u) \quad , \text{ for all } i\varepsilon X \qquad (3.4.16)$$

Then we can rewrite the dynamic programming recursion (a) (b) of Theorem 3.4.1 as:

$$V_T(\Pi,y) = \Pi\ C_T$$

$$V_k(\Pi,y) = \inf_{u\varepsilon U}[\ C_k(u) + \sum_{\psi\varepsilon Y} V_{k+1}(\frac{\Pi M(k,u,y)D(k+1,\psi)}{\Pi M(k,u,y)D(k+1,\psi)e},\psi)\cdot M(k,u,y)D(k+1,\psi)e]$$

$$k = 0,1,\ldots,T-1. \qquad (3.4.17)$$

In the case of strictly nonanticipative policies, we have the following:

Theorem 3.4.2: For $0\leq k\leq T$, define the functions $V_k(\cdot)$ on π such that

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

(b) $V_k(\Pi)=\inf_{u\varepsilon U}\{\Pi\ C_k(u)+\sum_{\psi\varepsilon Y} V_{k+1}(\frac{\Pi S(k,u,\psi)}{\Pi S(k,u,\psi)e})\cdot \Pi\ D(k,\psi)e\}$

$$(3.4.18)$$

Then for $\gamma\varepsilon\Gamma_0$

$$V_k(\Pi_{k|k-1}(\zeta^{\gamma,k-1}))\leq J_k^{\gamma} \qquad \text{for } k=0,1,2,\ldots,T. \qquad (3.4.19)$$

Furthermore let $\gamma^*\varepsilon\Gamma_{s,0}$, be a separated policy such that $g_k^*(\ )$ achieves the infimum in (b). Then γ^* is optimal in Γ_0 and with probability one

$$V_k(\Pi_{k|k-1}(\zeta^{\gamma^*,k-1})) = J_k^{\gamma^*} \qquad , \text{ for } k=0,1,\ldots,T \quad (3.4.20)$$

Proof: Now $\tau^{\gamma,k} = \zeta^{\gamma,k-1}$ defined in (3.3.9). The proof is almost identical to that of Theorem 3.4.1 and we only give the analog of (3.4.12) here. From (3.3.23), (3.4.7)

$$J_k^{\gamma}=E\{c(k,x^{\gamma}(k),u^{\gamma}(k))+E\{\sum_{t=k+1}^{T-1} c(t,x^{\gamma}(t),u^{\gamma}(t))+c(T,x^{\gamma}(T))|\zeta^{\gamma,k}\}|\zeta^{\gamma,k-1}\}$$

(by induction hypothesis)

$$= E\{c(k,x^{\gamma}(k),u^{\gamma}(k))+V_{k+1}(\Pi_{k+1|k}(\zeta^{\gamma,k}))|\zeta^{\gamma,k-1}\}$$

(by (3.3.31) and (3.3.38)

$$= E\{c(k,x^{\gamma}(k),u^{\gamma}(k))+V_{k+1}(\frac{\Pi_{k|k-1}(\zeta^{\gamma,k-1})S(k,u^{\gamma}(k),y^{\gamma}(k))}{\Pi_{k|k-1}(\zeta^{\gamma,k-1})S(k,u^{\gamma}(k),y\ (k))e})|\zeta^{\gamma,k-1}\}$$

(since $\gamma\varepsilon\Gamma_0$)

$$= \Pi_{k|k-1}(\zeta^{\gamma,k-1})C_k(u^{\gamma}(k)) +$$

$$+ \sum_{\psi\varepsilon} V_{k+1}(\frac{\Pi_{k|k-1}(\zeta^{\gamma,k-1})\ S(k,u^{\gamma}(k),\psi)}{\Pi_{k|k-1}(\zeta^{\gamma,k-1})\ S(k,u^{\gamma}(k),\psi)e})\ \Pi_{k|k-1}(\zeta^{\gamma,k-1})D(k,\psi)e$$

(by (b))

$$\geq V_k(\Pi_{k|k-1}(\zeta^{\gamma,k-1})) \qquad \text{QED.} \qquad (3.4.21)$$

Theorems 3.4.1 and 3.4.2 characterize the 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-prediction formulas (3.3.22), (3.3.26).

Application to the Two Competing QueuesProblem

The finite horizon partial observation formulation is now applied to the combined two competing queue problem of Section 3.2. The instantaneous cost, observations and state dynamics are given, respectively, by (3.2.4), (3.3.52) and (3.3.22), (3.3.26), (3.3.60). By transforming the partial observation problem into a complete observation problem, the state space is defined over the set of probability vectors, π defined in (3.4.5). Specifically, we let Π_0 denote the initial probability vector for the combined state (3.3.58) with the components ordered according to the sequence

00,01,02,...,10,11,12,...,20,21,22,... , In (3.45), the state set, X for the unbounded queueing system is the Cartesian product, $X = ZxZ$ where Z is the set of positive integers. For the bounded queueing system, the state set becomes $X = Z_1 x Z_2$ where

$$Z_i = \{0,1,2, \ldots , N_i\} \qquad \text{for } i=1,2. \tag{3.4.22}$$

For a policy $\gamma \varepsilon \Gamma$ (or $\gamma \varepsilon \Gamma_0$), the cost is the average aggregate delay

$$J_f^\gamma(\Pi_0) = E[\sum_{t=0}^{T} (c_1 x_1^\gamma(t) + c_2 x_2^\gamma(t) | \Pi_0] \tag{3.4.34}$$

From (3.4.2), (3.4.16), it follows that

$$C_k(i_1,i_2) = c_1 i_1 + c_2 i_2 \qquad \text{for all } (i_1,i_2) \varepsilon X \tag{3.4.25}$$

Let the vectors

$$\begin{aligned} e_i &= (1,1,1, \ldots , 1, \ldots)^T \varepsilon Z_i \\ v_i &= (0,1,2, \ldots , n, \ldots)^T \varepsilon Z_i \text{ for } i=1,2 \end{aligned} \tag{3.4.26}$$

where $\{Z_i; i=1,2\}$ is the set of positive integers for the unbounded queue case and is the set defined by (3.4.22) for the bounded queue case. Consequently, (3.4.25) becomes

$$C_k = c_1(\nu_1 \otimes e_2) + c_2(e_1 \otimes \nu_2)$$

where $\otimes$ denotes the tensor product with C_k ordered according to the sequence given above. Since C_k does <u>not</u> depend on k, the subscript to dropped from our notation

$$C_k = C \qquad \text{for } k=0,1,2,\ldots,T$$

First, we consider strictly nonanticipative control policies, Π_0 as defined in (3.3.6), (3.3.8) - (3.3.9). In this case the optimal policy and value function are determined by Theorem 3.4.2. Due to the "decoupling" of the filtering-prediction recursions (3.3.60), the dynamic programming recursion of Theorem 3.4.2 reduces to the following:

$$\begin{aligned} V_T(\Pi^1,\Pi^2) &= c_1\, \Pi^1\, \nu_1 + c_2\, \Pi^2\, \nu_2 \\ V_k(\Pi^1,\Pi^2) &= \inf_{u \varepsilon U}\{c_1\, \Pi^1\, \nu_1 + c_2\, \Pi^2\, \nu_2 + \end{aligned} \qquad (3.4.27)$$

$$+ \sum_{\psi_1,\psi_2 \varepsilon \{0,1\}} V_{k+1}\left(\frac{\Pi^1 S^1(k,u,\psi_1)}{\Pi^1 S^1(k,u,\psi_1) e_1}, \frac{\Pi^2 S^2(k,u,\psi_2)}{\Pi^2 S^2(k,u,\psi_2) e_1}\right) (\Pi^1 D^1(k,\psi_1) e_1) \cdot$$

$$\cdot\, (\Pi^2 D^2(k,\psi_2) e_2)\}$$

In (3.4.27) the probability vectors Π^1,Π^2 are defined over X, and $\{S^i, D^i;\ i=1,2\}$ are described for each queue by (3.3.42) - (3.3.49). Recall from (3.3.10) that the matrix of transition probabilities is given for each queue by

$$P^i(t,v) = \sum_{\psi=0}^{1} S^i(t,v,\psi) \qquad \text{for } i=1,2. \qquad (3.4.28)$$

Our aim is to show that (3.4.27) can be solved a priori and that all functions $\{V_k;\ k=0,1,\ldots, T\}$ are piecewise linear in Π^1,Π^2. Backwards induction on (3.4.27) is the most elementary method to establish these results. For $k = T$, (3.4.27) implies

$$V_T(\Pi^1,\Pi^2) = \Pi^1\, d_T^1 + \Pi^2\, d_T^2 \qquad \text{for all } (\Pi^1,\Pi^2)\, \varepsilon\, \pi, \quad (3.4.29)$$

where

$$d_T^i = c_i\, \nu_i \qquad \text{for } i = 1,2 \qquad (3.4.30)$$

are column vectors of dimension equal to the cardinality of $\{Z_i; i=1,2\}$ in (3.4.26). Next at k = T-1, (3.4.27) implies the following:

$$V_{T-1}(\Pi^1,\Pi^2) = \min_{u\varepsilon U}\{\Pi^1 \, d_T^1 + \Pi^2 \, d_T^2$$

$$+ \sum_{\psi_1,\psi_2=0}^{1} \Pi^1 \, S^1(T-1,u,\psi_1)d_T^1 \cdot [\Pi^2 \, D^2(T-1,\psi_2)e_2]$$

$$+ \sum_{\psi_1,\psi_2=0}^{1} \Pi^2 \, S^2(T-1,u,\psi_2)d_T^2 \cdot [\Pi^1 \, D^1(T-1,\psi_1)e_1]\}$$

$$= \min_{u\varepsilon U}\{\Pi^1[I_1 + P^1(T-1,u)]d_T^1 +$$

$$+ \Pi^2[I_2 + P^2(T-1,u)]d_T^2\} \qquad (3.4.31)$$

where the second equality follows from (3.4.28) and $\{I_i; i=1,2\}$ are identity operators of dimension equal to the cardinality of $\{Z_i; i=1,2\}$ in (3.4.26). Clearly, the optimal control as a function of Π^1,Π^2 is described as follows. The set π defined in (3.4.5) is separated in two disjoint subsets

$$A_1 = \{(\Pi^1,\Pi^2) \quad \text{such that} \qquad (3.4.32)$$

$$\Pi^1[P^1(T-1,0)-P^1(T-1,1)]d_T^1 \geq \Pi^2[P^2(T-1,1)-P^2(T-1,0))d_T^2]\}$$

A_0 = complement of A_1 in π.

We associate the index 1 and A_1, the index 0 with A_0, so that

$$u^*(T-1) = \begin{cases} 1 & \text{on } A_1 \\ 0 & \text{on } A_0 \end{cases} \qquad (3.4.33)$$

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

$$a_{T-1}(\Pi^1,\Pi^2) = \begin{cases} 1 & \text{if } (\Pi^1,\Pi^2)\varepsilon A_1 \\ 0 & \text{if } (\Pi^1,\Pi^2)\varepsilon A_0 \end{cases} \tag{3.4.34}$$

and

$$\begin{aligned} d^1_{T-1}(\Pi^1,\Pi^2) &= [I_1 + P^1(T-1,a_{T-1}(\Pi^1,\Pi^2))]\ d^1_T \\ d^2_{T-1}(\Pi^1,\Pi^2) &= [I_2 + P^2(T-1,a_{T-1}(\Pi^1,\Pi^2))]\ d^2_T\ . \end{aligned} \tag{3.4.35}$$

It is now clear that

$$V_{T-1}(\Pi^1,\Pi^2) = \Pi^1\ d^1_{T-1}(\Pi^1,\Pi^2) + \Pi^2\ d^2_{T-1}(\Pi^1,\Pi^2), \tag{3.4.36}$$

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

<u>Lemma 3.4.3</u>: Define the binary-valued functions $\{a_\ell;\ell=0,1,\ldots,T-1\}$ on π, as defined in (3.4.5) and the column vectors $\{d^i_\ell;i=1,2;\ell=0,1,\ldots,T\}$ by the backwards recursion

$$d^i_T = c_i\nu_i$$

$$d^i_{T-\ell} = d^i_T + P^i(T-\ell,a_{T-\ell})\ d^i_{T-\ell+1}(a_{T-\ell+1},\ldots,a_{T-1})$$

$$a_{T-\ell}(\Pi^1,\Pi^2) = \begin{cases} 1, \text{ if } \Pi^1[P^1(t-\ell,0)-P^1(T-\ell,1)]d^1_{T-\ell+1}(a_{T-\ell+1},\ldots,a_{T-1}) \\ \quad \geq \Pi^2\ P^2(T-\ell,1)-P^2(t\text{-}\ell,0]d^2_{T-\ell+1}(a_{T-\ell+1},\ldots,a_{T-1}) \\ 0, \text{ otherwise} \end{cases} \tag{3.4.37}$$

for i = 1,2; ℓ=1,2,...,T-1 .

Then for k=0,1,...,T, and $(\Pi^1,\Pi^2)\ \varepsilon\pi$

$$V_k(\Pi^1,\Pi^2) = \Pi^1 \; d_k^1(a_k,\ldots,a_{T-1}) + \Pi^2 \; d_k^2(a_k,\ldots,a_{T-1}) \tag{3.4.38}$$

In other words, $V_k(\cdot,\cdot)$ is piecewise linear for each k.

Proof: By mathematical induction. It follows from (3.4.31) - (3.4.36) that (3.4.38) holds for k=T-1. Let us assume that the result holds for $k=T-\ell$. Then from (3.4.27), after computations identical to (3.4.36), we have

$$V_{T-\ell-1}(\Pi^1,\Pi^2) = \min_{u\varepsilon U}\{\Pi^1 \; d_T^1 + \Pi^2 \; d_T^2 + $$

$$+ \Pi^1 P^1(T-\ell-1,u) d^1_{T-\ell} + \Pi^2 P^2(T- -1,u) d^2_{T-\ell}\}$$

It follows now by the definition of $a_{T-\ell-1}(\Pi^1,\Pi^2)$ and $d^1_{T-\ell-1}$, $d^2_{T-\ell-1}$ that (3.4.38) holds for $k = T-\ell-1$. QED.

Remark 3.4.2. Recall from Chapter 2 Lemma 2.3.8, a similar sequence of $\{a_\ell\}$ and $\{d^i_\ell;i=1,2\}$ were defined. In that context, particular elements of the column vectors:

$$[P^1(T-\ell,0) - P^1(T-\ell,1)] \; d^1_{T-\ell+1}(a_{T-\ell+1},\ldots,a_{T-1})$$

and (3.4.39)

$$[P^2(T-\ell,1) - P^2(T-\ell,0)] \; d^2_{T-\ell+1}(a_{T-\ell+1},\ldots,a_{T-1})$$

were compared in determining the corresponding value for $a_{T-\ell}(\cdot,\cdot)$. For the partial observation case, we have a generalizations; specifically the state probability vectors of each queue $\{\Pi^i;i=1,2\}$ are averaged over the corresponding elements of these vectors (3.4.37) to define $a_{T-\ell}(\cdot,\cdot)$. Clearly, if with probability one the combined queue states were at a particular $(i,j)^{th}$ component, them Lemma 3.4.3

reduces to Lemma 2.3.8. As in Remark 2.3.4, the recursion above proceeds diagrammatically as follows:

$$\begin{tikzcd} {\begin{bmatrix} d_T^1 \\ d_T^2 \\ a_{T-1} \end{bmatrix}} \arrow[r] & {\begin{bmatrix} d_{T-1}^1 \\ d_{T-1}^2 \\ a_{T-2} \end{bmatrix}} \arrow[r] & \cdots \arrow[r] & {\begin{bmatrix} d_{T-\ell}^1 \\ d_{T-\ell}^2 \\ a_{T-\ell-1} \end{bmatrix}} \arrow[r] & {\begin{bmatrix} d_{T-\ell-1}^1 \\ d_{T-\ell-1}^2 \\ a_{T-\ell-2} \end{bmatrix}} \arrow[r, phantom, "\cdots"] & {\begin{bmatrix} d_0^1 \\ d_0^2 \\ \end{bmatrix}} \end{tikzcd}$$

We also have established the corollary.

<u>Corollary 3.4.4</u>: The optimal control policy in feedback form, as a function of $(\Pi^1,\Pi^2)\varepsilon\pi$, is given by

$$g_f^*(k;\Pi^1,\Pi^2) = a_k(\Pi^1,\Pi^2) \qquad \text{for } k=0,1,\ldots,T-1.$$

Combining now the results of Lemma 3.4.3, Corollary 3.4.4 and (3.4.20) of Theorem 3.4.2, we have established the following result.

<u>Theorem 3.4.5</u>: The optimal server time allocation strategy and expected aggregate delay, for the finite horizon partially observed queueing system with strictly nonanticipative strategies are determined as follows. First the vectors $\{d_\ell^i;\ i=1,2;\ell=0,1,\ldots,T\}$ and binary-valued functions $\{a_\ell;\ell=0,1,\ldots,T-1\}$ are computed off-line and stored <u>a priori</u> from Lemma 3.4.3. Foe each queue, the one-step queue predicted probability vectors $\{\Pi_{k|k-1}^i;i=1,2\}$ are computed, using the recursions (3.3.50) (3.3.51) with initial conditions (3.3.58). The optimal strategy at time k is given by

$$g_f^*(k;\Pi_{k|k-1}) = a_k(\Pi_{k|k-1}^1,\Pi_{k|k-1}^2),\ k=0,1,2,\ldots,T-1. \tag{3.4.40}$$

The optimal average aggregate delay has the form:

$$V_0(\Pi_0^1,\Pi_0^2) = \Pi_0^1 \, d_0^1 + \Pi_0^2 \, d_0^2 \tag{3.4.41}$$

Note: The vectors d_0^1, d_0^2 in (3.4.41) are functions of Π_0^1,Π_0^2.

Remark 3.4.3: The implementation of the optimal strategy is similar to that of the complete observation case (see Remark 2.3.6). The decision space π is divided at the k^{th} step into at most to 2^{T-k} subsets which are characterized by binary numbers with T-k binary digits, i.e. $a_k a_{k+1} \cdots a_{T-1}$. The first binary digit of the number associated with the subset provides according to Corollary 3.4.4 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 (3.3.50) (3.3.51) which as have been shown elsewhere [33] are easily implemented on a microprocessor. Observe that the value functions $V_k(\cdot,\cdot)$ are concave in Π^1, convave in Π^2 for $k=0,1,\ldots,T$; a fact that follows easily from the defining backwards recursion (3.4.27) by an inductive argument.

We consider next nonanticipative control policies, Γ as defined in (3.3.5) - (3.3.7). In this case, the optimal policy and value function are determined by Theorem 3.4.1. Again due to the "decoupling" of the filtering-prediction recursions, the dynamic programming recursion of Theorem 3.4.1 reduces to the following:

$$V_T(\Pi^1,\Pi^2,n_a^1,n_a^2) = c_1 \Pi^1 \nu_1 + c_2 \Pi^2 \nu_2$$

$$V_k(\Pi^1,\Pi^2),n_a^1,n_a^2) = \inf_{u\varepsilon U}\{c_1 \Pi^1 \nu_1 + c_2 \Pi^2 \nu_2 + \qquad (3.4.42)$$

$$\sum_{\psi_1,\psi_2\varepsilon\{0,1\}} V_{k+1}(\frac{\Pi^1 M^1(k,u,n_a^1)D^1(k+1,\psi_1)}{\Pi^1 M^1(k,u,n_a^1)D^1(k+1,\psi_1)e_1}, \frac{\Pi^2 M^2(k,u,n_a^2)D^2(k+1\psi_2)}{\Pi^2 M^2(k,u,n_a^2)D^2(k+1,\psi_2)e_2},\psi_1,\psi_2)$$

$$\cdot [\Pi^1 M^1(k,u,n_a^1) D^1(k+1,\psi_1)e_1]\cdot[\Pi^2 M^2(k,u,n_a^2) D^2(k+1,\psi_2)e_2]\}$$

In (3.4.42), the probability vectors Π^1,Π^2 are defined over $\chi$and $\{M^i,D^i;i=1,2\}$ are described for each queue by (3.3.47) - (3/3/49). The reason for the reduction in (3.4.42) is the independence condition introduced earlier. If one rederives (3.4.12) under these assumption and the cost structure (3.4.25), the recursion reduces to (3.4.42). The same inductive step employed in (3.4.12) establishes (3.4.42), while for k=T the form given in (3.4.42) is apparent.

Again we can show that (3.4.42) can be solved a priori and that all functions $\{V_k;\ k=0,1,...,T\}$ are piecewise linear in Π^1,Π^2. Backwards induction on (3.4.42) is applied. For k=T (3.4.42) implies

$$V_T(\Pi^1,\Pi^2,n_a^1,n_a^2) = \Pi^1 \delta_T^1 + \Pi^2 \delta_T^2 \text{ , for all } (\Pi^1,\Pi^2)\varepsilon\pi;(n_a^1,n_a^2)\varepsilon y \qquad (3.4.43)$$

where

$$\delta_T^i = c_i \nu_i \quad \text{for } i=1,2 \qquad (3.4.44)$$

are column vectors of dimension equal to the cardinality of $\{Z_i;i=1,2\}$ in (3.4.26). Next at k=T-1, (3.4.42) implies

$$V_{T-1}(\Pi^1,\Pi^2,n_a^1,n_a^2) = \min_{u\varepsilon U}\{\Pi^1 \; \delta_T^1 + \Pi^2 \; \delta_T^2 +$$

$$+ \sum_{\psi_1,\psi_2=0}^{1} \Pi^1 \; M^1(T-1,u,n_a^1) \; D^1(T,\psi_1) \; \delta_T^1[\Pi^2 \; M^2(T-1,u,n_a^2) \; D^2(t,\psi_2)e_2]$$

$$+ \sum_{\psi_1,\psi_2=0}^{1} \Pi^2 \; M^2(T-1,u,n_d^a) \; D^2(T,\psi_2) \; \delta_T^2[\Pi^1 M^1(T-1,u,n_a^1) \; D^1(T,\psi_1)e_1]\} \tag{3.4.45}$$

Recall from (3.3.49) that

$$D^i(k+1,0) + D^i(k+1,1) = I_i \text{ for } i=1,2; \text{ for all } k$$

where $\{I_i;i=1,2\}$ are identity operators of dimension equal to cardinality of $\{Z_i;i=1,2\}$ of (3.4.26). Then any one of the sums in (3.4.45) reduces to

$$\sum_{\psi_1,\psi_2=0}^{1} (\ldots) = \Pi^1 \; M^1(T-1,u,n_a^1) \; D^1(T,0) \; \delta_T^1[\Pi^2 M^2(T-1,u,n_a^2)e_2]$$

$$+ \Pi^1 \; M^1(T-1,u,n_a^1) \; D^1(T,1) \; \delta_T^1[\Pi^2 M^2(T-1,u,n_a^2)e_2]$$

$$= \Pi^1 \; M^1(T-1,u,n_a^1) \; \delta_T^1 \tag{3.4.46}$$

in view of (3.3.47) - (3.3.49). Thus

$$V_{T-1}(\Pi^1,\Pi^2,n_a^1,n_a^2) = \min_{u\varepsilon U}\{\Pi^1[I_1+M^1(T-1,u,n_a^1)] \; \delta_T^1 +$$

$$+ \Pi^2[I_2+M^2(T-1,u,n_a^2)] \; \delta_T^2\}. \tag{3.4.47}$$

The set of information states $\pi x\{0,1\} \times \{0,1\}$ is separated in two disjoint subsets

$$A_1 = \{(\Pi^1,\Pi^2,n_a^1,n_a^2), \text{ such that}$$

$$\Pi^1[M^1(T-1,0,n_a^1) - M^1(T-1,1,n_a^1)]\delta_T^1 \geq$$

$$\Pi^2[M^2(T-1,1,n_a^2) - M^2(T-1,0,n_a^2)]\delta_T^2\} \tag{3.4.48}$$

A_0 = complement of A_1 in πx{0,1} x {0,1}.

Clearly the optimal control is

$$u^*(T-1) = \begin{cases} 1 & \text{on } A_1 \\ 0 & \text{on } A_0 \end{cases} \tag{3.4.49}$$

Let us define the binary-valued function

$$b_{T-1}(\Pi^1,\Pi^2,n_a^1,n_a^2) = \begin{cases} 1 & \text{on } A_1 \\ 0 & \text{on } A_0 \;. \end{cases} \tag{3.4.50}$$

i.e. the characteristic function of the set A_1. If we let

$$\delta^1_{T-1}(\Pi^1,\Pi^2,n_a^1,n_a^2) = [I_1 + M^1(T-1,b_{T-1}(\ldots),n_a^1)]\delta^1_T$$

(3.4.51)

$$\delta^2_{T-1}(\Pi^1,\Pi^2,n_a^1,n_a^2) = [I_2 + M^2(T-1,b_{T-1}(\ldots),n_a^2)]\delta^2_T$$

it is immediate that

$$V_{T-1}(\Pi^1,\Pi^2,n_a^1,n_a^2) = \Pi^1\;\delta^1_{T-1}(\Pi^1,\Pi^2,n_a^1,n_a^2) + \Pi^2\;\delta^2_{T-1}(\Pi^1,\Pi^2,n_a^1,n_a^2)$$

(3.4.52)

and therefore $V_{T-1}(\cdot,\cdot)$ is piecewise linear in Π^1,Π^2 also. The general induction step is identical and leads to the following:

Lemma 3.4.6: Define the binary- valued functions $\{b_\ell;\ell=0,1,\ldots,T-1\}$ on π x{0,1}x{0,1} and the column vectors $\{\delta^i_\ell;i=1,2;\ell=0,1,\ldots,T\}$ by the backwards recursion

$$\delta_T^i = c_i \; \nu_i$$

$$\delta_{T-\ell}^i = \delta_T^i + M^i(T-\ell, b_{T-\ell}, n_a^i) \; \delta_{T-\ell+1}^i$$

$$b_{T-\ell}(\Pi^1,\Pi^2,n_a^1,n_a^2) = \begin{cases} 1, \text{ if } \Pi^1[M^1(T-\ell,0,n_a^1)-M^1(T-\ell,1,n_a^1)]\delta_{T-\ell+1}^1 \geq \\ \qquad \Pi^2[M^2(T-\ell,1,n_a^2)-M^2(T-\ell,0,n_a^2)]\delta_{T-\ell+1}^2 \\ 0, \text{ otherwise} \end{cases}$$

for $i=1,2;\ell=1,2,\ldots,T-1$

Then for $k=0,1,\ldots,T$, $(\Pi^1,\Pi^2,n_a^1,n_a^2) \varepsilon \bar{\pi} x\{0,1\}x\{0,1\}$

$$V_k(\Pi^1,\Pi^2,n_a^1,n_a^2) = \Pi^1 \; \delta_k^1 + \Pi^2 \; \delta_k^2.$$

In other words, $V_k(\cdot,\cdot)$ is piecewise linear in Π^1,Π^2 for each k.

We also have established the corollary.

Corollary 3.4.7: The optimal control policy in feedback form, as a function of Π^1,Π^2,n_a^1,n_a^2 is given by

$$g_f^*(k;\Pi^1,\Pi^2,n_a^1,n_a^2) = b_k(\Pi^1,\Pi^2,n_a^1,n_a^2), \; k=0,1,\ldots,T-1.$$

Combining now the results of Lemma 3.4.6, Corollary 3.4.7 and Theorem 3.4.1, we have the following:

Theorem 3.4.8: The optimal server time allocation strategy and expected aggregate delay, for the finite horizon partially observed queueing system with nonanticipative strategies are determined as follows. First the vectors $\{\delta_\ell^i;i=1,2,\ell=0,1,\ldots,T\}$ and binary-valued functions $\{b_\ell;\ell=0,1,\ldots,T-1\}$ are computed and stored a priori from Lemma 3.4.6. For each queue, the filtered probability vectors $\{\Pi_{k|k}^i;\; i=1,2\}$ are computed, using the recursions (3.3.50) (3.3.51) with initial conditions (3.3.58). The optimal strategy at time k is given by:

$$g_f^*(k;\Pi_{k|k},y(k)) = b_k(\Pi^1_{k|k},\Pi^2_{k|k},n^1_a(k),n^2_a(k)),\ k=0,1,\ldots,T-1 \tag{3.4.53}$$

The optimal average aggregate delay is

$$V_0(\Pi^1_0,\Pi^2_0) = \Pi^1_0\ \delta^1_0 + \Pi^2_0\ \delta^2_0 \tag{3.4.54}$$

Note: The vectors δ^1_0, δ^2_0 in (3.4.54) are functions of Π^1_0,Π^2_0.

Remark 3.4.4: As was noted in Remark 3.4.3, the nature of the solution (3.4.53) (3.4.54) suggests a computer-oriented implementations. The decision space is $\pi x\{0,1\}\ x\{0,1\}$. We concentrate on the π part as an index set. Essentially, the computation of the optimal policy proceeds by subdividing the decision space π to regions, which define the optimal control at each time. When X is countable, π is a convex subset of the positive cone in $R^{N_1}xR^{N_2}$. At any rate the subsets of π where b_k takes the value 1 or 0 are on each side, respectively of a hyperplane, as is easily seem by the definition of b_k in Lemma 3.4.6. In effect at each time $k=0,1,\ldots,T-1$, π is divided at most to $2^{3(T-k)}$ subsets which are characterized each by two binary numbers, one with T-k binary digits and one with 2(T-k) binary digits. The first number has as digits the sequence of values $b_k,b_{k+1},\ldots,b_{T-1}$. The second number just describes a possible sample path of $n^1_a(t),n^2_a(t)$, $t=k,k+1,\ldots,T-1$ and therefore has T-k slots with two digit binary numbers each. To determine the control to be applied, one computes the filtered probability vectors $\{\Pi^i_{k|k};i=1,2\}$ and finds the subset characterized by

$\{\Pi^i_{k|k}, n^i_a(k); i=1,2\}$. The first binary digit of the first number associated with the subset provides according to Corollary 3.4.7 the optimal control in feedback form.

3.5 An Example

As an illustration of the foregoing theory, we consider the problem posed in Section 3.4, Figure 3.1 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, Γ_0 of Theorem 3.4.5 is compared with two suboptimal policies by means of Monte Carlo simulation.

The combined two queue system is modelled as described in (3.3.52) - (3.3.60) with the performance objective of (3.4.24). 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 (3.2.12)).

(ii) no customer departs from queue i when either queue i has zero customers or queue j ($j\neq i$) is being served (see (3.2.8)).

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 3.4.3. To weight the merits of the a priori calculation of these normals, 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

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

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

where π is defined in (3.4.5) and let

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

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

The probability-of-zero strategy $\{u(k)^{\Pi}; k=0,1,\ldots,T-1\}$ chooses to serve the queue which has the higher probability of being nonempty. The MMSE strategy $\{u(k)^{MMSE}; k=0,1,\ldots,T-1\}$ selects the queue having the higher estimated queue size. Recall from (3.3.13), (3.4.26)

$$\hat{x}_i(k|k-1) = E\{x_i(k)|\zeta^{k-1}\} = \Pi^i_{k|k-1} \nu_i; \; i=1,2$$

Both suboptimal policies are simpler computationally but disregard the future evolutions of the Markov chain and consider only 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 parameter 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 traffic.

Figures 3.2 and 3.3 show results obtained in the first case, intended to represent light to moderate traffic. Here $\lambda^1=\lambda^2=0.35$, $\mu=0.70$. In Figure 3.2, 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 Figure 3.2, 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 Figure 3.3 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

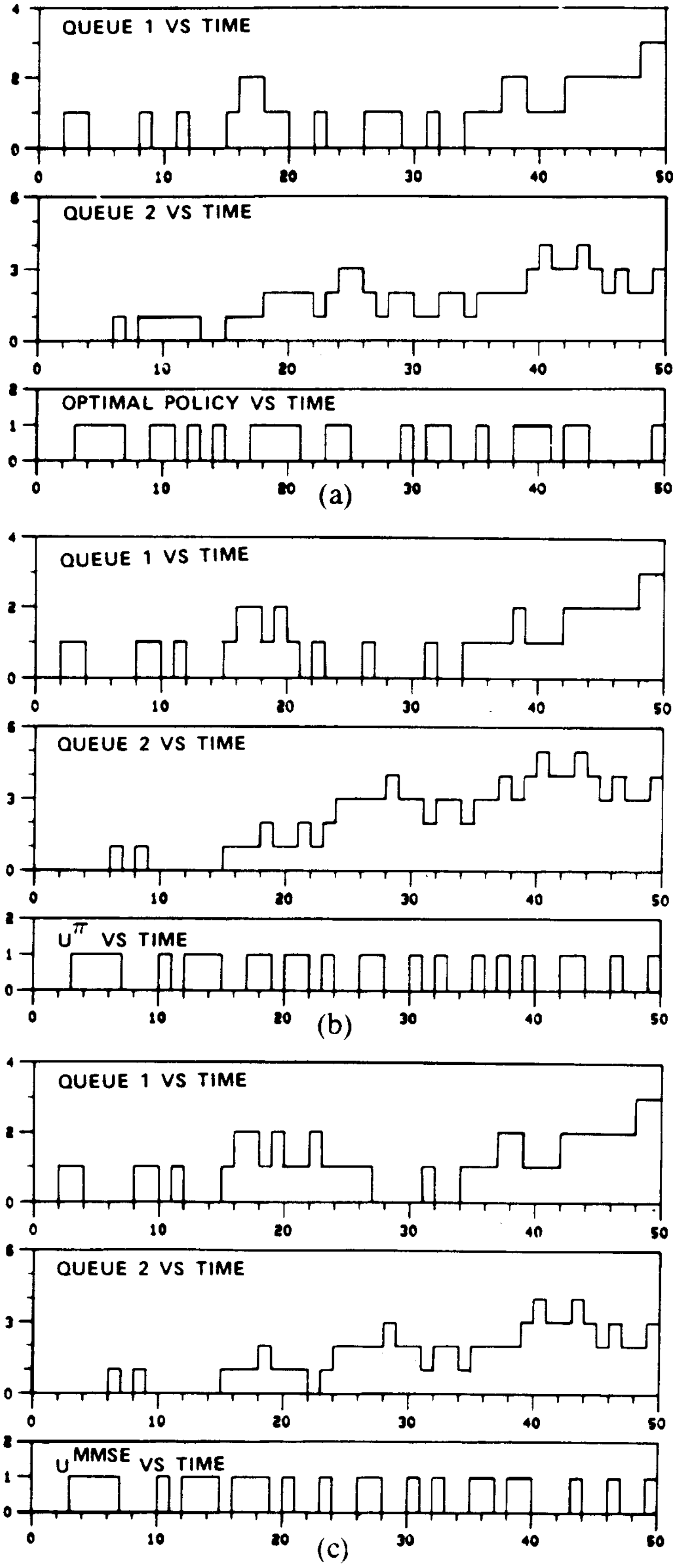


Fig. 3.2. Comparison of three policies when $\lambda_1=\lambda_2=0.35$, $\mu=0.7$ (a) optimal policy u^*, (b) policy u^π, (c) policy u^{MMSE}.

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. 3.3. Aggregate delays achieved in 50 samples, by each policy. $\lambda_1 = \lambda_2 = 0.35$, $\mu = 0.7$.

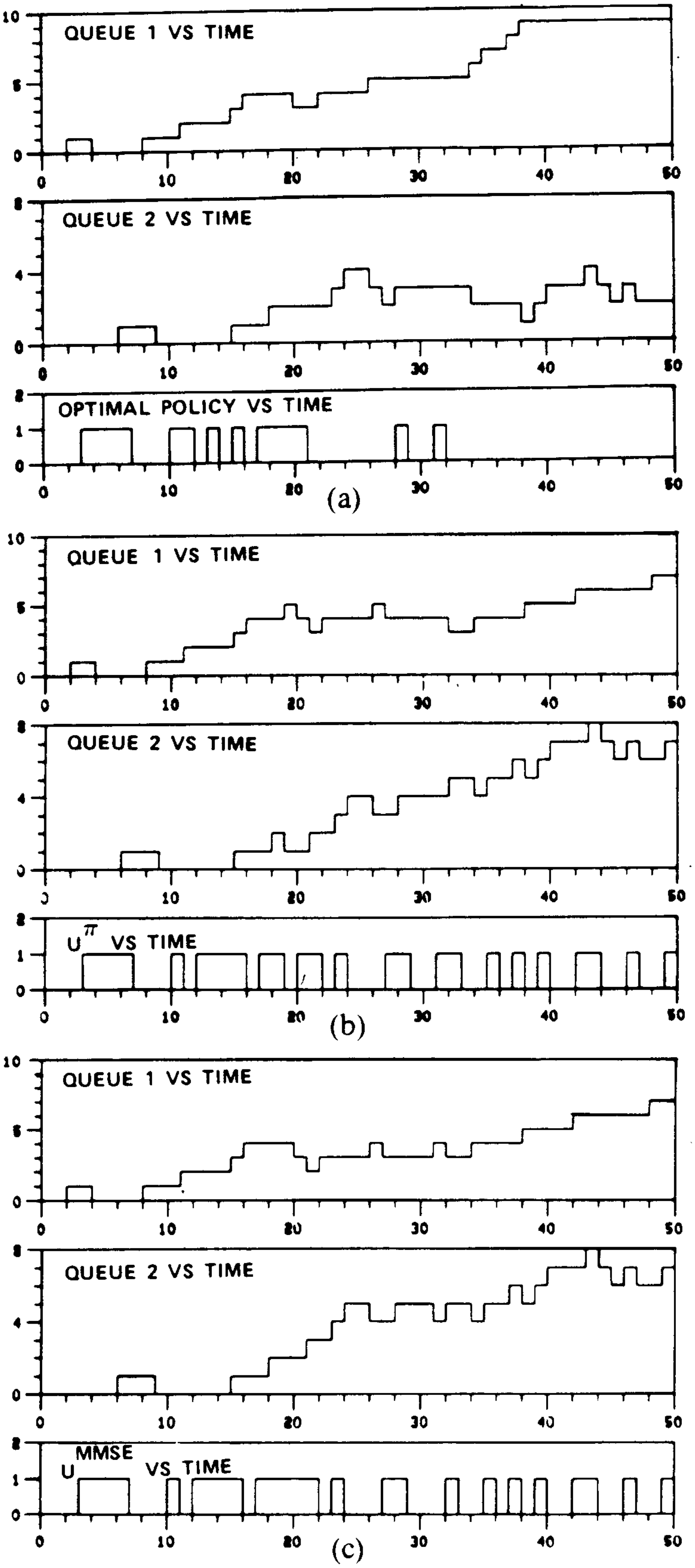


Fig. 3.4. Comparison of three policies when $\lambda_1=\lambda_2=0.35$, $\mu=0.35$ (a) optimal policy u^*, (b) policy u^π, (c) policy u^{MMSE}.

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. 3.5. Aggregate delays achieved in 50 samples, by each policy, $\lambda_1 = \lambda_2 = 0.35$, $\mu = 0.35$.

results. This problem will be studied elsewhere.

Similar results are presented in Figures 3.4 and 3.5 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. The analytical establishment, however is an open problem.

4. ADAPTIVE CONTROL OF TWO PARTIALLY OBSERVED COMPETING QUEUES

4.1 Introduction

The problems studied here have considered a finite-controlled Markov chain where the control objective is to minimize certain performance objectives. The aforementioned models have assumed a priori knowledge of the chain's transition probabilities. In Chapter 2, we dealt with the completely observed queueing system while in Chapter 3, a partially observed queueing problem was considered. Here, our discussion deals with the partially observed system whereby the a priori knowledge is assumed incomplete; specifically the parameters characterizing the chain's transition probabilities are assumed unknown. The problem then is to identify the unknown parameters while seeking the optimal control policy. In other words, the optimal controller of such a system needs to perform the dual, simultaneous function of realizing the desired performance objective and reducing system uncertainty. Such a problem is referred to as an adaptive control problem [50], [51].

Our objective is to derive an adaptive control strategy for the two competing queue problem presented earlier, when the arrival and departure rates $\{\lambda_i, \mu_i : i=1,2\}$ are unknown constants. The problem formulation is similar to the complete observation problem studied in Chapter 2. Two parallel queues are served by a single server with the control selected so that the infinite horizon aggregate delay is minimized. The problem is formulated in discrete time with arrival and departure processes modelled by Bernoulli streams. At each service completion time, a control is selected to decide which queue to service next. The controller observes both the arrival and departure processes. The control is to be selected as a function of the past histories of the observed processes and the control process. The instantaneous cost is linear in the waiting time of each queue. Because the transition probabilities depend on the unknown parameters, we have an infinite horizon, adaptive control problem.

The adaptive control problem is analyzed via two different methods. First, an approximate solution to the problem is posed whereby parameter estimates are substituted for the true ones in the optimal control policy. Recall from Chapter 2, the optimal infinite horizon strategy was the μc-rule. By this approach at each service completion time, a maximum likelihood (ML) estimate is generated for the unknown departure rates $\{\mu_i;i=1,2\}$ and substituted into the control policy. The convergence properties of the parameter estimate and the adaptive control policy are discussed. This method in the adaptive control literature is often called certainty equivalence or self-tuning [50], [51]. The second approach treats the adaptive control problem as a stochastic control problem with partial observations. In this case, the parameters $\{\lambda_i, \mu_i; i=1,2\}$ are treated as additional states, which are however unobservable. Since they are constant, they have trivial transition probabilities. The stochastic control methodology for partially observed Markov chains (e.g. Chapter 3) is applied to obtain the optimal control strategy. By construction, the resulting strategy is adaptive.

The problem of adaptively controlling a Markov chain, whose transition probabilities depend on an unknown parameter, has appeared recently in the literature [52] - [60]. Each adaptive controller is derived assuming the certainty equivalence principle [39, p. 339] holds. Mandl's pioneering work [52] considered an adaptive control scheme under the following assumption:

Identifiability Condition (IC) For each pair $\alpha, \alpha^1 \varepsilon A$ - a finite parameter set, if $\alpha \neq \alpha^1$ then there exists a $i \varepsilon Z$ (finite) for which

$$[p(i,1;u,\alpha),\ p(i,2;u,\alpha),..,p(i,n;u,\alpha)]$$
$$\neq [p(i,1;u,\alpha^1),\ p(i,2;u,\alpha^1),...,p(i,n;u,\alpha^1)]$$
$$\text{for every control } u \varepsilon U$$

where $p(i,j;\nu,\alpha) \equiv \Pr\{x(t+1)=j \mid x(t)=i, u(t+1)=v, \alpha\}$ (4.1.1)

If the chain satisfied the IC condition, then Mandl showed that the minimum constrast estimate (of which the ML estimate is one) of the unknown parameter converged to the true parameter almost surely (a.s.). Moreover for the infinite horizon, average cost per unit time performance criterion, the cost using the adaptive scheme converged a.s. to the optimal cost achieved, if the true parameter were known a priori. The adaptive scheme enforced the separation principle by first updating the parameter estimate and then selecting the control. The results were obtained for an arbitrary cost function with the restrictions that (i) the optimal control law be stationary and (ii) the Markov chain be irreducible.

Mandl's result guaranteed that the certainty space equivalence adaptive controller converged to the true optimal control policy. This fact was a consequence of the identifiability condition. However even in the simplest models [53], this IC condition need not hold. Moreover, Mandl's parameter estimate converged to the true value irrespective of the control policy employed. Consequently, it seems advisable to weaken Mandl's IC assumption.

Several papers have appeared in an effort to study the behavior of the parameter estimator and adaptive controller when Mandl's IC condition is relaxed. An important result by Borkar and Varaiya [53], [54] examined the adaptive control problem with the identifiability condition replaced by the assumption:

There exists an $\varepsilon>0$ such that for every $i,j\varepsilon Z$, either

$$p(i,j;u,\alpha)>\varepsilon \qquad \text{or} \qquad p(i,j;u,\alpha) = 0 \text{ for all } u,\alpha \tag{4.1.2}$$

For this case and for control laws of the form:

$$u(t) = \phi(\alpha(t), x(t-1)) \qquad (4.1.3)$$

they showed that the ML estimate, $\hat{\alpha}_t$ (generated at each stage) converges a.s. to a random variable α^* such that

$$p(i,j;\phi(\alpha^*,i),\alpha^*) = p(i,j;\phi(\alpha^*,i),\alpha^0) \qquad (4.1.4)$$

$$\text{for all } i,j\varepsilon X$$

where $\alpha^0(\varepsilon A)$ is the true parameter. In other words, the maximum likelihood estimate of the parameter converges to a value in the parameter set A such that the closed-loop transition probabilities are indistinguishable with those corresponding to the true system. It can happen however that the parameter estimator may not converge to the true value and hence (4.1.4) not be satisfied. Moreover by relaxing Mandl's IC condition, the average cost per unit time performance criterion using the adaptive scheme may exceed the corresponding optimal cost achieved if the true parameter were known a priori. In [53], the result (4.1.4) was obtained for finite state space χ and a finite parameter set A while in [54] the results were generalized for a countable state space and a compact separable metric space, A. Borkar and Varaiya's results were less general than Mandl's in one sense; namely they considered only maximum likelihood estimators.

Variations on the results of [52] - [54] have followed by several researchers [55] - [60]. In [55], Doshi and Shreve showed that by choosing a randomized estimate, $\hat{\alpha}_t$

from among all those α's which nearly maximized the log-likelihood function, the resulting estimate converged a.s. to the true parameter. Kumar et al [56] - [58] considered the adaptive control scheme using a modified maximum likelihood estimator; namely a likelihood function that is biased in favor of parameters which yielded lower optimal costs. In [56] for finite parameter, control and state spaces, they showed the corresponding adaptive controller obtained the performance precisely equal to that of the optimal performance attainable if the system were known a priori. The performance criterion was the average cost per unit time. Moreover, the overall performance of the adaptive system <u>did</u> <u>not</u> depend on whether the parameter estimate converged or diverged. This represents a significant step in that the adaptive controller primarily attempts to achieve overall system performance and only secondarily considers the corresponding parameter estimation. In [57], the restriction of finiteness of the parameter space is relaxed, but the state and control spaces are finite. The finiteness restrictions on the state and control spaces is removed in [58]. In [59], Sagalovsky considered the results of [53], [54] with the additional assumption that the unknown parameter enter the transition probabilities linearly; specifically

$$p(i,j;u_k,\alpha) = a_k(i,j,u_k)\alpha + b_k(i,j,u_k) \tag{4.1.5}$$

where $a_k(\cdot,\cdot,\cdot)$, $b_k(\cdot,\cdot,\cdot)$ are known functions. He showed that for $\alpha^o \varepsilon A$-a closed bounded interval, if the sequence $\{a_k\}$ approaches zero asymptotically, then the ML estimate $\hat{\alpha}_t$ does not converge. In other words for the ML estimate not to converge, the transistions should provide less and less information as k grows. Conversely, Sagalovsky showed that if the sequence $\{a_k\}$ satisfies a certain sufficiency condition, then the ML estimate converges a.s. to a parameter value α^* (where α^* may not necessarily equal the true value). One limitation of the ML estimator is its nonrecursive nature. Within the framework of this adaptive control problem, El-Fattah [60] considered a recursive stochastic gradient algorithm for the parameter estimation. He showed that for the class of randomized control laws, the adaptive control scheme and corresponding parameter estimate converged almost surely.

The adaptive control problem considered here differs in two respects with the aforementioned works. First, the competing queue's Markov chain does not satisfy the sufficent condition (4.1.2) for convergence of the adaptive control scheme given in [53] - [60]. Nevertheless exploiting the special structure of the Markov chain and its corresponding optimal policy, we are able to establish convergence for the unbounded system. Second, various information patterns are available to the system: both to the controller and the parameter estimator. In our case, two alternative ML estimators are developed and their convergence properties analyzed. Our analysis incorporates the results for the complete observation problem of Chapter 2.

A similar problem to ours has been studied by Hermandez-Lerma and Marcus [61]. They considered a continuous time, non-Markovian decision process with K-priority classes. For random sample times, they show that

(i) the sample mean estimator $\hat{\mu}_t$ converges a.s. to the true parameter, μ^o.

(ii) the adaptive control scheme using $\hat{\mu}_t$ converges a.s. to the optimal policy achieved if the true parameter were known a priori.

(iii) the expected long-run, average cost per unit time performance, using the adaptive scheme converged a.s. to the optimal cost, achieved if the true parameter were known a priori.

Our work parallels theirs in (i) and (ii) for the discrete-time system.

This chapter is organized as follows. In Section 4.2, we formulate the adaptive control problem and establish the relevant notation. The basic questions to be studied in this chapter are discussed. In Section 4.3, our main results on certainty-equivalence, adaptive control of the two competing queue system are described. The results of Borkar and Varaiya [53, 54] are modified for the queueing system's dynamics. The adaptive control problem treated as a stochastic control problem with partial observation is presented in Section 4.4. Although straightforward, this latter adaptive control scheme is computationally unfeasible for the general case. In Section 4.5, we present computations and evaluations of the adaptive strategies obtained, as the theory is applied to a simple problem.

4.2 Problem Formulation and Notation

We consider a queueing system similar to the one introduced in Section 2.2; we shall therefore be brief. Two queues are served by the same server in discrete time. The time is divided into equal length time slots (which are prespecified) with the convention that the t^{th} time slot is the half open interval $[t-1, t)$. We let $t = 0,1,2,...$ be the index of these time slots and the length of each slot is assumed to be unity. During each time slot, arrivals and service completions can occur. The situation is depicted in Figure 4.1 below.

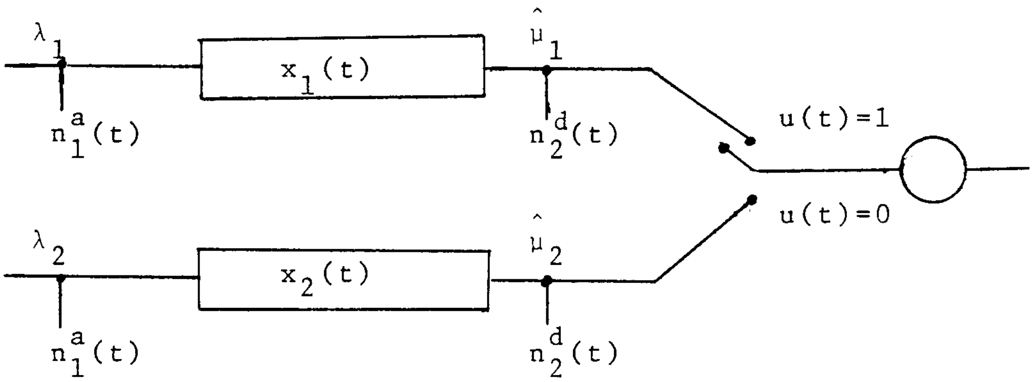


Figure 4.1. The adaptive control problem.

Customers arrive into queues 1 and 2 according to two independent Bernoulli streams with constant rates λ_1, λ_2 respectively. Each queue competes for the services of the single server. When the server serves queue i, i = 1,2, service completions follow a Bernoulli stream with constant rate $\{\mu_i;\ i = 1,2\}$. By implication at most one arrival and one service can occur during each time slot, when each queue operates alone.

Let $x_i(t)$ be the number of customers in queue $i(i = 1,2)$ at the end of the t^{th} time slot, the customer in service (if any) included. The control is used to allocate server time to queue 1 or to queue 2. Namely 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. Following the notational convention of Chapter 2, let $\{n_i^a(\cdot), n_i^d(\cdot);\ i = 1,2\}$ denote respectively the two arrival and the two departure processes with associated rates given by (see (2.2.2) and (2.2.3)):

$$\lambda_i = \Pr\{n_i^a(t) = 1\} \qquad ;\ i = 1,2,\ . \tag{4.2.1}$$

$$\mu_i(t,k,v) = \Pr\{n_i^d(t) = 1 \mid x_i(t-1) = k,\ u(t)=v\};\ i=1,2 \tag{4.2.2}$$

where in particular

$$\mu_1(t,k,v) = \begin{cases} \mu_1\ v\ , & \text{if } k\neq 0 \\ 0\ , & \text{if } k=0 \end{cases} \tag{4.2.3}$$

$$\mu_2(t,k,v) = \begin{cases} \mu_2(1-v)\ , & \text{if } k\neq 0 \\ 0\ , & \text{if } k=0 \end{cases} \tag{4.2.4}$$

We assume that both queues can grow without bound. This allows analytical treatment of the problem. When the queues are bounded, the methods used here lead to numerical treatment; analytical solutions have not been obtained. In the latter case if $\{N_i, i=1,2\}$ are the maximum queue sizes for each queue, we have additional constraints on the arrival rates

$$\lambda_i(t,k,v) = \begin{cases} \lambda_i, & \text{if } k\neq 0, \text{ all } t, v, \\ 0, & \text{if } k=N_i, \text{ all } t, v, \end{cases} \quad \text{for } i = 1,2 \tag{4.2.5}$$

Each queue is modelled as a chain with a countable state space and having transition probabilities as given in (2.2.7) - (2.2.10). Moreover the transition probability matrix for the chain describing both queues is given by:

$$P(v) = P^1(v) \otimes P^2(v) \quad , \text{ for all } v \tag{4.2.6}$$

where $P^1(\cdot)$ and $P^2(\cdot)$ are given in (2.2.9), (2.2.10) respectively and $\otimes$ indicates matrix tensor product, For any value of the control variable v(i.e. 0 or 1), P(v) will not be a block diagonal matrix and therefore any state will communicate with any other. In other words, P(v) is irreducible [19, p. 232] for each value of v. We also observe that for each value of v, there are no absorbing states.

The controller decides the value of $u(\cdot)$ for the t^{th} slot at the end of the $(t-1)^{th}$ slot. The decision is based on past histories of control values, departure and arrival data up to the decision time (time slot by time slot). Therefore the controller knows the queue sizes at decision times. In Chapter 3, we analyzed the partially observed case where the controller had available only arrival data. The difference here is that the controller does not know the values of the parameters $\{\lambda_i, \mu_i; i = 1,2\}$. They have to be estimated on the basis of the observed data $\{n_i^a(s), n_i^d(s); s \le t; i = 1,2\}$ for each decision epoch. We shall assume that $\{\lambda_i, \mu_i; i = 1,2\}$ are constant but unknown. Furthermore, we shall assume that the a priori information on these parameters is of the form:

$$\begin{aligned} &\mu_i \varepsilon M_i \qquad ,i = 1,2 \\ &\lambda_i \varepsilon \Lambda_i \qquad ,i=1,2 \end{aligned} \tag{4.2.7}$$

where (a) M_i, Λ_i are compact intervals or

(b) M_i, Λ_i are finite sets.

This latter assumption reflects a common practical situation where depending on measurement accuracy and quantization levels, some apriori information on system demand or server performance is summarized in the sets Λ_i, M_i.

The controller performance criterion is the expected long-run average cost per unit time denoted by:

$$J_a^\gamma = \liminf_{T\to\infty} \frac{1}{T} E[\sum_{t=0}^{T-1} c(x(t),u(t))] \tag{4.2.8}$$

where the instantaneous cost, $c(x(\cdot),u(\cdot))$ is linear in the state, $x(\cdot)$ and has the form:

$$c(x(t),u(t)) = c_1 x_1(t) + c_2 x_2(t) \tag{4.2.9}$$

and c_1, c_2 are positive constants modelling the relative weight the controller attaches on delays in queue 1 versus those occurring in queue 2. At each decision time, the controller assigns the value 1 or 0 to the control variable u(t) based on the following information:

$$\begin{aligned}
&\{n_i^a(s)\ ;\ s = 0,\ ],\ 2,\ \ldots\ ,\ t-1\} && \text{for } i = 1,2 \\
&\{n_i^d(s)\ ;\ s = 0,\ 1,\ 2,\ \ldots\ ,\ t-1\} && \text{for } i = 1,2 \qquad (4.2.10) \\
&\{u(s)\ ;\ s = 0,\ 1,\ 2,\ \ldots\ ,\ t-1\}
\end{aligned}$$

Let

$$\begin{aligned}
y^t &= \{n_i^a(s),\ n_i^d(s);\ s = 0,\ 2,\ \ldots\ ,\ t;\ i = 1,2\} \qquad (4.2.11) \\
u^t &= \{u(s);\ s = 1,\ 2,\ \ldots\ ,\ t\}.
\end{aligned}$$

We denote by Γ the set of admissible stationary control policies, whereby each $\gamma\varepsilon\Gamma$ has the form:

$$\gamma = (g,\ g,\ \ldots\ ), \qquad (4.2.12)$$

where

$$u(t) = g(y^{t-1},\ u^{t-1}) \qquad \text{for all } t=0,1,2,\ldots \qquad (4.2.13)$$

and each g takes values in {0, 1}. Since at all times the controller knows the queue sizes, then

$$x_i(t) = n_i^a(t) - n_i^d(t) + x_i(t-1) \quad ;\ i = 1,\ 2. \qquad (4.2.14)$$

Service is assumed to be non-preemptive and server idling is not allowed; specifically

$$u(t) = \begin{cases} 1, & \text{if } x_1(t-1)\neq 0,\ x_2(t-1) = 0 \\ \\ 0, & \text{if } x_1(t-1) = 0,\ x_2(t-1)\neq 1. \end{cases} \qquad (4.2.15)$$

If both queues are empty at a decision time then either decision is acceptable. The decision slots, i.e. the slots when control values can change, are either service completion slots or arrival slots when the other queue is empty. Finally, the superscript γ in (4.2.8) refers to the control strategy as defined by (4.2.12).

Our objective is to derive optimal strategies which are adaptive. Two methods for analyzing this adaptive control problem are considered. By the first method, known as the certainty-equivalence adaptive controller, the stochastic optimal control problem with known parameters $\{\lambda_i, \mu_i; \ i=1,2\}$ is considered (see results in Section 2.5). For the cost considered here (4.2.9), the optimal strategy is stationary; specifically $\gamma = \{g, g, g \ldots\}$, with $g(x) = g(\alpha, x)$ explicitly depending on the parameter values $\alpha=(\mu_1, \mu_2)$. At each time $t=1,2,\ldots$ the controller selects via a maximum likelihood criterion, an estimate of the unknown parameters, denoted by:

$$\hat{\alpha}(t) = (\hat{\mu}_1(t), \hat{\mu}_2(t)). \qquad (4.2.16)$$

He then uses the feedback control law of the form:

$$u(t) = g(\hat{\alpha}(t), x) \qquad (4.2.17)$$

as a candidate for the adaptive controller. By this method, the analysis focuses on the following issues:

(i) Does the parameter estimator, $\hat{\alpha}(t)$ converge almost surely?

(ii) Does the adaptive control scheme, $g(\hat{\alpha}(t),x)$ converge to the optimal control policy achieved if the true parameter were known apriori?

The second method augments the state space and treats the adaptive control problem as a stochastic control problem with partial observations. We apply the methodology developed in Chapter 3 to obtain the optimal control strategy. The resulting strategy is of course adaptive by construction.

4.3 Certainty Equivalence Adaptive Control

In this section, the expected long-run average aggregate delay (4.2.8), (4.2.9) problem for the queueing system (4.2.1) - (4.2.7) is considered. The adaptive control law is constructed as follows. At each decision epoch t, a maximum likelihood (ML) estimate $\hat{\alpha}(t)$ of the unknown parameters is made. If the maximizer is achieved by more than one value, we assume that only one of these is chosen according to some prescribed rule. Given the ML estimate $\hat{\alpha}(t)$, the certainty-equivalence adaptive controller selects the control action according to the rule $u(t) = g(\hat{\alpha}(t),x(t))$. For the average cost problem, the performance objective for known α is to minimize over the set of admissible stationary control policies Γ, defined in (4.2.11) - (4.2.13) such that

$$J_a^{\gamma^*}(x) = \min\{J_a^{\gamma}(x) : \gamma\varepsilon\Gamma\} \qquad (4.3.1)$$

for all $x\varepsilon\chi$ - the state space. From Theorem 2.5.1, the optimal stationary strategy $\gamma^*=(g,g,\ldots)\varepsilon\Gamma$ is of the form:

$$g(\alpha;i_1,i_2) = \begin{cases} 0 & \text{if } \mu_2c_2>\mu_1c_1;i_2\neq 0 \\ 1 & \text{if } \mu_1c_1>\mu_2c_2;i_1\neq 0 \\ \text{arbitrary} & \text{otherwise} \end{cases} \qquad (4.3.2)$$

The resulting "closed-loop" system is shown in Figure 4.2.

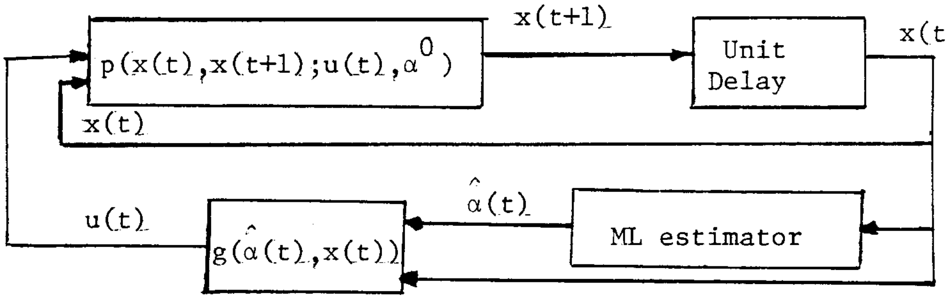


Figure 4.2. The closed-loop adaptive control system.

In the queueing system (4.2.1)-(4.2.7), the information available to the controller includes the past histories of the control, arrival and departure data. For the parameter estimator, two alternative ML estimates, $\hat{\alpha}(t)$ are developed; each has available different information. First, we consider the case where the parameter estimator has the <u>same</u> information available as the controller. The sequence of actions are described as follows. At the end of the (t-1)th time slot, the controller decides on which queue to serve next, based on the available information $\{y^{t-1},u^{t-1}\}$. During the t^{th} time slot, the arrivals $\{n_i^a(t);\ i=1,2\}$ and departures $\{n_i^d(t);\ i=1,2\}$ are observed. Based on the information $\{y^t,\ u^t\}$, the ML estimator $\hat{\alpha}(t+1)$ is selected. By

(4.2.15), this latter set contains the equivalent information as given by

$$\{n_i^a(s),\ x_i(s),\ u(s);\ s\leq t,\ i=1,2\}$$

Hence, the estimate $\hat{\alpha}(t+1)$ is chosen to maximize the likelihood function:

$$\begin{aligned} L(t,\alpha) &= \Pr\{n^a(s),\ x(s),\ u(s);\ s\leq t\,|\,x(0),\alpha\} \\ &= \prod_{s=0}^{t-1} \Pr\{x(s+1)\,|\,x(s),\ u(s+1),\ n^a(s+1),\alpha\} \\ &\cdot \Pr\{n^a(s+1)\,|\,x(s),\ u(s+1),\ n^a(s),\alpha\} \\ &\cdot \Pr\{u(s+1)\,|\,x(s),\ u(s),\ n^a(s),\alpha\} \qquad (4.3.3) \end{aligned}$$

where the second equality follows from properties of conditional independence, $n^a(t) = (n_1^a(t),\ n_2^a(t))$ and $n^d(t) = (n_1^d(t), n_2^d(t))$. The certainty equivalence adaptive control law need only estimate $\{\mu_i;\ i=1,2\}$ since the optimal strategy with known parameter (4.3.2) does not depend on the arrival rates $\{\lambda_i;\ i=1,2\}$. Namely, let $\alpha = (\mu_1,\mu_2)$ in (4.3.4), then the maximum likelihood estimate is given by:

$$\hat{\alpha}(t+1) = \arg\max\{L(t,\alpha):\alpha\varepsilon(M_1,M_2),\ M_i \text{ compact intervals}\} \qquad (4.3.4)$$

An explicit form of the ML estimate, $\hat{\alpha}(t+1)$ is obtained as follows. From (4.2.1) - (4.2.7), we have

$$\begin{aligned} &\Pr\{x(s+1)\,|\,x(s),\ u(s+1),\ n^a(s+1),\alpha\} \\ &= \prod_{i=1}^{2} \Pr\{x_i(s+1)\,|\,x_i(s),\ u(s+1),\ n_i^a(s+1),\alpha\} \qquad (4.3.5) \end{aligned}$$

where

$$\Pr\{x_1(s+1)|x_1(s),\ u(s+1),\ n_1^a(s+1),\alpha\}$$

$$= \begin{cases} \mu_1 u(s+1)\ I(x_1(s))\ n_1^a(s+1) & \text{if a departure occurs in the } t^{th} \text{ time slot of queue 1} \\ [1-\mu_1 u(s+1)\ I(x_1(s))](1-n_1(s+1)) & \text{otherwise} \end{cases} \quad (4.3.6)$$

$$I(x_1(s)) = \begin{cases} 1 & \text{if } x_1(s)\neq 0 \\ 0 & \text{otherwise} \end{cases} \quad (4.3.7)$$

and $\Pr\{x_2(s+1)|\cdot,\cdot,\cdot,\cdot\}$ is similarly defined. Also, it follows for the non-idling, service rate control, μc-rule that

$$\Pr\{n^a(s+1)|x(s),\ u(s+1),\ n^a(s),\alpha\} = \Pr\{n_1^a(s+1)\}\cdot\Pr\{n_2^a(s+1)\}\} \quad (4.3.8)$$

and

$$\Pr\{u(s+1)|x(s),\ u(s),\ n^a(s),\alpha\} = 1 \quad (4.3.9)$$

By combining (4.3.5) - (4.3.9) into (4.3.3), we have

$$L(t,\alpha) = L_1(t,\alpha)\ \cdot\ L_2(t,\alpha) \quad (4.3.10)$$

where

$$L_1(t,\alpha) = \prod_{s=0}^{t-1} [\mu_1\ u(s+1)\ I(x_1(s))n_1^a(s+1)]\ \prod_{s=0}^{t-1} [1-\mu_1 u(s+1)\ I(x_1(s))](1-n_1(s+1)) \quad (4.3.11)$$

$$L_2(t,\alpha) = \prod_{s=0}^{t-1} [\mu_2(1-u(s+1))\ I(x_2(s))n_2^a(s+1)]\ \prod_{s=0}^{t-1} [1-\mu_2(1-u(s+1))\ I(x_2(s))]\ (1-n_2(s+1)) \quad (4.3.12)$$

Note in (4.3.10), the probabilities associated with the arrival processes (4.3.8) have been neglected (independent of α). Now the event $\{n_1^a(s+1) = 1,\ u(s+1) = 1,\ I(x_1(s)) = 1\}$ reduces to the event $\{n_1^a(s+1) = 1\}$ so that (4.3.11) reduces to

$$L_1(t,\alpha) = \left[\prod_{\substack{s=0 \\ \text{s.t. } n_1^d(s+1)=1}}^{t-1} \mu_1 \quad \cdot \prod_{\substack{s=0 \\ \text{s.t. } n_1^d(s+1)=0,\ u(s+1)=1 \\ I(x_1(s) = 1}}^{t-1} (1-\mu_1) \right]$$

$$= \mu_1^{\Sigma n_1^d(s+1)} \cdot (1-\mu_1)^{\Sigma(1-n_1^d(s+1))u(s+1)\ I(x_1(s))} \qquad (4.3.13)$$

Let

$$a = \sum_{s=0}^{t-1} n_1^d(s+1) \ , \quad b = \sum_{s=0}^{t-1} u(s+1)\ I(x_1(s))$$

so that (4.3.13) becomes

$$L_1(t,\alpha) = \mu_1^a \cdot (1-\mu_1)^{b-a}$$

By (4.3.4), the ML estimate of μ_1 implies

$$a\ \mu_1^{a-1}(1-\mu_1)^{b-a} - (b-a)\mu_1^a(1-\mu_1)^{b-a-1} = 0$$

or equivalently

$$\mu_1^{a-1}(1-\mu_1)^{b-a-1}[a(1-\mu_1) - (b-a)\mu_1] = 0$$

so that

$$\hat{\mu}_1(t+1) = \frac{a}{b} = \frac{\sum_{s=0}^{t-1} n_1^d(s+1)}{\sum_{s=0}^{t-1} u(s+1)\ I(x_1(s))} \qquad (4.3.14a)$$

Similarly, it follows from (4.3.4), (4.3.12)

$$\hat{\mu}_2(t+1) = \frac{\sum_{s=0}^{t-1} n_2^d(s+1)}{\sum_{s=0}^{t-1} (1-u(s+1))\ I(x_2(s))} \qquad (4.3.14b)$$

In case the denominates in (4.3.14) are unchanged, the previously estimated value is used.

For the adaptive control scheme with parameter estimates generated by (4.3.14) and control law $g(\hat{\alpha};x)$ as in (4.3.2), we have the following result:

Lemma 4.3.1. If the controller satisfies the condition

$$\sum_{s=0}^{t-1} u(s+1)\ I(x_1(s)) = 0(t) \qquad \text{as } t\to\infty$$

$$\sum_{s=0}^{t-1} (1-u(s+1))\ I(x_2(s)) = 0(t) \text{ as } t\to\infty$$

where $F(t) = 0(t)$ denotes that for t large, the function $F(\cdot)$ grows linearly in t, then the parameter estimates $\hat{\alpha}(t) = (\hat{\mu}_1(t),\hat{\mu}_2(t))$ converge a.s. to the true parameters $\alpha^0 = (\mu_1^0,\mu_2^0)$ and the adaptive control law $g(\hat{\alpha}(t);\cdot)$ converges a.s. to the optimal policy achieved if the true parameters were known apriori.

Proof: Without loss of generality, we consider the convergence of $\hat{\mu}_1(t)$ in (4.3.14). By (A1), we have by the law of large numbers

$$\lim_{t\to\infty} \hat{\mu}_1(t) = \lim_{t\to\infty} \frac{\sum_{s=0}^{t-1} n_1^d(s)}{\sum_{s=0}^{t-1} u(s)\ I(x_1(s-1))} = E[n_1^d(s)]=\mu_1 \text{ (a.s.)} \tag{4.3.15}$$

Consequently by the continuity of the μc-rule to the parameters $\{\hat{\mu}_1;\ i=1,2\}$, we have by (4.3.2)

$$\lim_{t\to\infty} g(\hat{\mu}_1(t),\hat{\mu}_2(t);x) = g(\mu_1,\mu_2;x) \qquad \text{(a.s.)}$$

$$\text{for all } x\varepsilon\chi \tag{4.3.16}$$

QED.

Remark 4.3.1. Heuristically, one can view condition (A1) as follows. First, suppose the system always serves queue 1 (i.e. $u(t) = 1$ for all t). Then the ML estimate (4.3.4) reduces to the standard one for a single queue [19]; specifically $\hat{\mu}_1(t)$ converges to μ_1 a.s. and $\hat{\mu}_2(t) = 0$ a.s. for all t. The converse is true when the system always serves queue 2. The condition (A1) implies that the server switch infinitely often between the two queues, requiring certain stability conditions on the queueing system $\{\lambda_i,\mu_i: i=1,2\}$ and on the stationary policy (4.3.2). Sufficient conditions to insure that (A1) holds is an area of future research. For the unbounded system, it can be shown that if

$$\frac{\lambda_1}{\mu_1} + \frac{\lambda_2}{\mu_2} > 1 \tag{4.3.17}$$

then the system is unstable, i.e. one queue grows without bound. Hence, the parameter estimates (4.3.14) and adaptive control law are degenerate.

An alternative ML estimate is obtained by using different information than that used by the controller. The information available to the estimator, in this case includes only the past histories of the control and the queue sizes, $\{u^t,x^t\}$. Clearly, this is a reduced set of information. The estimate $\hat{\alpha}(t+1)$ is chosen to minimize the likelihood function:

$$
\begin{aligned}
L(t,\alpha) &= \Pr\{x(s),u(s);s\leq t\,|\,x(0),\alpha\} \\
&= \prod_{s=0}^{t-1} \Pr\{x(s+1)\,|\,x(s),u(s+1),\alpha\}\cdot\Pr\{u(s+1)\ x(s),u(s),\alpha\} \\
&= \prod_{s=0}^{t-1} p(x(s),x(s+1);u(s+1),\alpha) \qquad (4.3.18)
\end{aligned}
$$

where the second equality follows from conditional independence and the last equality follows from (4.1.1), (4.3.9). From (4.2.1) - (4.2.7), we have

$$
\begin{aligned}
&p(x(t),x(t+1);v,\alpha) \\
&= p^1(i_1,j_1;v,\alpha)\cdot p^2(i_2,j_2;v,\alpha) \qquad (4.3.19)
\end{aligned}
$$

for $x(t) = (i_1,i_2)$, $x(t+1) = (j_1,j_2)$ and $\alpha = (\mu_1,\mu_2)$ where

$$
p^1(i_1,j_1;v,\alpha) = \begin{cases} (1-\lambda_1)\mu_1 v & \text{if } j_1 = i_1-1\ ;\ i_1\neq 0 \\ (1-\lambda_1)(1-\mu_1 v) + \lambda_1\mu_1 v & \text{if } j_1 = i_1 \\ \lambda_1(1-\mu_1 v) & \text{if } j_1 = i_1+1 \\ 0 & \text{elsewhere} \end{cases} \qquad (4.3.20)
$$

and $p^2(\cdot,\cdot;\cdot,\cdot)$ is similarly defined. The resulting ML estimates $\hat{\alpha}(t+1) = (\hat{\mu}_1(t+1),\ \hat{\mu}_2(t+1))$, given (4.3.4), (4.3.19) - (4.3.20), follows from the solution of the quadratic equation:

$$
\begin{aligned}
&(1-\lambda_i)N_i^3(t)\left[\frac{1-\hat{\mu}_i(t+1)}{\hat{\mu}_1(t+1)}\right]^2 \\
&+ [(2\lambda_i-1)N_i^2(t)+\lambda_i N_i^3(t)-(1-\lambda_i)N_i^1(t)]\left[\frac{1-\hat{\mu}_i(t+1)}{\hat{\mu}_i(t+1)}\right] \\
&= \lambda_i N_i^1(t) \qquad \text{for } i = 1,2 \qquad (4.3.21)
\end{aligned}
$$

where

$$N_1^1(t) = \sum_{s=0}^{t-1} I_{\{j_1>i_1\}},\quad N_1^2(t) = \sum_{s=0}^{t-1} I_{\{j_1=i_1\}} \text{ and}$$

$$N_1^3(t) = \sum_{s=0}^{t-1} I_{\{j_1<i_1\}} \qquad (4.3.22)$$

The counting processes $\{N_2^i(t);\ i = 1,2,3\}$ are similarly defined and the function, $I_{\{\cdot\}}$ denotes the indicator function of the specified set.

Remark 4.3.2. The ML estimate (4.3.21) appears initially to be more complex than the earlier one presented (4.3.14). Clearly, its convergence properties are not as easily state as in (4.3.14). However, it may follow that the equivalent sufficient condition (A1) for estimate (4.3.21) is more physically apparent due to its dependency only on the queue size and not on the control values. This is an open research problem.

Remark 4.3.3. The implementation of either estimator (4.3.14), (4.3.21) is straightforward. In the former, three adders are required while in the latter six are needed. Recursive expressions for the ML estimate in (4.3.14) have been obtained. To develop an understanding of the adaptive control scheme, a numerical evaluation is presented in Section 4.5.

4.4 Partial Observation Adaptive Control

In this section, the adaptive control problem of Section 4.2 is considered as a stochastic control problem with partial observations. In this case, the parameters $\{\lambda_i,\ \mu_i;\ i=1,2\}$ are assumed to be unknown constants and are treated as additional states. Furthermore, we will assume that the apriori information on these paramaters is of the form:

$$\mu_i \varepsilon M_i \ , \ \lambda_i \varepsilon \Lambda_i \qquad \text{for } i = 1,2 \tag{4.4.1}$$

where the finite parameter sets are given by:

$$M_i = \{\mu_i^0, \mu_i^1, \ldots , \mu_i^{r_i}\}$$

$$\Lambda_i = \{\lambda_i^0, \lambda_i^1, \ldots , \lambda_i^{s_i}\} \qquad \text{for } i = 1,2 \tag{4.4.2}$$

The true parameter values $\{\lambda_i^0, \mu_i^0: i = 1,2\}$ are assumed to be elements in these sets.

Following the development of Chapter 3, (3.3.52) - (3.3.55), we define the state and observation spaces as follows:

$$x(t) = (x_1(t), x_2(t), \mu_1(t), \mu_2(t), \lambda_1(t), \lambda_2(t)) \varepsilon \chi$$

where

$$\chi = Z \times Z \times M_1 \times M_2 \times \Lambda_1 \times \Lambda_2 \tag{4.4.3}$$

and

$$y(t) = (n_1^a(t), n_2^a(t), n_1^d(t), n_2^d(t)) \varepsilon Y \tag{4.4.4}$$

The combined joint statistics of the observations and state transitions is denoted by:

$$S_{ij}(t,v,\psi) = \Pr\{x(t+1) = j, y(t) = \psi | x(t)=i, u(t)=v\} \tag{4.4.5}$$

where

$$i = (i_1, i_2, m_1, m_2, \ell_1, \ell_2), \ j \varepsilon \chi \quad \psi \varepsilon Y$$

and $v \varepsilon U = \{0,1\}$. Note, the additional state parameters $\{\lambda_i(t), \mu_i(t): i = 1,2\}$ are unobservable while the queue sizes $\{x_i(t);\ i = 1,2\}$ are observable. Hence, we have a partial observed stochastic control problem within the framework of Chapter 3.

To proceed within the framework of (3.3.1) (3.3.4) (3.3.53) (3.3.54), we need to specify the joint statistics (4.4.5) in terms of these parameters. By properties of conditional probabilities and independence, we have

$$\begin{aligned}
&\Pr\{x(t+1) = j,\ y(t) = \psi \mid x(t) = i,\ u(t) = v\} \\
&= \Pr\{x(t+1) = j,\ y(t) = \psi \mid x(t) = i,\ u(t) = v,\ \mu_1(t) = \mu_1^{n_1}, \\
&\qquad \mu_2(t) = \mu_2^{n_2},\ \lambda_1(t) = \lambda_1^{k_1},\ \lambda_2(t) = \lambda_2^{k_2}\} \\
&\cdot \Pr\{\mu_1(t) = \mu_1^{n_1} \mid x(t) = i,\ u(t) = v\} \\
&\cdot \Pr\{\mu_2(t) = \mu_2^{n_2} \mid x(t) = i,\ u(t) = v\} \\
&\cdot \Pr\{\lambda_1(t) = \lambda_1^{k_1} \mid x(t) = i,\ u(t) = v\} \\
&\cdot \Pr\{\lambda_2(t) = \lambda_2^{k_2} \mid x(t) = i,\ u(t) = v\} \qquad (4.4.6)
\end{aligned}$$

where for $i = (i_1,i_2,m_1,m_2,\ell_1,\ell_2)\varepsilon\chi$

$$\Pr\{\mu_1(t) = \mu_1^{n_1} \mid x(t) = i,\ u(t) = v\} = \begin{cases} 1 & \text{if } n_1=m_1 \\ 0 & \text{otherwise} \end{cases} \qquad (4.4.7)$$

$$\Pr\{\lambda_1(t) = \lambda_1^{k_1} \mid x(t) = i,\ u(t) = v\} = \begin{cases} 1 & \text{if } k_1= \ell_1 \\ 0 & \text{otherwise} \end{cases} \qquad (4.4.8)$$

and similarly for the other terms in (4.4.6). In other words, the transition probabilities for the parameters $\{\lambda_i(t),\ \mu_i(t);\ i = 1,2\}$ are identity matrices. The first term of (4.4.6) is characterized in (3.3.42) - (3.3.45), (3.3.53), (3.3.54) with the exception here that the parameters $\{\lambda_i, \mu_i; i = 1,2\}$ are constants, independent of time and queue size.

In Chapter 3, the formulation was the finite horizon average aggregate delay. For the performance criterion (4.2.8), (4.2.9), the approach introduced in Section 4.3 may be applicable; specifically treat the expected long-run average cost as the limiting case of the finite horizon problem. This is an area of future research. The main utility of analyzing the adaptive scheme as a partially observed stochastic control problem is to obtain performance estimates for more practical schemes, such as the one studied in Section 4.3. An analytical solution of this problem has not been obtained; the methods presented here lead to a numerical treatment similar to that of Chapter 3.

4.5 Evaluation - Finite Queue System

In this section, the adaptive control scheme of Section 4.3 for the bounded queueing system (4.2.1) - (4.2.7) is considered. For a finite capacity system, the arrival rates on the upper boundary states are zero (see equation (2.2.6)). The discussion here leads to a numerical treatmemt of the problem; these results augment the discussion of Section 2.6. We intend to demonstrate via numerical examples the behavior of the queueing system, under the average cost optimal, adaptive and μc-rule control strategies.

The bounded queueing system is modelled as described in (4.2.1) - (4.2.7) with the performance objective (4.2.8), (4.2.9), (4.3.1). A finite buffer size ($N = N_1 = N_2 = 7$) is simulated, with each queue initialized to five (5) customers. For a finite horizon ($T = 100$), Bernoulli arrivals and departures are generated at each time step such that

(i) no customers arrive in a queue when it is full (see (4.2.3), (4.2.4)).

(ii) no customers depart from queue i when either queue i has zero customers on queue $j(j \neq i)$ is being served (see 4.2.15).

The selection of the control sequence depends on the respect policy under consideration. For the optimal average cost policy, the parameters $\{\lambda_i, \mu_i; \ i = 1,2\}$ are assumed known. The optimal policy is generated using the policy iteration method (2.6.7), (2.6.8) of Howard [68]. Because the parameters remained fixed throughout the sample path, the policy iteration is only invoked once. For the μc-rule strategy, the sample paths were generated for comparison with the other two policies. Clearly for the finite buffer case, the μc-rule is suboptimal, as noted in Section 2.6. The adaptive control strategy was implemented as follows. At each time step, the parameter estimates $\hat{\mu}_1(\cdot)$, $\hat{\mu}_2(\cdot)$ are updated using (4.3.14). Given these parameter values with $\{\lambda_i; \ i = 1,2\}$ known, the stationary control strategy (4.3.2) is generated using the policy iteration method (2.6.7), (2.6.8). Because of multiple calls to the policy iteration program, the excution time of adaptive control sample paths were exceedingly long.

Given these preliminaries, the cases studied are shown in Table 3.1. In each figure, the queue sizes $\{x_i(t);\ i = 1,2\}$, the control values, u(t) and the running average cost

$$J_a^\gamma(t) = \frac{1}{t} \sum_{s=0}^{t-1} [c_1 x_1(t) + c_2 x_2(t)]$$

as a function of time are shown, respectively in (a) - (d). For the optimal average cost strategy, the resulting optimal policy for the specified parameters is shown (see Section 2.6 for notational convention). For the adaptive control scheme, both the apriori (true) optimal policy and the adaptive control strategy (converged value) are shown. In addition, the parameter estimates $\{\hat{\mu}_i(t):\ i = 1,2\}$ are shown respectively in (e), (f). Note in Figures 2.6 - 2.8 that the parameter values are the same as those of Figure 2.4.

In Figures 4.3 - 4.5, the queueing system under each strategy is shown to perform quite similarly. Each system has the same (final) running average cost (4.5.1). A comparison of the average cost to the μc-rule shows that their sample paths are identical. This can be explained as follows. The two policies differ only in the states $\{(1,0), (4,1), (7,4), (7,5), (i,j):\ i = 5,6,7;\ j = 2,3,4\}$. Since the state process $\{x(t)\}$ never enters these states, the control value sequence are identical. A comparison of the average cost to the adaptive control strategy is more involved. Up until the first service time to queue 2, the sample paths are identical; queue 1 is always serviced. After queue 2 is serviced, each queueing system performs differently. Under the adaptive scheme, the system goes into a learning mode attempting to determine the departure rate, $\hat{\mu}_2(t)$ (Figure 4.4(f)). The learning exercise is at the expense of increased cost for queue 1 (Figure 4.4(a), (b)). After queue 2 empties (t = 52), the adaptive system returns to a mode similar to the optimal average cost system (compare Figure 4.3, 4.4(a), (b)). Observe that during the learning cycle ($40 \leq t < 60$), the running average cost, $J_a^\gamma(\cdot)$ is slightly higher for the adaptive system compared to the optimal average cost systems (Figure 4.3, 4.4(d)). The parameter estimates $\{\hat{\mu}_i(t);\ i = 1,2\}$ and

Table 4.1 - Finite Queue System (N=7)

Figure	Strategy	λ_1	c_1	λ_2	c_2	μ_1	μ_2	Optimal Cost	$J_a^\gamma(T)$
4.3	Optimal	.40	1.00	.25	2.00	.60	.20	14.56	13.23
4.4	Adaptive	.40	1.00	.25	2.00	.60	.20	14.56*	13.23
4.5	μc-rule	.40	1.00	.25	2.00	.60	.20	-	13.23
4.6	Optimal	.40	1.00	.20	2.00	.60	.20	13.88	10.05
4.7	Adaptive	.40	1.00	.20	2.00	.60	.20	13.88	12.47
4.8	μc-rule	.40	1.00	.20	2.00	.60	.20	-	12.49

*Optimal cost computed from converged parameter values

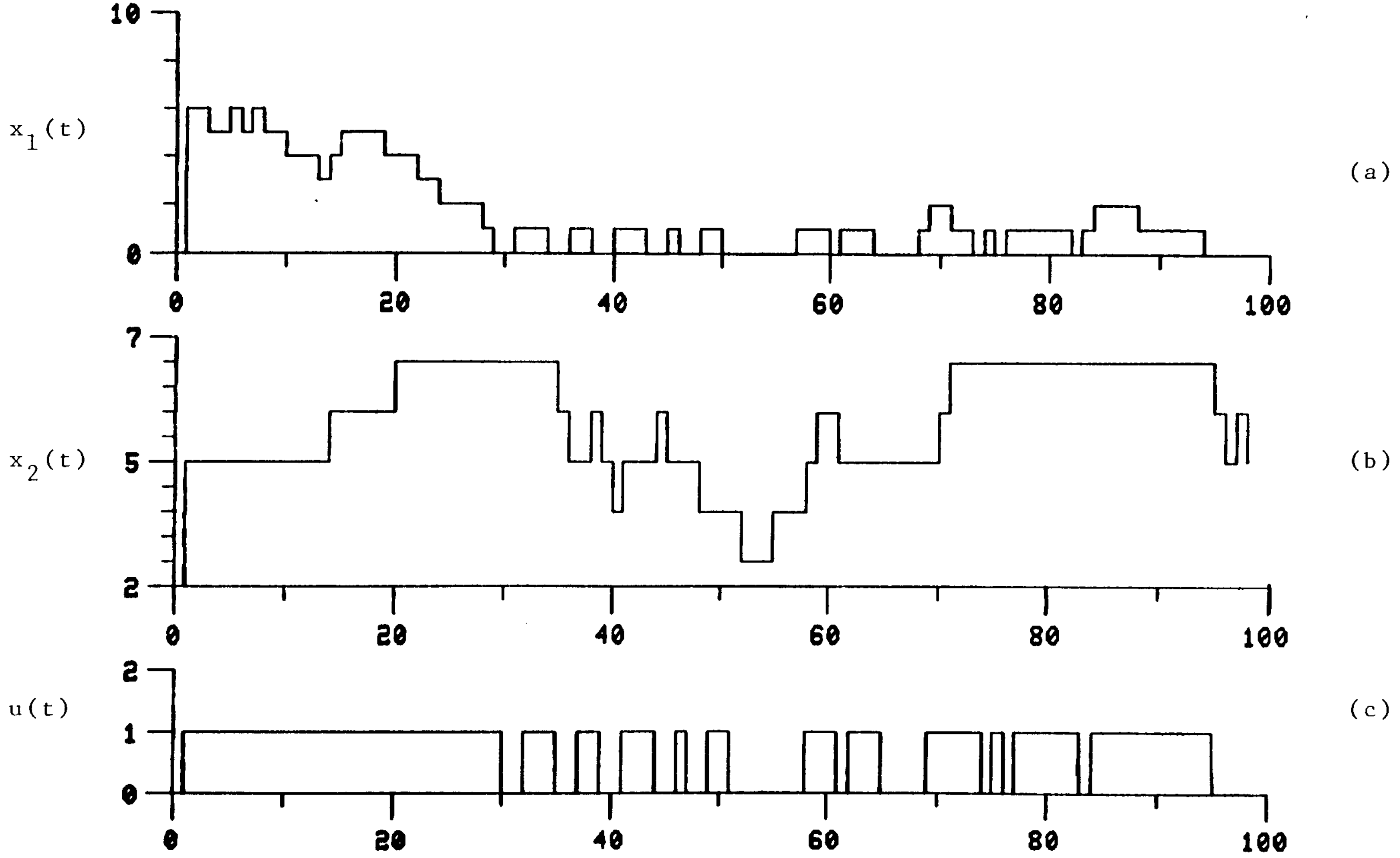
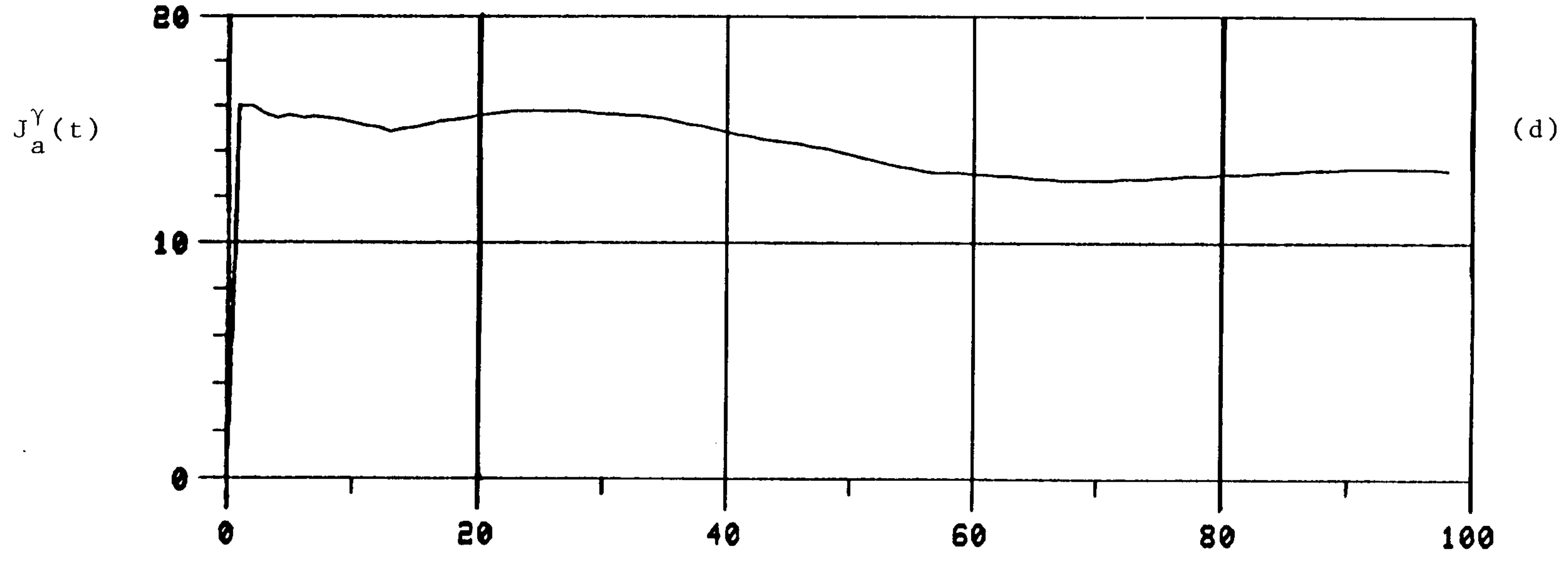


Figure 4.3 Strategy·Optimal Average Cost $\lambda_1 = 0.4$, $c_1 = 1.0$, $\mu_1 = 0.6$

Optimal Cost = 14.56 $\lambda_2 = 0.25$, $c_2 = 2.0$, $\mu_2 = 0.2$



OPTIMAL CONTROL FINITE QUEUE CASE

```
0 1 1 1 1 1 1 1
0 1 1 1 1 1 1 1
0 1 1 1 1 1 1 0
0 1 1 1 1 1 1 0
0 1 1 1 1 0 0 0
0 1 1 1 1 0 0 0
0 1 1 1 0 0 0 0
2 0 1 1 1 1 1 1
```

Figure 4.3 Strategy: Optimal Average Cost (Cont.)

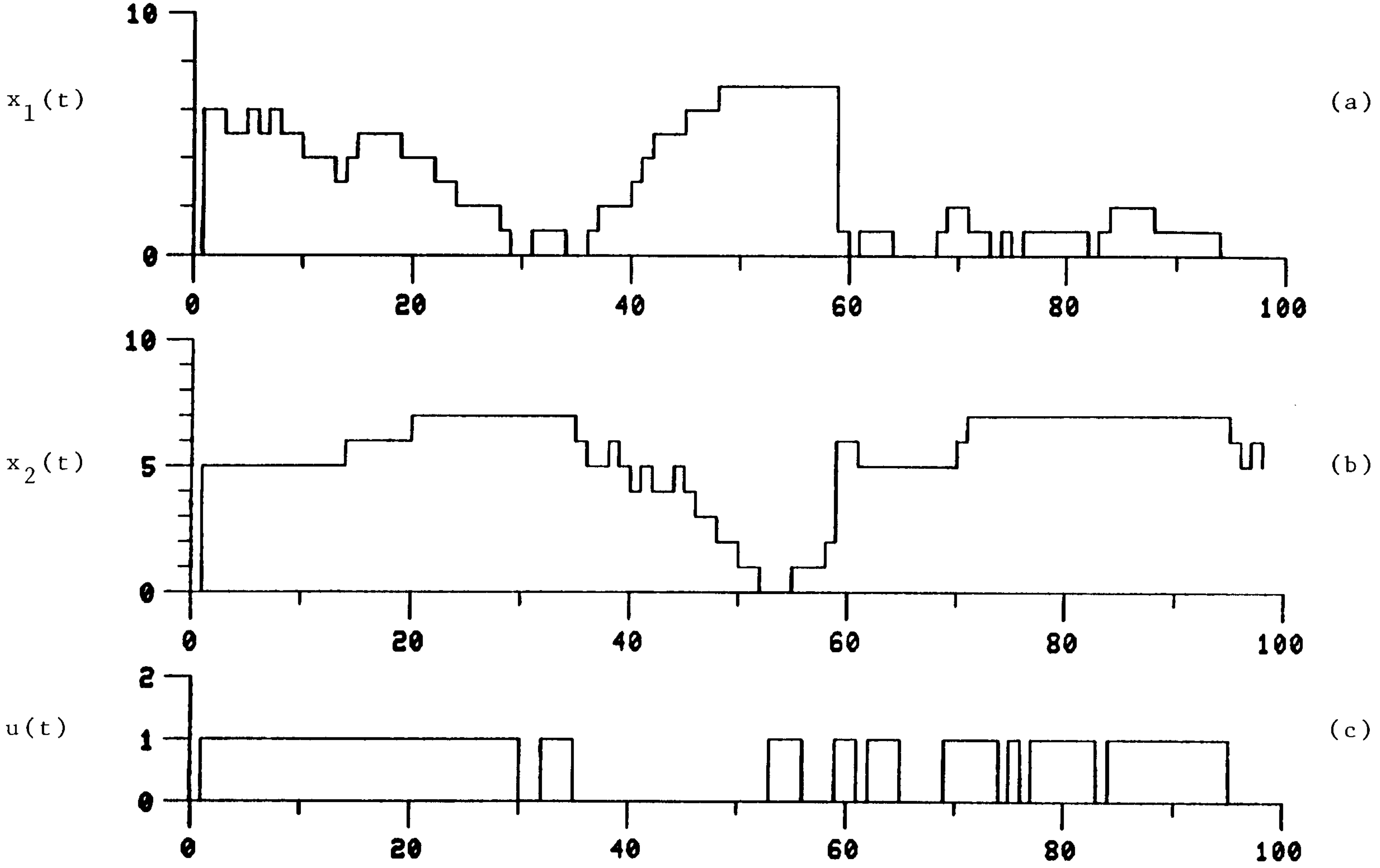
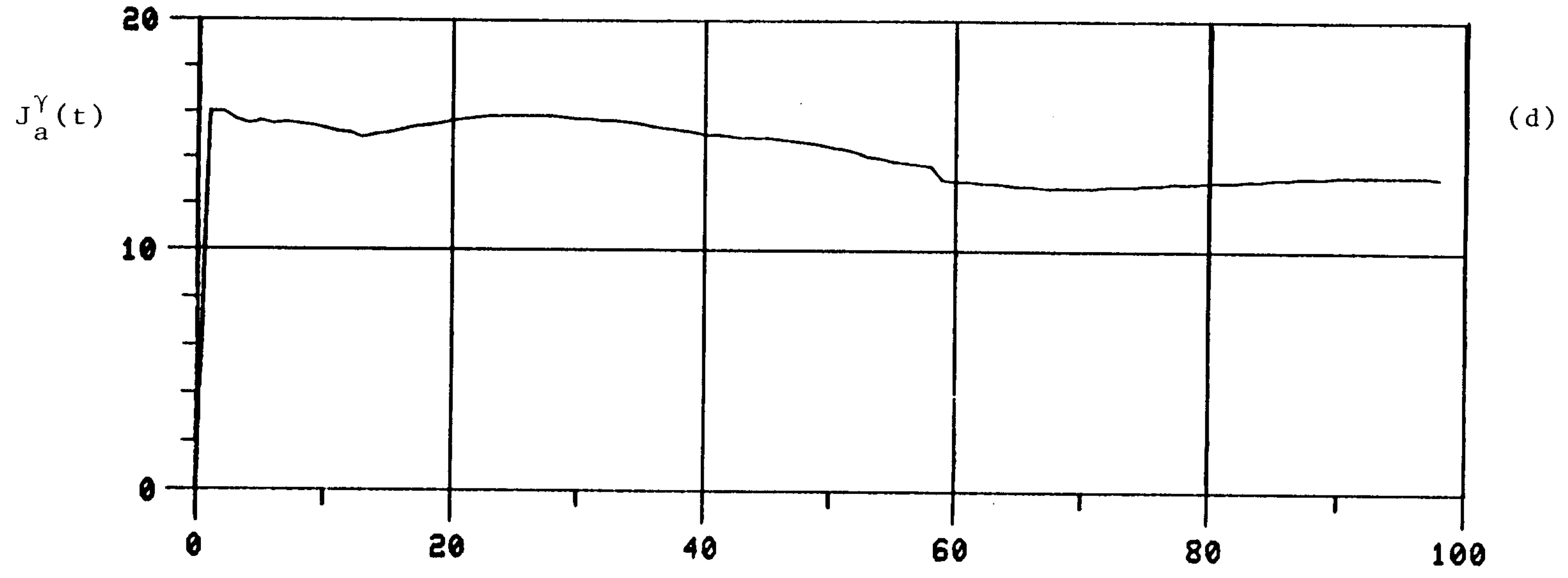


Figure 4.4 Strategy:Adaptive Control $\lambda_1 = 0.4$, $c_1 = 1.0$, $\mu_1 = 0.6$

Optimal Cost = 14.56 $\lambda_2 = 0.25$, $c_2 = 2.0$, $\mu_2 = 0.2$



```
OPTIMAL CONTROL FINITE QUEUE CASE          ADAPTIVE CONTROL FINITE QUEUE CASE
            0 1 1 1 1 1 1 1                             0 1 1 1 1 1 1 1
            0 1 1 1 1 1 1 1                             0 1 1 1 1 1 1 1
            0 1 1 1 1 1 1 0                             0 1 1 1 1 1 1 0
            0 1 1 1 1 1 1 0                             0 1 1 1 1 1 1 0
            0 1 1 1 1 0 0 0                             0 1 1 1 1 0 0 0
            0 1 1 1 1 0 0 0                             0 1 1 1 1 0 0 0
            0 1 1 1 0 0 0 0                             0 1 1 1 0 0 0 0
            2 0 1 1 1 1 1 1                             2 0 1 1 1 1 1 1
```

Figure 4.4 Strategy:Adaptive Control (Cont.)

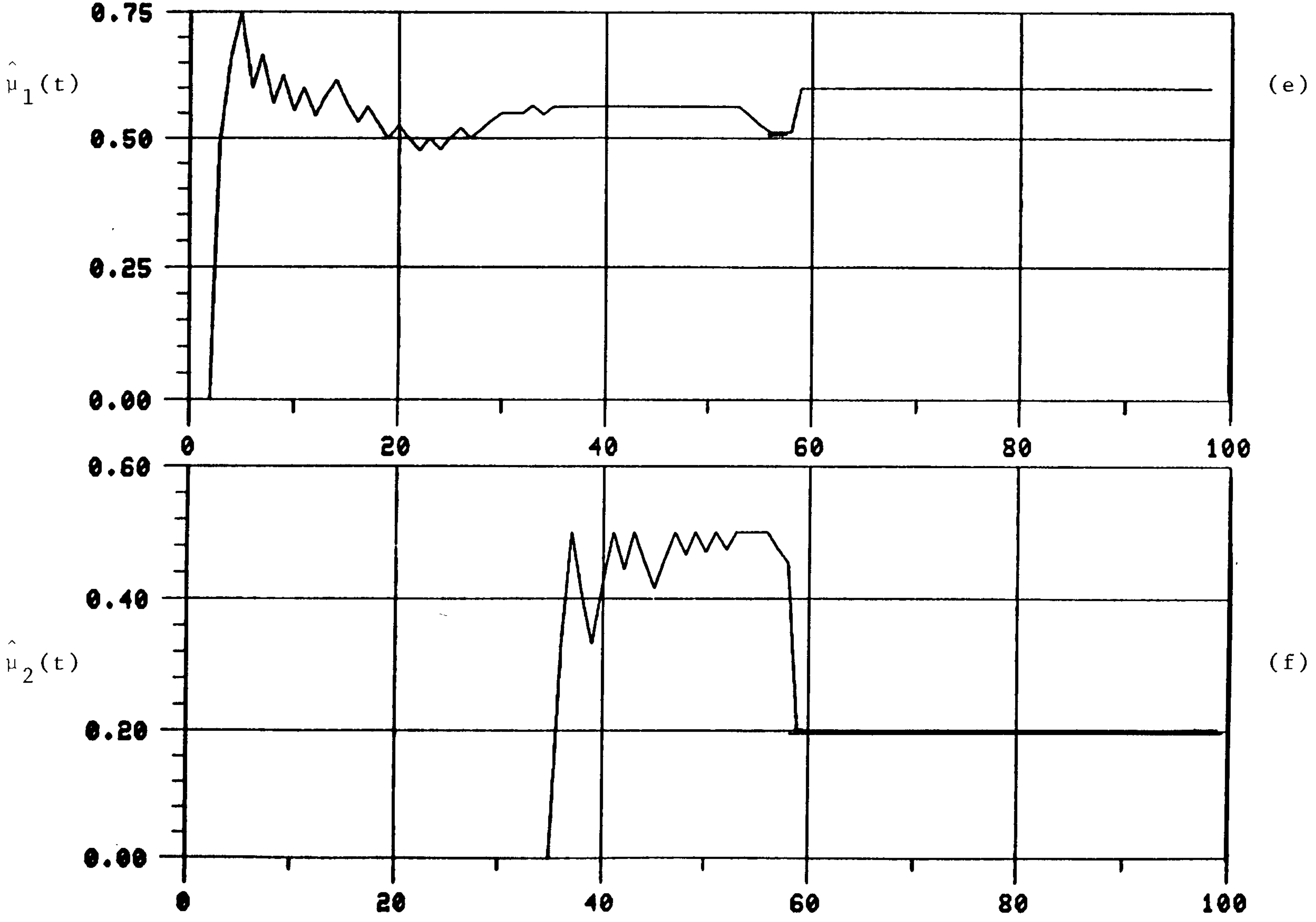


Figure 4.4 Strategey·Adaptive Control (Cont.)

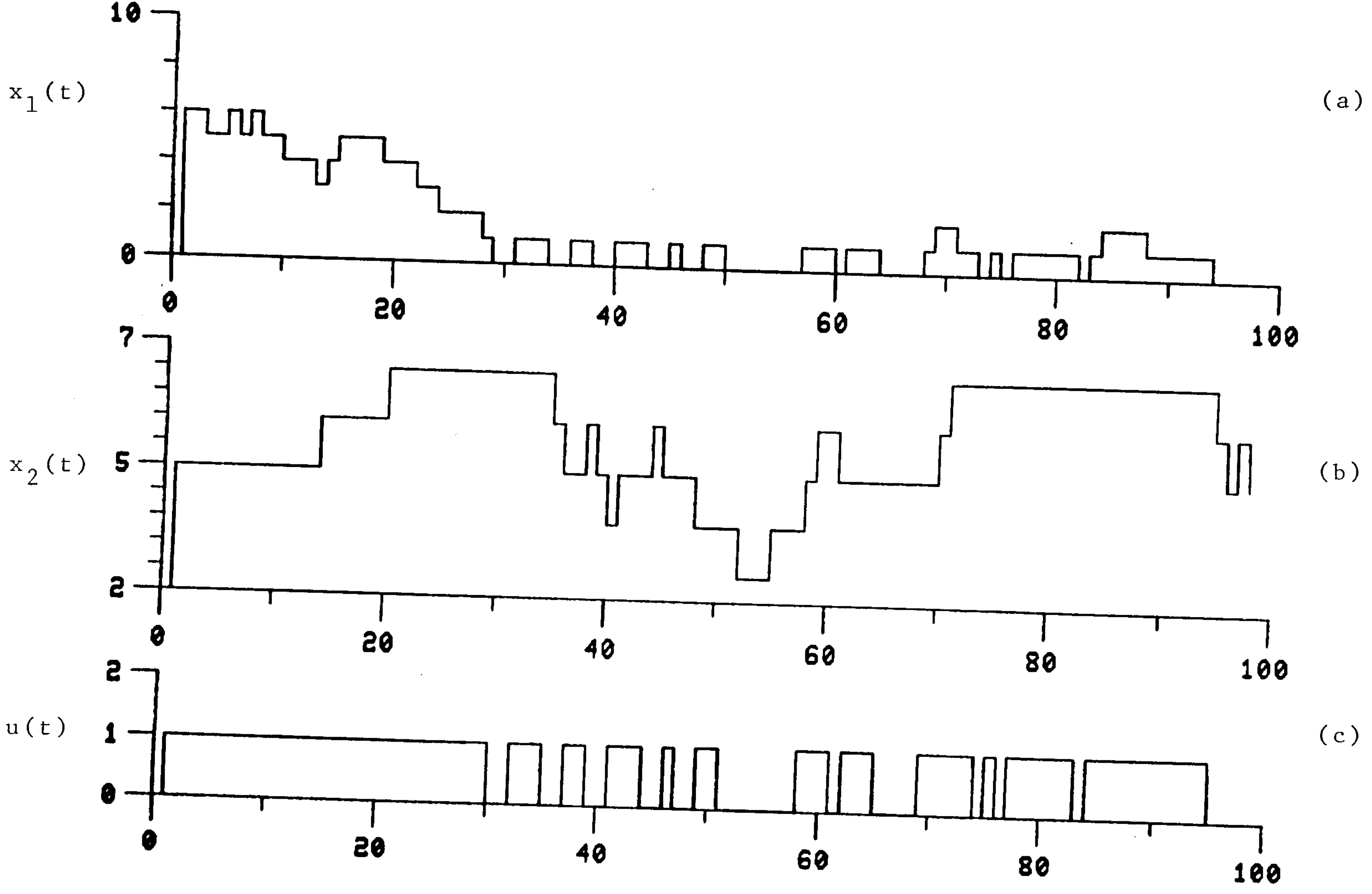


Figure 4.5 Strategy:μc-rule

$\lambda_1 = 0.4, \ c_1 = 1.0, \ \mu_1 = 0.6$

$\lambda_2 = 0.25, \ c_2 = 2.0, \ \mu_2 = 0.2$

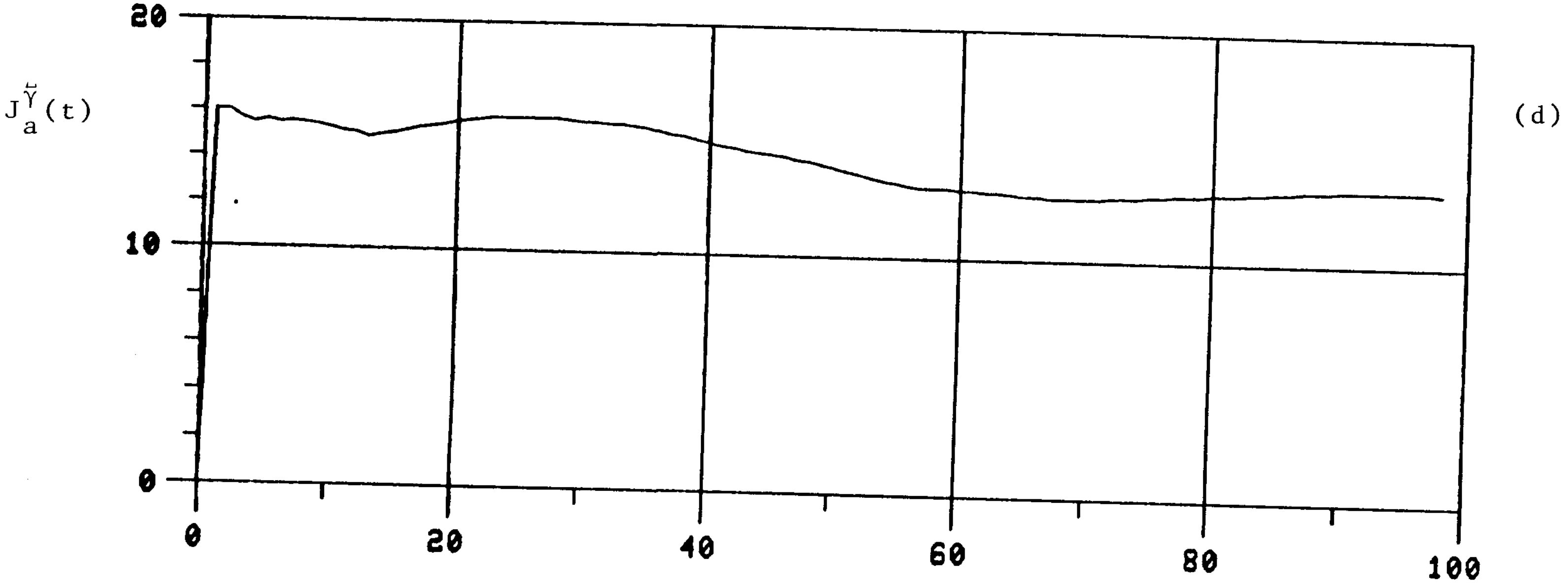


Figure 4.5 Strategy: μc-rule (Cont.)

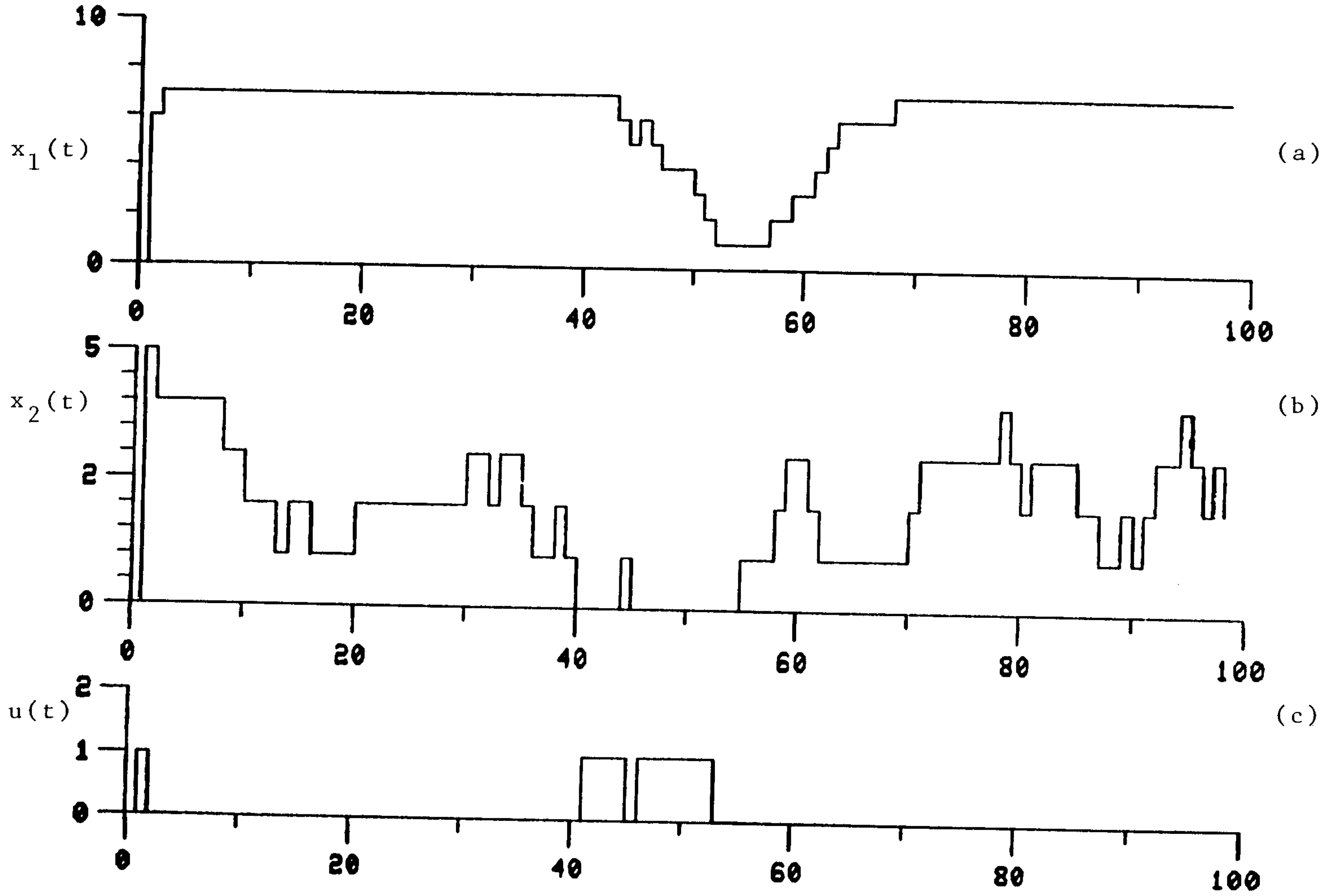
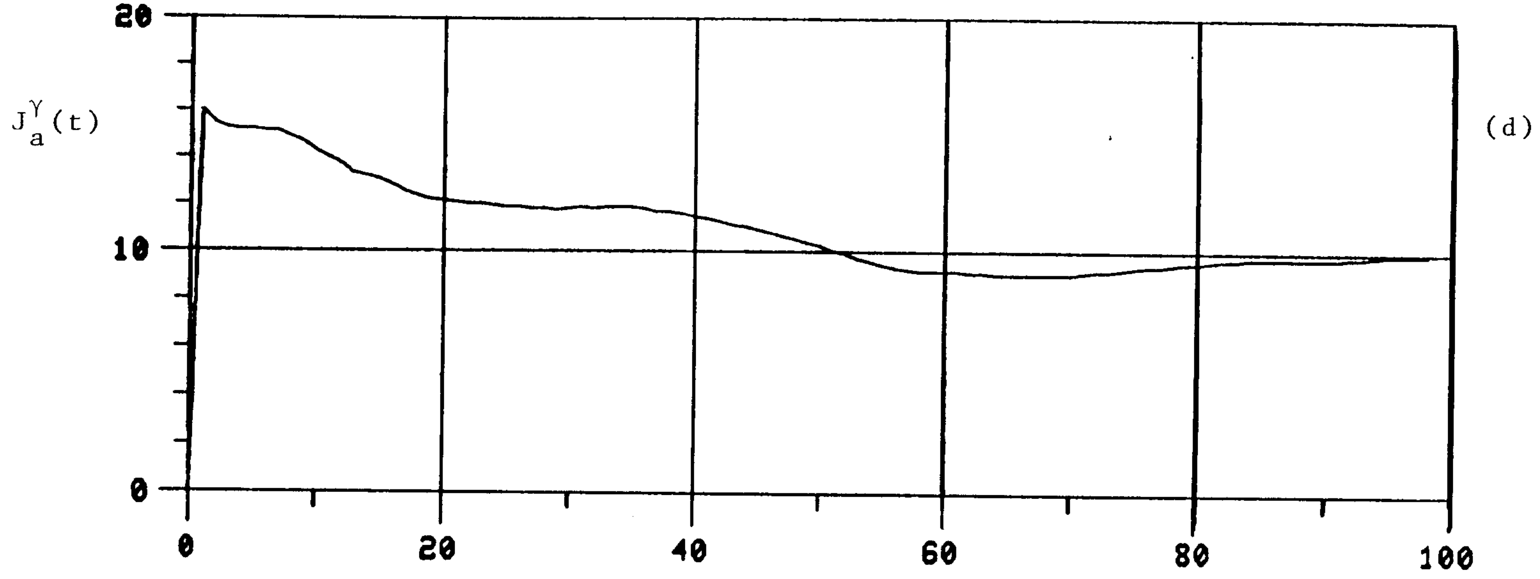


Figure 4.6 Strategy:Optimal Average Cost $\lambda_1 = 0.4$, $c_1 = 1.0$, $\mu_1 = 0.6$

Optimal Cost = 13.88 $\lambda_2 = 0.2$, $c_2 = 2.0$, $\mu_2 = 0.2$



```
OPTIMAL CONTROL FINITE QUEUE CASE
            0 1 1 1 1 1 1 0
            0 1 1 1 1 0 0 0
            0 0 0 0 0 0 0 0
            0 0 0 0 0 0 0 0
            0 0 0 0 0 0 0 0
            0 0 0 0 0 0 0 0
            0 0 0 0 0 0 0 0
            2 0 1 1 1 1 1 1
```

Figure 4.6 Strategy·Optimal Average Cost (Cont.)

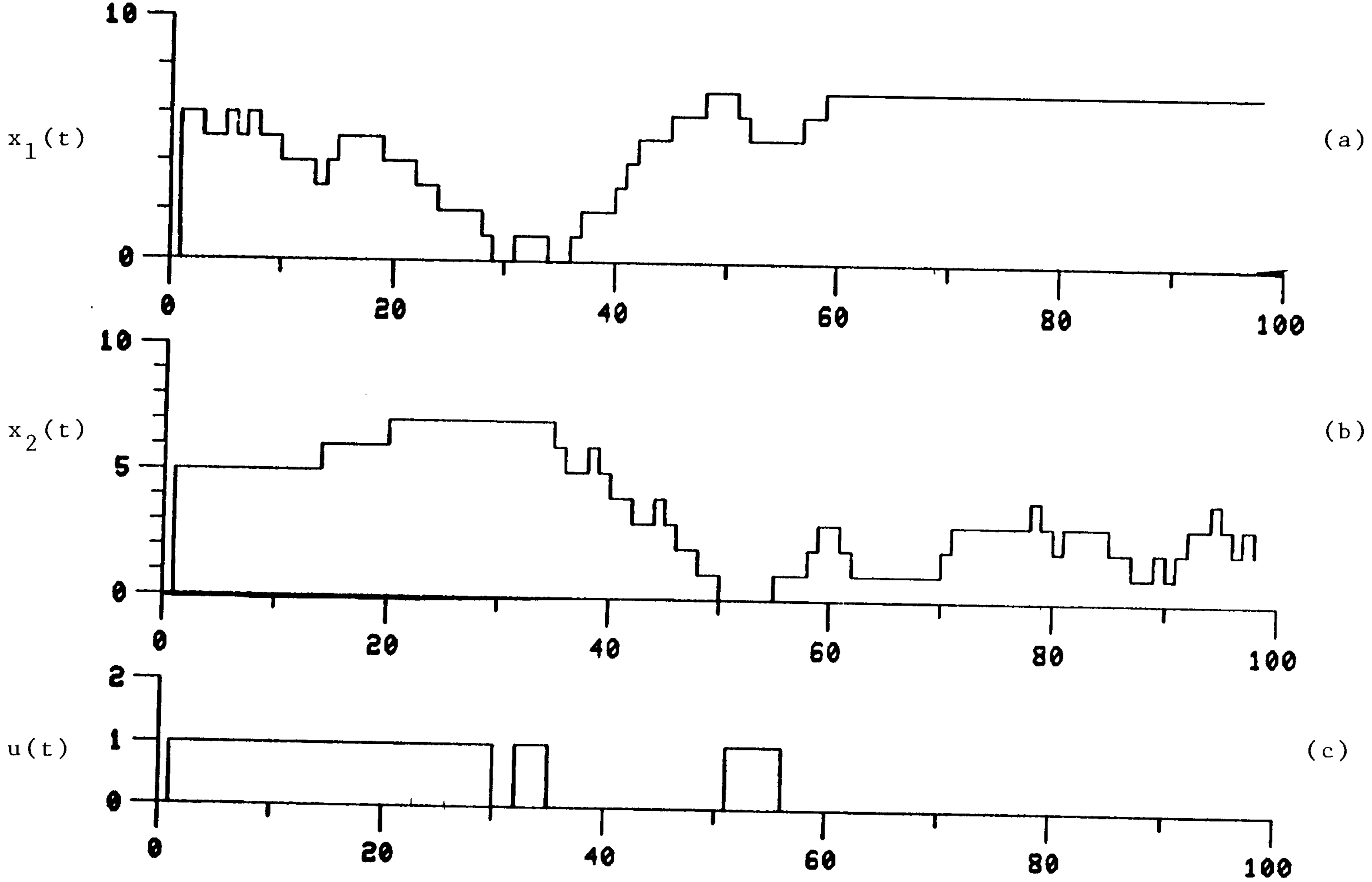
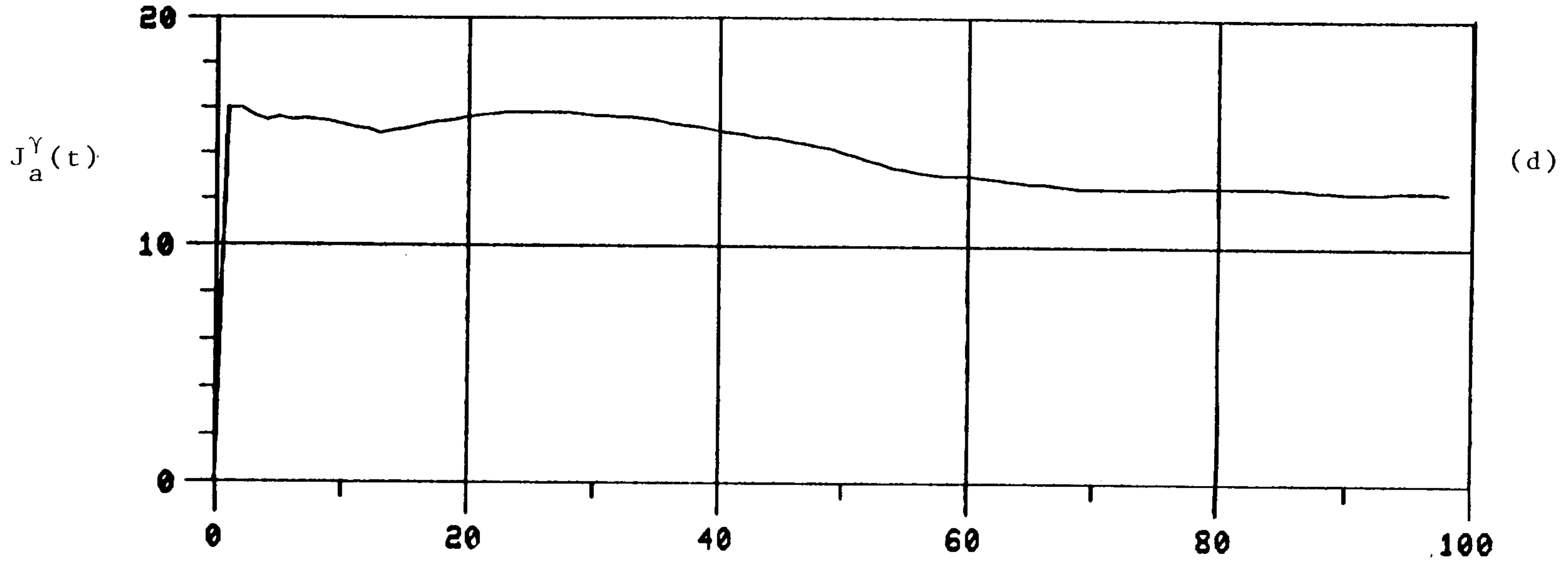


Figure 4.7 Strategy: Adaptive Control $\lambda_1 = 0.4$, $c_1 = 1.0$, $\mu_1 = 0.6$

Optimal Cost = 13.88 $\lambda_2 = 0.2$, $c_2 = 2.0$, $\mu_2 = 0.2$



```
OPTIMAL CONTROL FINITE QUEUE CASE   ADAPTIVE CONTROL FINITE QUEUE CASE
       0 1 1 1 1 1 1 0                       0 0 0 0 0 0 0 0
       0 1 1 1 1 0 0 0                       0 0 0 0 0 0 0 0
       0 0 0 0 0 0 0 0                       0 0 0 0 0 0 0 0
       0 0 0 0 0 0 0 0                       0 0 0 0 0 0 0 0
       0 0 0 0 0 0 0 0                       0 0 0 0 0 0 0 0
       0 0 0 0 0 0 0 0                       0 0 0 0 0 0 0 0
       0 0 0 0 0 0 0 0                       0 0 0 0 0 0 0 0
       2 0 1 1 1 1 1 1                       2 1 1 1 1 1 1 1
```

Figure 4.7 Strategy:Adaptive Control (Cont.)

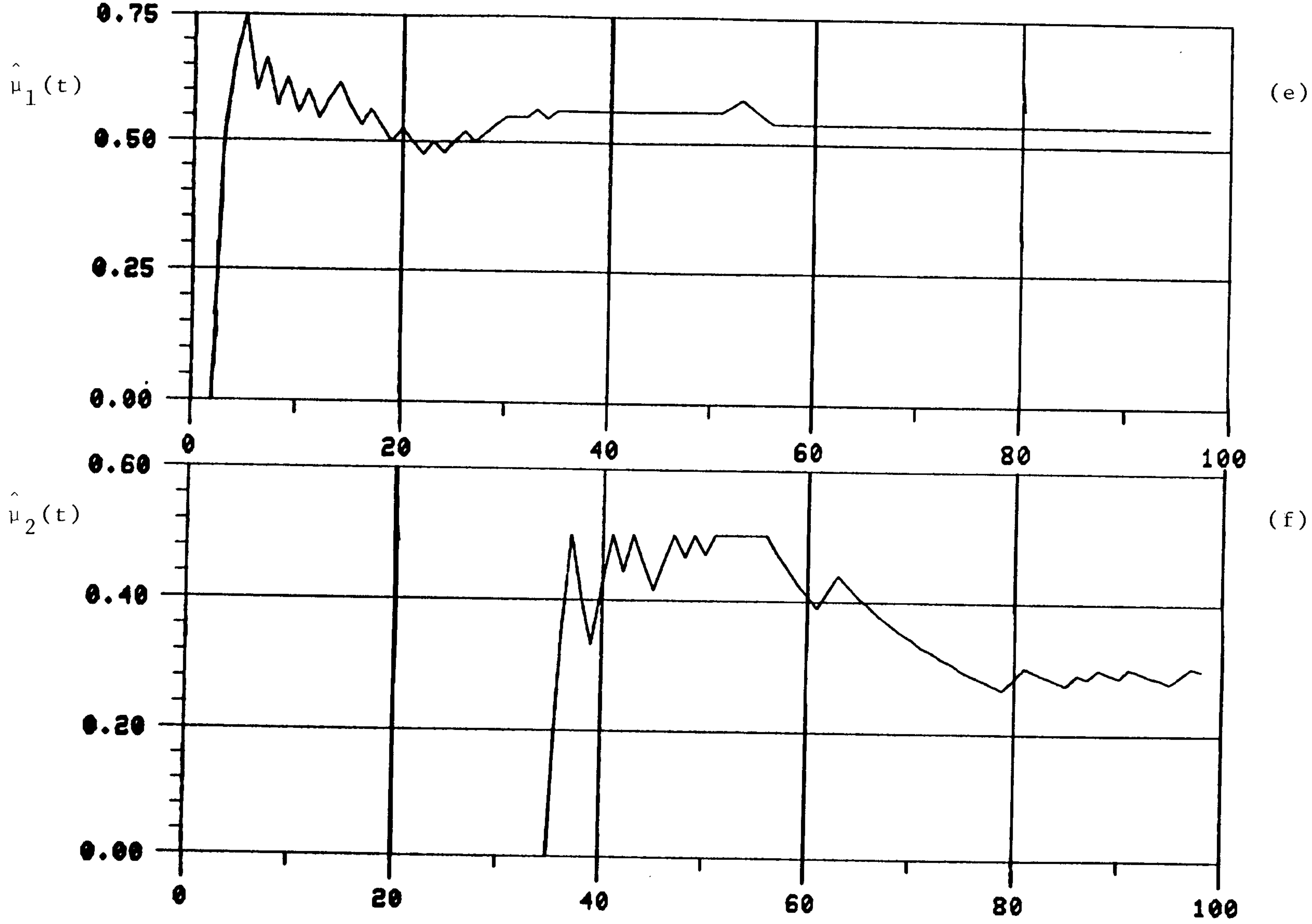


Figure 4.7 Strategy·Adaptive Control (Cont.)

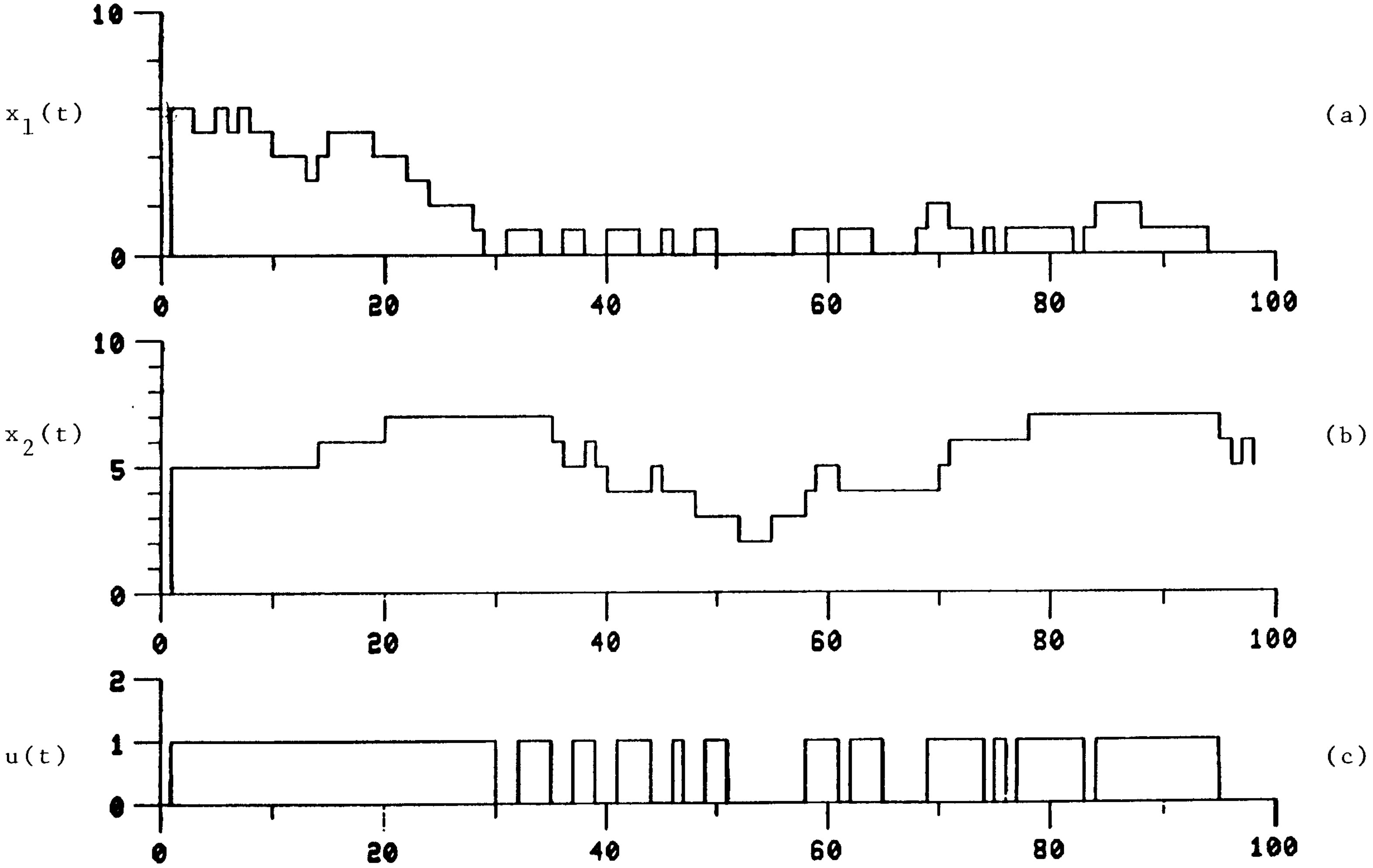


Figure 4.8 Strategy:μc-rule

$\lambda_1 = 0.4,\ c_1 = 1.0,\ \mu_1 = 0.6$

$\lambda_2 = 0.2,\ c_2 = 2.0,\ \mu_2 = 0.2$

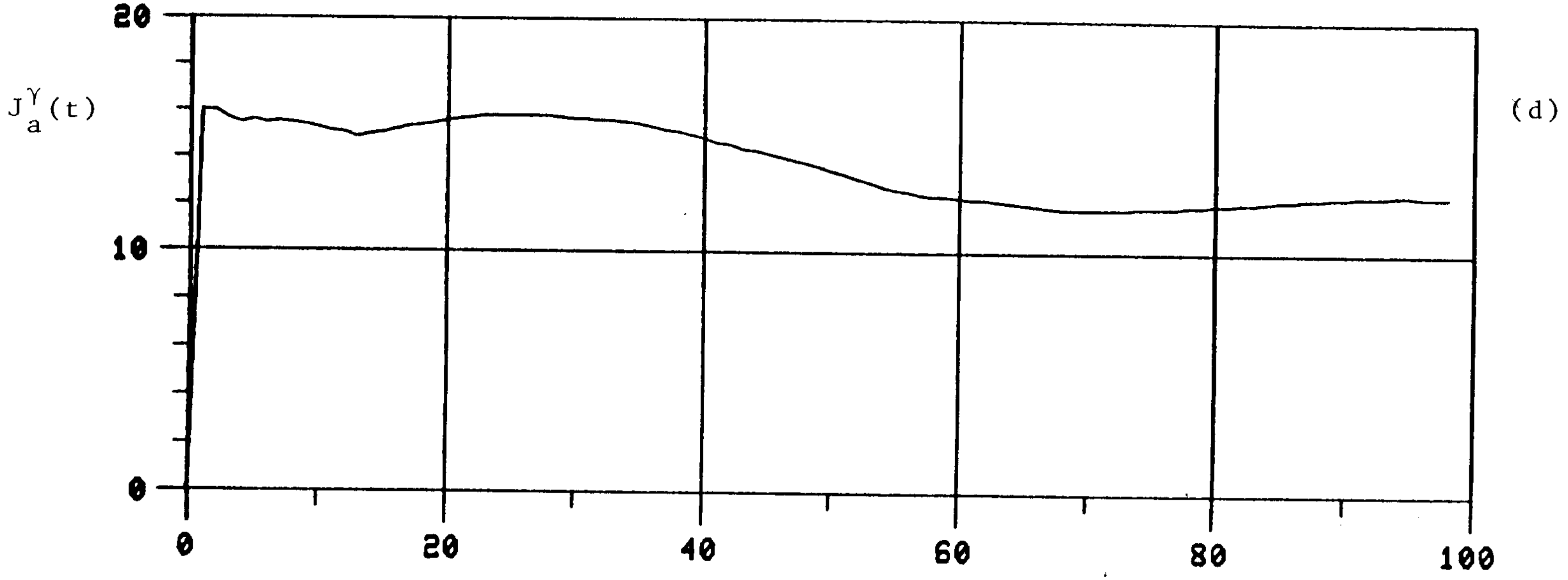


Figure 4.8 Strategy:μc-rule (Cont.)

the adaptive control law both converge to the values of the true system (Figure 4.4(e), (f)).

In Figures 4.6 - 4.8, the system parameters were chosen to move the switch curve further away from the y-axis $\{(7,j);\ j = 1,2,\ldots,7\}$. As a consequence, the queueing system under each strategy performed differently. First, we compare the system under the average cost verses the μc-rule strategies. Under the average cost policy (Figure 4.6), the optimal control provides serve to queue 1 only when queue 2 is empty or queue 1 is near its capacity. The sample path (Figure 4.6(a) - (c)) displays this property. Conversely, the μc-rule services queue 1 until it empties at which time service is provided to queue 2. It is more apparent that the μc-rule is suboptimal when comparing the running average costs of the two systems (Figure 4.6, 4.8(d)).

Now we compare the adaptive control scheme to the optimal average cost. Again the adaptive system has a learning mode which starts with serve to queue 1 and ends after queue 2 is empty (Figure 4.7(a) - (c), (e), (f)). Once the learning mode is completed, the adaptive system follows the average cost optimal system. The running average cost, $J_a^\gamma(\cdot)$ for the adaptive system is higher than the optimal average cost and furthermore is slightly higher than the corresponding μc-rule system (Figure 4.6 - 4.8(d)). The parameter estimate for queue 1 is fairly accurate but the parameter estimate for queue 2 is this sample has not stabilized. Note, the adaptive control strategy does <u>not</u>

converge to the optimal control law. These latter two observations are a consequence of a finite sample path.

5. CONCLUSIONS AND FURTHER RESEARCH

This dissertation has dealt with the priority assignment problem of two queues competing for the service of a single server. Within an optimal control theory framework, we have established for the unbounded, completely observed system that the optimal server time allocation strategy is the μc-rule. Furthermore, we have shown that the optimal value function is convex in its arguments. For the bounded system, we have demonstrated via numerical results that the optimal solution is a true feedback strategy in the sense that it depends on the current queue size and system parameters. For the partially observed system, two different approaches were adopted. First, we considered the systems with known parameters and whose queue size was unobservable. We showed that the one-step, predicted density of the state was a sufficient statistic for control. As a consequence, we obtained an explicit, easily implementable algorithm whose properties were evaluated. Second, the partially observed problem was treated as an adaptive control problem. Here, the parameters were considered unknown constants and the controller observed the past histories of the control and queue size. The analysis lead to a certainty-equivalence, adaptive controller. Provided that a certain sufficient condition on the control value is satisfied, we showed that the adaptive control law and parameter estimator converged to their true values. The interdependencies of the adaptive control law and parameter estimator for a finite capacity system were investigated via simulation.

Our analysis of this simple queueing system leads to several areas for further research. First, for the finite capacity completely observed system, a bound on the cost under the suboptimal μc-rule strategy to the optimal cost is useful. The μc-rule is an elementary control strategy. From a practical standpoint such a bound could justify the additional complexity. Our results provide a framework to compute such bounds numerically; analytical bounds are within reach. Second, the partial observation problem of Chapter 3 retains much of the structure of the complete observation problem. It is conjectured that there exists a simple priority assignment rule for the unbounded, partial observation problem that depends on the costs and on the probability of non-zero queues. An extension of the alternative proof of Theorem 2.3.7 could provide this result. Third, the analysis of the adaptive control scheme requires alternative sufficient condition on the control values. For the two competing queue system, this condition needs to be restated in terms of the parameters $\{\lambda_i, \mu_i; i = 1,2\}$. Extension of the results in Chapter 2 for general arrival processes are discussed in [6].

REFERENCES

[1] T.B. Crabill, D. Gross and M.J. Magazche, "A Classified Bibliography of Research on Optimal Design and Control of Queues," Operations Research, Vol. 25, No. 2, March-April 1977, pp. 219-232.

[2] M.J. Sobel, "Optimal Operation of Queues," in Mathematical Methods in Queueing Theory (A.B. Clarke, Edt.), Lecture Notes in Economics and Math. Systems, 98, 1974, pp. 231-261.

[3] S. Stidham, Jr. and N.V. Prabhu, "Optimal Control of Queueing Systems," Ibid, pp. 263-294.

[4] B. Hajek and T. Van Loon, "Decentralized Control of a Multi-Access Broadcast Channel," IEEE Trans. on Aut. Control, Vol. AC-27, No. 3, June 1982, pp. 559-569.

[5] P. Bremaud, "Optimal Thinning of a Point Process," SIAM J. Control and Optim., Vol. 17, No. 2, March 1979, pp. 222-230.

[6] J.S. Baras, A.J. Dorsey and A.M. Makowski, "Two Competing Queues with Linear Costs: The μc-rule is Often Optimal," submitted to J. Applied Probability, 1983.

[7] W. Lin and P.R. Kumar, "Servers of Different Rates," Proceedings of the Fifth International Conference on Analysis and Optimization of Systems, Versailles, France, December 14-17, 1982, preprint.

[8] Z. Rosberg, P. Varaiya and J. Walrand, "Optimal Control of Service in Tandem Queues," IEEE Trans. on Autom. Control, Vol. AC-27, No. 3, June 1982, pp. 600-610.

[9] D.R. Cox and W.L. Smith, Queues, London: Methuen, 1961.

[10] V.V. Rykov and E. Ye. Lembert, "Optimal Dynamic Priorities in Single Queueing Systems," Eng. Cybern. Vol. 5, No. 1, 1967, pp. 21-30.

[11] J.S. Kakalik, "Optimal Dynamic Operating Policies for a Service Facility", Technical Report 47, Operations Research Center, MIT Cambridge, MA, 1969.

[12] J.M. Harrison, "A Priority Queue with Discounted Linear Costs," Operations Research, Vol. 23, No. 2, March-April 1975, pp. 260-269.

[13] ______________, "Dynamic Scheduling of a Multiclass Queue: Discount Optimality", Operations Research, Vol. 23, No. 2, March-April 1975, pp. 270-282.

[14] V.V. Mova and L.A. Ponomarenko, "On the Optimal Assignment of Priorities Depending on the State of a Servicing System with a Finite Number of Waiting Places," Eng. Cybern., Vol. 12, No. 5, 1974, pp. 66-72.

[15] J.C. Walrand, "Discrete-time Jackson Networks," Proceedings of the 21st IEEE Conference on Decision and Control, Orlando, FL, December 8-10, 1982.

[16] S.A. Lippman, "Semi-Markov Decision Processes with Unbounded Rewards," Managm. Science, Vol. 19, No. 7, March 1973, pp. 717-731.

[17] S. Stidham, "Optimal Control of Arrivals to Queues and Networks of Queues," Presented at the 21st IEEE Conference on Decision and Control, Orlando, FL, Dec. 8-10, 1982, preprint.

[18] P. Bremaud, Point Processes and Queues: Martingale Dynamics, New York: Springer-Verlag, 1980.

[19] D.P. Heyman and M.J. Sobel, Stochastic Models in Operations Research, Vol. I, New York: McGraw-Hill, 1982.

[20] D. Bertsekas, Dynamic Programming and Stochastic Control, New York: Academic Press, 1976.

[21] R.E. Bellman, Introduction to matrix analysis, New York: McGraw Hill, 1960.

[22] S.M. Ross, Applied Probability Models with Optimization Applications, San Francisco, CA: Holden-Day, 1970.

[23] E.V. Denardo, "Contraction Mappings in the Theory Underlying Dynamic Programming," SIAM Rev., Vol. 9, 1967, pp. 165-177.

[24] 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.

[25] 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.

[26] E.J. Sondik, "The Optimal Control of Partially Observable Markov Processes," Ph.D. Dissertation, Department of Engineering-Economic Systems, Stanford University, Stanford, CA, June 1971.

[27] A. Segall, "Dynamic File Assignment in a Computer Network," IEEE Trans. on Autom. Control, Vol. AC-21, No. 2, April 1976, pp. 161-173.

[28] B. Hajek, "Dynamic Decentralized Estimation and Control in Multi-Access Broadcast Channel," *Proc. 19th Decision and Control Conference*, Albuquerque, New Mexico, Dec. 1980, pp. 618-623.

[29] J.S. Baras and A.J. Dorsey, "Stochastic Control of Two Partially Observed Competing Queues," *IEEE Trans. on Autom. Control*, Vol. AC-26, No. 5, Oct. 1981, pp. 1106-1117.

[30] W.M. Wonham, "Stochastic Problems in Optimal Control," in *Conv. Rec.*, 1963, *Int. Conf.*, Part II, pp. 114-124.

[31] H. Kushner, *Introduction to Stochastic Control*, Holt, Rinehart and Winston, New York, 1971.

[32] K.J. Astrom, *Introduction to Stochastic Control Theory*, New York: Academic Press, 1970.

[33] J.S. Baras, W.S. Levine and T.L. Lin, "Discrete-Time Point Processes in Urban Traffic Queue Estimation," IEEE Trans. on Autom. Control, Vol. AC-24, No. 1, Feb. 1979, pp. 12-27.

[34] K. Astrom, "Optimal Control of Markov Processes with Incomplete State Information I," J. Math. Anal. Appl., Vol. 10, 1965, pp. 174-205.

[35] K. Astrom, "Optimal Control of Markov Processes with Incomplete State Information II: Convexity of the Loss Function", J. Math. Anal. Appl., Vol. 26, 1969, pp. 403-406.

[36] C. Striebel, "Sufficient Statistics in the Optimal Control of Stochastic Systems," J. Math. Anal. Appl., Vol. 12, 1965, pp. 576-592.

[37] W.M. Wonham, "On the Separation of Stochastic Control," SIAM J. Control, Vol. 6, No. 2, 1968, pp. 312-326.

[38] A Segall, "Optimal Control of Noisy Finite-State Markov Processes," IEEE Trans. on Autom. Control, Vol. AC-22, No. 2, April 1977, pp. 179-186.

[39] J.W. Patchell and O.L.R. Jacobs, " Separability, Neutrality and Certainty Equivalence, Int. J. Control, Vol. 13, 1971, pp. 337-340.

[40] J.R. Jackson, "Networks of Waiting Lines," *Operations Res*. Vol. 5, 1957, pp. 518-521.

[41] P. Varaiya, "Notes on Stochastic Control," Unpublished Class Notes, University of California, Berkeley, Department of Electrical Engineering and Computer Sciences.

[42] J.S. Baras, W.S. Levine, A.J. Dorsey, and T.L. Lin "Advanced filtering and prediction software for urban traffic control systems," Department of Transportation, Contract DOT-05-60134, Final Report No. DOT/RSPA/DPB-50/80/17, July 1980.

[43] J.S. Baras, W.S. Levine, and A.J. Dorsey, "Estimation of traffic platoon structure from headway statistics," IEEE Trans. Autom. Control, Vol. AC-24, No. 4, pp. 553-559, August 1979.

[44] A.J. Dorsey, J.S. Baras, and W.S. Levine, "Point process disorder problem and its application to urban traffic estimation, Conference of Info. Science and Systems, John Hopkins University, pp. 37-42, March 1979.

[45] D. Gazis (Ed.), Traffic Science, New York: John Wiley, 1974.

[46] M. Rudemo, "State estimation for partially observed Markov chains," J. Math. Anal. Appl., Vol. 44, pp. 581-611, 1973.

[47] R. Kalman, P. Falb and M. Arbib, Topics in Mathematical System Theory, McGraw-Hill: New York, 1969.

[48] R.W. Brockett and J.M.C. Clark "Geometry of the Conditional Density Equation," Proc. Int. Conf. on An. and Opt. of Stoch. Syst., Oxford, England 1978.

[49] S.K. Mitter "On the Analogy Between Mathematical Problems of Non-linear Filtering and Quantum Physics," to appear in Richerche di Automatica, 1980.

[50] I.D. Landau, "A Survey of Model Reference Adaptive Techniques-Theory and Applications," Automatica, Vol. 10, 1974, pp. 353-370.

[51] B. Wittenmark, "Stochastic Adaptive Control Methods: A Survey," Int. J. Control, Vol. 21, 1975, pp. 705-734.

[52] P. Mandl, "Estimation and Control in Markov Chains," Adv. Appl. Prob., Vol. 6, 1974, pp. 40-60.

[53] V. Borkar and P. Varaiya, "Adaptive Control of Markov Chains I: Finite Parameter Set," IEEE Trans. on Autom. Control, Vol. AC-24, No. 6, December 1979, pp. 953-958.

[54] V. Borkar and P. Varaiya, "Identification and Adaptive Control of Markov Chains," SIAM J. Control and Optim., Vol. 20, No. 4, July, 1982, pp. 470-489.

[55] B. Doshi and S.E. Shreve, "Strong Consistence of a Modified Maximum Likelihood Estimator for Controlled Markov Chains, J. Appl. Prob., Vol. 17, 1980, pp. 726-734.

[56] P.R. Kumar and A. Becker, "A New Family of Optimal Adaptive Controllers for Markov Chains," IEEE Trans. on Autom. Control, Vol. AC-27, February 1982, pp. 137-146.

[57] P.R. Kumar and W. Lin, "Optimal Adaptive Controllers for Unknown Markov Chains," IEEE Trans. on Autom. Control, Vol. AC-27, Aug. 1982, pp. 765-774.

[58] P.R. Kumar, "Simultaneous Identification and Adaptive Control of Unknown Systems over Finite Parameter Sets," IEEE Trans. on Autom. Control, Vol. AC-28, January 1983, pp. 68-76.

[59] B. Sagalovsky, "Adaptive Control and Parameter Estimation in Markov Chains: A Linear Case," IEEE Trans. on Autom. Control, Vol. AC-27, April, 1982, pp. 414-419.

[60] Y.M. El-Fattah, "Gradient Approach for Recursive Estimation and Control in Finite Markov Chains," Advanced Appl. Prob., Vol. 13, 1981, pp. 778-803.

[61] O. Hernandez-Lerma and S.I. Marcus, "Optimal Adaptive Control of Priority Assignment in Queueing Systems, submitted to IEEE Trans. on Autom. Control.

[62] Y. Baram and N.R. Sandell, "Consistent Estimation on Finite Parameter Sets with Application to Linear System Identification," IEEE Trans. on Autom. Control, Vol. AC-23, June, 1978, pp. 451-454.

[63] P.R. Kumar, "Adaptive Control with a Compact Parameter Set," SIAM J. Control and Optim., Vol. 20, No. 1, January, 1982, pp. 9-13.

[64] H.L. Royden, Real Analysis, 2nd edition, the Macmillan Co., New York, 1968.

[65] D.R. Robinson, "Optimality Conditions for a Markov Decision Chain with Unbounded Costs," J. Appl. Prob., Vol. 17, 1980, pp. 996-1003.

[66] A. Hordijk, Dynamic Programming and Markov Potential Theory, Mathematical Centre Tracts, No. 51, Amsterdam.

[67] P. Wolfe and G.B. Dantzig, "Linear Programming in a Markov chain," Oper. Res., Vol. 10, 1962, pp. 702-710.

[68] R.A. Howard, Dynamic Programming and Markov Processes, Technology Press of M.I.T., Cambridge, MA, 1960.