Control of Manufacturing Systems with Delay

by

Garrett John Van Ryzin

B.S. Elec. Eng., Columbia University
(1985)

SUBMITTED TO THE DEPARTMENT OF
ELECTRICAL ENGINEERING
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
May, 1987

Signature of Author ....
Dept. of Electrical Engineering

Certified by.......... ....
Stanley B. Gershwin
Principal Research Scientist
Laboratory for Information and Decision Systems

Accepted by..
Prof. A. C. Smith
Chairman, Department Commitee

Control of Manufacturing Systems with Delay

by

Garrett John Van Ryzin

Submitted to the Department of Electrical Engineering
on May 9, 1987 in partial fulfillment of the
requirements for the Degree of Master of Science in
Electrical Engineering

ABSTRACT

Continuous flow formulations of manufacturing systems offer an appealing way to reduce the complexity inherent in traditional discrete parts modeling. However, many manufacturing processes have significant delays in the material flow and this phenomenon is absent in existing continuous formulations. Delay is also present in large aggregations of machines that process many parts at a time. A linear system approximation technique is proposed for dealing with such. Both analytical and practical results are derived. Heuristic, sub-optimal control laws are constructed from the analytical results by numerical experimentation. Features of the control for interconnected manufacturing systems are also explored and a simple heuristic for two machines in series is developed.

Thesis Supervisor: Dr. Stanley B. Gershwin

Title: Principal Research Scientist,
Laboratory for Information and Decision Systems

Lecturer, Department of Electrical Engineering and
Computer Science

ACKNOWLEDGMENTS

I would like to thank my advisor, Stan Gershwin, for his invaluable help and guidance during my time at MIT. There was much I learned from him about research practice, philosophy and presentation that I was completely oblivious to when I began my studies. He pointed out my false starts and oversights, without losing enthusiasm for the ideas we developed, and was always quite willing to talk about ideas, classes, my career and life in general. I also appreciate his patience and friendship during the difficult time of my father's illness and death.

I would also like to thank Sheldon Lou for the many stimulating discussions we have had. He injected much rigor into the ideas presented in this thesis and his many contributions are greatly appreciated. In particular, the proof in Appendix A contains many details developed or suggested by Sheldon.

I also thank my office mates Yong Choong, Therry LaMote and Camille Libosvar for the office squash matches, coffee and late night chinese dinners. Working with a great and famous researcher like Yong was both an honor and a privilege. Also, I discovered that French people are really not that different from the rest of us after all.

I would like to thank AT&T Bell Labs, especially my manager H. G. Vaidya and department head, J. Fehlauer, for their patience and support.

Finally, I would like to thank my wife, Mary Beth, for her constant love and support during both the joyful and sorrowful times of these past few years.

This thesis is dedicated to my loving father, John Van Ryzin, who died on March 6, 1987, while I was completing this thesis.

TABLE OF CONTENTS

CHAPTER 1 - THE MANUFACTURING ENVIRONMENT AND DELAY

1.1 INTRODUCTION

The problems in American manufacturing are increasingly the focus of public attention. We are now beginning to recognize that our ability to compete in world markets, to support sustained economic growth, and to realize an improving standard of living depends on our ability to produce products efficiently. Automation is seen by many as a solution for the problems our manufacturers face. The conventional wisdom holds that if enough robots, numerically controlled machine tools, radio controlled materials handling carts, machine vision systems, etc. are installed in factories, then our manufacturing woes will disappear. Conspicuously absent in this vision, however, is sufficient wisdom on how we will effectively control these enormously complex and expensive "factories of the future".

Recent experience (Nag 1986) suggest that automation often introduces as many problems as it solves. Managers are faced with the task of directing an extremely complex system that increasingly is being stripped of human operators - operators whose adeptness at reacting to the change of events on the factory floor is not easily duplicated in control algorithms. It appears that if we are truly going to benefit from the application of this new technology, a more rational understanding of factory behavior and control is needed.

The ultimate research goal in manufacturing systems is to develop models and techniques for analyzing a general manufacturing process. What we are presenting in this thesis addresses two features that are common in many manufacturing systems, namely delay and interconnected machines. We hope that our results and observations represent a step forward in the understanding of more general systems.

1.2 THE MANUFACTURING ENVIRONMENT

Before we can address the issue of control in an automated manufacturing environment, a more detailed discussion of the qualitative features of these systems and the problems inherent in their control is in order. In this section we provide a brief overview of the nature of the automated manufacturing problem and discuss some of the problems a controller must contend with.

Perhaps the most significant quality the new manufacturing technologies posses is flexibility. A machine tool that once only performed a single, restricted operation (e.g. drilling or cutting) is replaced by a numerically controlled machine capable of performing a complex sequence of operations, a sequence that may be changed by simply providing the machine with new data. A conveyer belt is replaced by an automated materials handling system that can redirect the flow of parts between machines and workstations on demand. An automated inspection station can dynamically examine parts for defects so that parts may be sent back for rework or discarded (Gershwin *et al.* 1986).

Flexibility, however, increases the complexity of the decisions that must be made on the shop floor. If this increased complexity can be successfully managed, the potential benefits of automation can be significant. But, flexibility also creates the potential for confusion and the misuse of resources.

The situation is made yet more complex by the presence of randomness in the form of machine failures and part defects and by the presence of delays, which, as we shall see later, are inherent in many manufacturing systems.

Manufacturing systems are often very large scale systems, which makes global control difficult. Some sort of decomposition is necessary to generate problem formulations that are tractable. One possible approach that seems particularly appropriate for manufacturing systems is to decompose the problems hierarchically by time scales. This approach is a common technique for treating large-scale traditional control problems and has the added appeal of paralleling the organizational structures found in most factories.

For example, a machine operator is concerned with the loading of individual parts, his supervisor is concerned with the daily scheduling of a whole line or work center and the plant manager is responsible for the weekly operation of the whole factory. Each is concerned with problems that span a different time horizon and scope, yet they must communicate and coordinate their individual decisions in such a way that the overall objectives of the factory are met without overburdening each other with information or responsibilities.

Gershwin @et al. (1986) has suggested a methodology for defining a hierarchy in which modules in the hierarchy pass information to lower levels in the form of requirements (e.g. production rates, set-up frequencies) and pass information to levels above them in the form of constraints (e.g. capacity, machine states). Each module then solves an optimization problem based on these constraints in order to generate the requirements for the next lower level. The control problem is thus reduced to a collection of smaller optimization problems which are analytically tractable.

1.3 PREVIOUS WORK

The work we propose falls within the general study of flexible manufacturing systems (FMS). Since we are primarily concerned with control issues involving delay, we will also review the optimal control literature concerning delay.

1.3.1 FMS Literature Survey

The rise of flexible manufacturing systems has paralleled the rise of microprocessors. With computing power no longer the limitation it was in the past, the real-time control of whole factories is now possible. This control capability has awakened an interest within the operations research and control theory communities in devising algorithms to optimally control flexible manufacturing processes. Reviews of progress in this area are given in Buzacott (1982), Buzacott and Yao (1982) and Gershwin *et al.* (1986).

Some of the research to date has addressed the problem of optimal routing in simple networks of queues. Examples of this work include Olsder and Suri (1980), Seidman and Schweitzer (1982), Tsitsiklis (1980), Ephremides *et al.* (1980), Maimon and Gershwin (1986) and Hahne (1981).

Stidham (1985) provides a good review of research progress in the area of controlled queues. It appears, however, that optimal controls for networks of queues are often difficult to derive even for the simplest systems possible and computational techniques appear to be limited (Hahne 1981). The fundamental difficulty in modeling manufacturing systems as networks of queues is the large state space required to represent such a system.

One technique for avoiding the dimensionality problem is to represent the material flow in a factory as a continuous flow. Kimemia (1982) and Kimemia and Gershwin (1983) proposed a hierarchical control system where the top levels use a model in which the flow of material between workstations is continuous. The discrete dispatching of parts is left to lower level work station controllers. The flow level control accounts for changes in the state of workstations (i.e. machine failures) and observes capacity constraints. Using the methods of Rishel (1975), they derive a control law based on the gradient of the optimal value function. However, it is only possible to compute the optimal value function for very simple systems (cf. Akella and Kumar (1986) and Bielecki and Kumar (1986)). Using quadratic approximations to the optimal value function, Akella _et al_ (1984) developed sub-optimal strategies that have proved very effective in handling simulated systems of realistic size.

One key assumption made in most continuous flow models is that the time that parts spend in a workstation is negligible compared to the rate of flow. This limits the applicability of the model for, as we show in Chapter 3, delay through a workstation is often unavoidable.

1.3.2 Delay Literature

Since continuous flow manufacturing models are a recent development, delay issues have not yet been addressed. However, theoretical and practical techniques for accommodating delay have been addressed for traditional control problems.

Donoghue (1977) gives a good comparison between the Smith predictor, a classical block diagram approach to handling delays (Smith, 1957), and modern optimal control results for linear quadratic (LQ) systems with delays in the control. He demonstrates that even though the optimal control approach can be used to generate an optimal controller, a predictor is often easier to implement. It is also intuitively easier to interpret and adjust. However, the uncertainty and constraints involved in complex manufacturing systems makes it difficult to construct a predictor.

Mariani and Nicolleti (1973) address a general discrete time optimal control problem with delays in the dynamics and controls. They derive a discrete maximum principle for a deterministic system with convex costs, linear dynamics and convex constraints. Their results are primarily useful for static, very large scale non-linear programming approaches. Such an approach would involve discretizing time over a finite horizon and then optimizing over the states at all time steps simultaneously. However, the dimensionality of a typical manufacturing system would make this type of approach computationally infeasible.

Ross and Flugge-Lotz (1969) derive the optimal control for a deterministic LQ problem with delay and show that the gain matrix contains a term which is an integral of product of the past states and a matrix function. This matrix function is, however, difficult to compute.

Koivo (1969) shows that for similar LQ delay systems with white Gaussian noise in the process and measurement (LQG), certainty equivalence holds. Certainty equivalence says that using the expected value of the state in the deterministic optimal control law for the same system without noise is optimal. The separation of estimation and control is thus preserved for LQG systems with delay.

Hess (1972) and Hess and Hyde (1973) use an approximation technique due to Repin (1965) to estimate the matrix function of Ross and Flugge-Lotz. Though the problem they address is a LQ problem with delays in the state and no randomness, their approximation technique forms the basis for our analysis.

These traditional control problems give us valuable techniques for accommodating delay. However, the results are, by themselves, not directly applicable to our problem because the LQ model is not appropriate for manufacturing systems.

1.4 PREVIEW AND OUTLINE OF THESIS

In Chapter 2, we present the continuous flow FMS model of Kimemia and Gershwin. Kimemia and Gershwin made several important assumptions in developing their continuous flow model:

1) The setup time is much less than the processing time of parts.

2) The operation times are much less than the mean time time to repair (MTTR) and mean time between failure (MTBF) of the machines.

3) The MTTR and MTBF of machines is much less than the planning horizon.

A consequence of these assumptions is that the material movement can be modeled as a continuous flow between machine failures and repairs. This flow can then be described by an ordinary differential equation. However, as we show in Chapter 3, systems that operate on many parts at a time require a delay differential equation to describe the material flow.

Chapter 3 contains a simple analysis of the source of delay in continuous flow models and the effect of delay on internal inventories. The FMS control problem for simple delay systems is then formulated.

In Chapter 4, we present a strategy for approximating delay by a finite-dimensional system of first order equations without delay following the work of Hess and Repin. The approximated system is then analyzed using the Kimemia-Gershwin results. The approximation can be made arbitrarily good by increasing the dimensionality of the approximate system to arrive at an optimal control for the original system.

While this approach yields the optimal control in theory, the lack of methods for computing the optimal value function for a non-delay system means that a direct application of the approximation technique is still not possible. However, using quadratic approximations to the value function we arrive at a particularly simple and interesting form for the control law involving a linear term in the current inventory level and a "correlation" or "convolution" type term in the past controls. Simulation results based on this control law are also presented.

Another assumption in the Kimemia-Gershwin model is that the instantaneous capacity not be violated. The instantaneous capacity is a set that describes the mix of parts that can be produced at any given time. If storage is allowed in a FMS, however, it is possible to begin production on parts even though some machines needed for part production are not operational. Increased capacity can thus be realized if a FMS is subdivided and internal storage is allowed.

In Chapter 5, the consequences of the instantaneous capacity restriction are discussed and a surplus space representation of interconnected systems is introduced. The surplus is defined as the difference between what has been produced by a machine and the cumulative demand at a given time.

Numerical optimal controls are generated for a simple interconnected system and then graphed in surplus space. The data suggest that the optimal control can be characterized by two regions. In one region the control is approximately a surplus hedging point policy, while in the other region, the control is approximately a buffer hedging point policy. A heuristic strategy based on this observation, called a two-point control, is introduced. A simple technique for generating the parameters of a two-point control is developed.

Chapter 6 contains a summary of the results discussed in the previous chapters and suggests topics for further research.

CHAPTER 2 - THE KIMEMIA-GERSHWIN MODEL

In this chapter we review the results of Kimemia and Gershwin (1983) for a continuous flow model of an FMS. The remaining chapters require a great degree of familiarity with many features of this basic model.

2.1 PROBLEM FORMULATION

The manufacturing system we consider is represented in Figure 2.1. The FMS consists of a collection of individual workstations and processes a given family of parts. Each workstation, in turn, consists of a set of identical machines. The supply of material at the input to the FMS is assumed infinite and the demand is known. Machines are subject to random failures.

There are several important assumptions that are made in this model concerning the relative frequency of events. These time scale assumptions are that 1) the time parts spend in the system is negligible, 2) many operations take place between machine failures and repairs so that a flow rate can be meaningfully defined in these intervals, and 3) many machine failures and repairs occur during the planning horizon so that an infinite horizon criterion is appropriate.

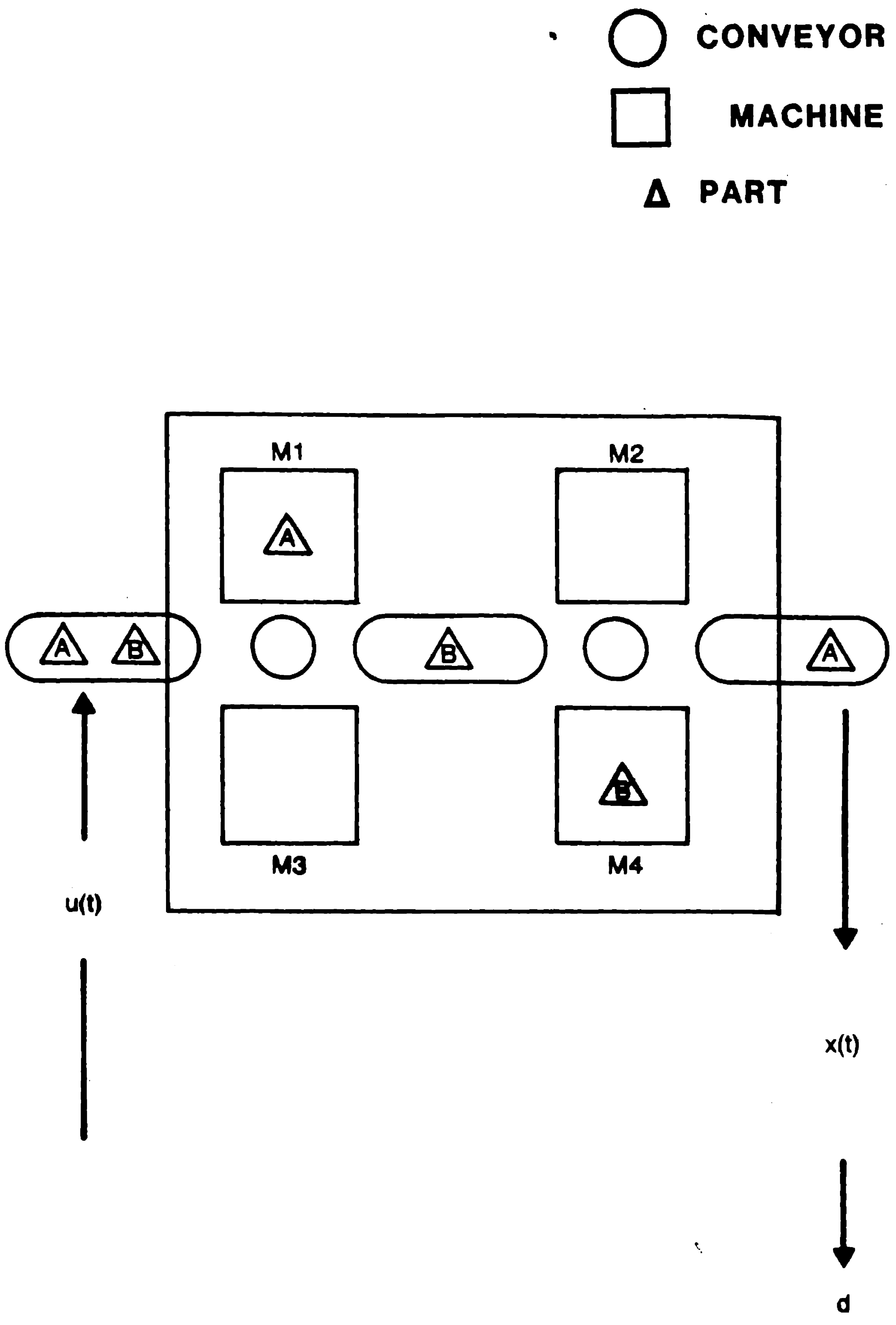


FIGURE 2.1 BASIC FMS MODEL

Some of these time scale assumptions can be relaxed, and this is discussed further in Chapters 3 and 5. For now, we will assume they always hold and proceed with the development of the basic model.

Control of the system is organized hierarchically as shown in Figure 2.2. At the top level, long-term production rates for each part type are determined based on estimates of demand and the steady state system capacity. These production rates are passed down to the flow control level, which calculates short term production rates based on the instantaneous capacity of the FMS. The flow control level tries to keep production close to the long term rate without creating excess inventory.

The lowest level is the dispatch level. The dispatch level determines when to load individual parts based on the short-term rates set by the flow control level.

Our focus here and throughout this thesis is on the flow control level. We now give a precise mathematical description of the FMS model. The workstations of the FMS are denoted $m=1,2,....M$ and the part types are denoted $n=1,2,...N$. Each workstation has L_m identical machines.

Let u(t) be the N-vector of production rates and d be the N vector of demand rates. The total number of finished parts produced is measured against the long-term production demand, d,

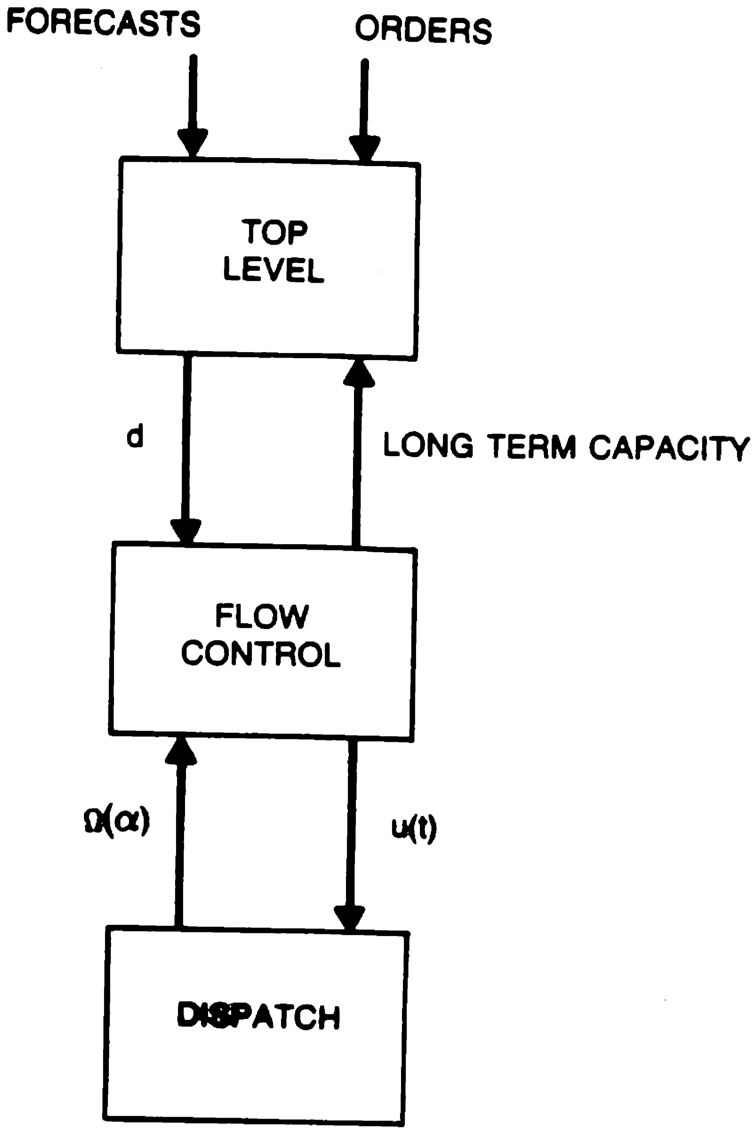


FIGURE 2.2 HIERARCHICAL ORGANIZATION OF FMS CONTROL

by forming a production surplus measure, x(t). Negative values of x(t) represent a backlog of parts. The N-vector x(t) satisfies

$$\dot{x}(t) = u(t) - d \tag{2.1}$$

The quantity x(t) is simply a measure of how far ahead or behind the FMS is relative to the target production rate d. It is an accounting quantity and does not measure the amount of material anywhere on the factory floor. The objective is to keep x(t) close to zero by penalizing the FMS for being to far ahead or too far behind the target production rate, d.

If multiple FMS's are connected together and each follows such a policy, it is reasonable to expect that, in the long run, no single FMS will get too far ahead or too far behind the others. We explore this issue in greater detail in Chapter 5 where we discuss inter-connected systems.

The repair state of a FMS is denoted by an M-tuple of integer variables

$$\alpha(t) = (\ \alpha_1(t),\ \alpha_2(t),\ \ldots\ ,\ \alpha_M(t)\ ) \tag{2.2}$$

where

$$\alpha_m(t) = k \quad \text{if } k \text{ machines at workstation } m \text{ are operational at time } t. \tag{2.3}$$

Machines are assumed to have independent, exponentially distributed failures and repairs. Therefore, the rate at which machines are repaired in a workstation is proportional to the number of broken machines and the rate at which machines fail is proportional to the number of operational machines. The dynamics of α are thus given by:

$$P[\ \alpha_m(t+\delta t) = k-1 \mid \alpha_m(t) = k\ ] = kp_m\delta t \tag{2.4}$$

$$P[\ \alpha_m(t+\delta t) = k+1 \mid \alpha_m(t) = k\ ] = (L_m-k)r_m\delta t \tag{2.5}$$

For any two states i and j we define

$$\lambda_{ij}\delta t = P[\ \alpha(t+\delta t) = j \mid \alpha(t) = i\ ] \text{ for } i \neq j \tag{2.6}$$

$$\lambda_{ii} = -\sum_{j}\lambda_{ij} \tag{2.7}$$

The production rates are constrained by the current capacity of the system. We define

$$\tau_{nm} = \text{Processing time of type n parts on machine m.} \tag{2.8}$$

$$u_n(t) = \text{Production rate of type n parts.} \tag{2.9}$$

We now make the important assumption that no material is allowed to accumulate in the system. Only production rates that can be satisfied at every workstation in the system are allowed. The quantity $\sum_n \tau_{nm} u_n(t)$ has units of material (product of production rate and time) and represents the "quantity" of parts being processed at time t. Note that in using the continuous model, we must speak of a "quantity" of parts that can have a continuum of values rather than a "number" of parts that has discrete values. The number of available machines at time t is $\alpha_m(t)$. Since we can not have have more material being processed than the number of available machines, we must have

$$\sum_n \tau_{nm} u_n(t) \le \alpha_m(t) \quad \text{for all } m \tag{2.10}$$

The instantaneous capacity set is therefore defined by

$$\Omega(\alpha) = \{ u \mid u \ge 0, u \text{ satisfies } 2.10 \} \tag{2.11}$$

The instantaneous capacity set is a polyhedral in u. It defines the set of production rate vectors, u, that can be achieved given the current state of the workstation. Note that by constraining u(t) in this way we do not allow a production rate at the input to the FMS that cannot be satisfied at every workstation in the system. We examine the consequences of this restriction in Chapter 5.

We are now ready to formulate an optimization problem for the the flow control level. Given an initial production state x and machine state α, we would like to minimize the cost

$$J(x,\alpha) = \lim_{T\to\infty} \frac{1}{T} E\{ \int_0^T g(x(t))\, dt \mid x(0) = x , \alpha(0) = \alpha \} \tag{2.12}$$

where g(x) penalizes each component of x for being too positive or too negative. In particular we require

$$g(x) = \sum_n g_n(x_n) \tag{2.13}$$

$$\lim_{|x|\to\infty} g_n(x) \to \infty \tag{2.14}$$

and

$$g_n(0) = 0 \tag{2.15}$$

We now look for control laws, or policies, that minimize the cost, (2.12), and satisfy the capacity (2.11). These policies will be functions of the current buffer level and the current machine state. We define the cost-to-go from a state x and α using the policy u by

$$J_u(x,\alpha) = \lim_{T\to\infty} \frac{1}{T} E\{ \int_0^T g(x(t))\, dt \mid x(0) = x \,,\, \alpha(0) = \alpha \} \tag{2.16}$$

It is shown in Kimemia (1982) using the results of Rishel (1975) that the optimal policy, $u^*(x,\alpha)$, and optimal cost-to-go, $J_u^*(x,u)$, satisfy

$$0 = \min_{u\in\Omega(\alpha)} \{ g(x) + \nabla_x J_u^* \, \dot{x} + \sum_\beta \lambda_{\alpha\beta} \, J_u^*[x,\beta] \} \tag{2.17}$$

If we substitute the state equation

$$\dot{x} = u - d \tag{2.18}$$

into (2.17) we get

$$0 = \min_{u\in\Omega(\alpha)} \{ g(x) + \nabla_x J_u^* \, (u-d) + \sum_\beta \lambda_{\alpha\beta} \, J_u^*[x,\beta] \} \tag{2.19}$$

The optimal control, $u^*(x,\alpha)$, in this equation is determined by

$$\min_{u\in\Omega(\alpha)} \nabla_x J_u^* \, u \tag{2.20}$$

Equation (2.20) is the main result of Kimemia and Gershwin. It shows that the optimal control must satisfy a linear program at every time instant. (Recall that the capacity set, $\Omega(\alpha)$, is polyhedral.)

As x(t) changes the coefficients of the linear program (2.20) will change. The value function thus divides the x-space into regions where the control takes on a particular extreme point value of $\Omega(\alpha)$. For example, Figure 2.3 shows the set Ω and the regions of x-space for a single workstation producing two types of parts, Type 1 and Type 2 when all machines are operational.

For a some x and α states, the solutions to the linear program (2.20) will be at extreme points of Ω, which corresponds to an interior region of x-space as shown in Figure 2.3. If a machine state has sufficient capacity to meet demand, there exists a point in x-space, called a <u>hedging point</u>, where $\nabla J_x(x)=0$ or, if the gradient does not exist, where the minimization is not determinable. This is the point labeled x^* in Figure 2.3.

If the system reaches the hedging point, the optimal policy is to set u(t)=d and let the system remain there until the machine state changes. Gershwin, Akella and Choong (1985) discuss this behavior in detail. They also discuss the behavior of the control on the boundaries between regions.

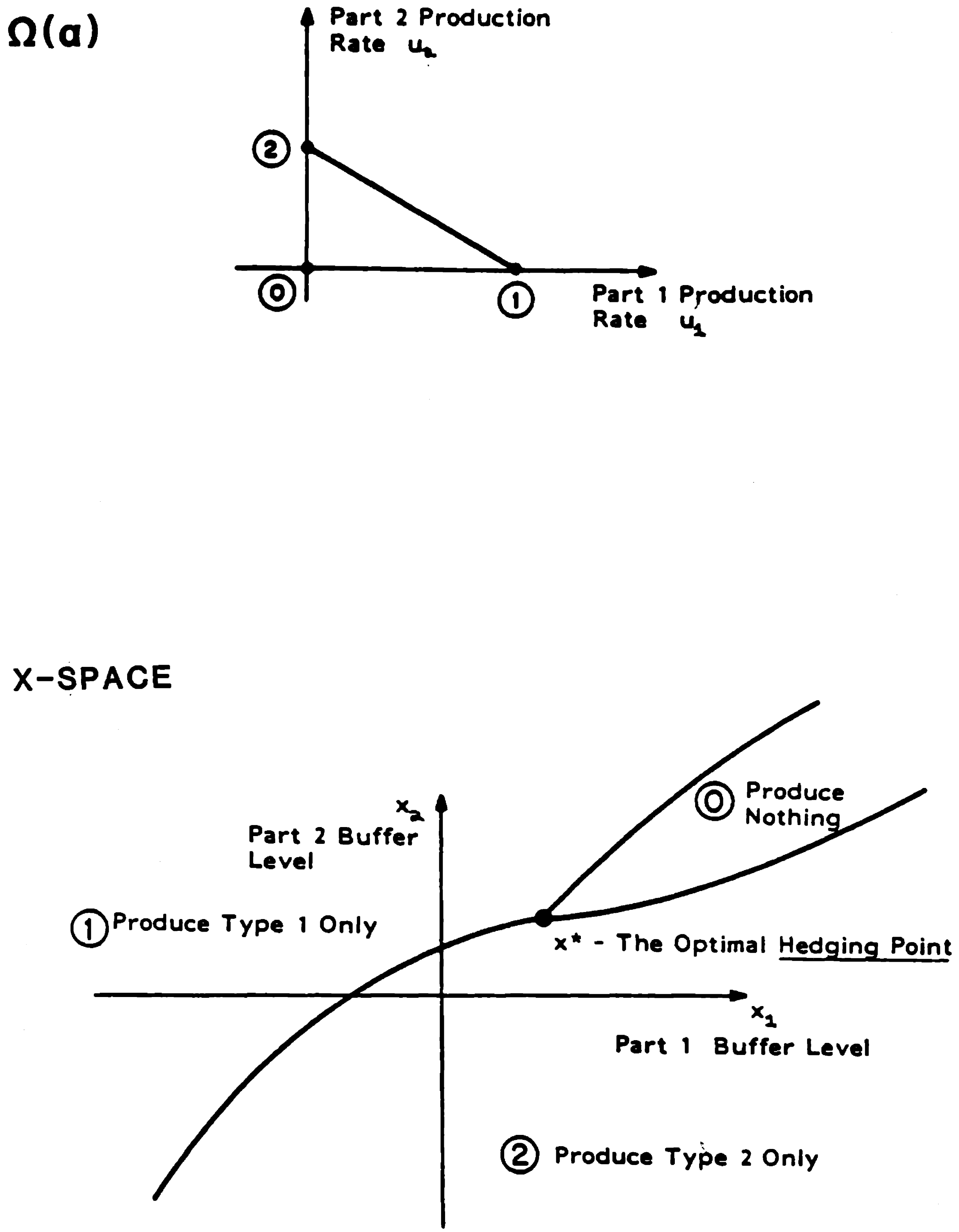


FIGURE 2.3 CONSTRAINT SET AND X-SPACE

Some boundaries are attractive, meaning the control regions adjacent to the boundary drives x(t) back toward the boundary. This can result in a back-and-forth, or chattering, type of behavior. On attractive boundaries, the optimal control is to follow the boundary. Other boundaries are repulsive and the control simply changes extreme points as these boundaries are crossed.

Unfortunately, we need to know the gradient of the optimal value function to use (2.20). In general it is difficult to compute the optimal value function for problems of realistic dimension and approximations are needed in practical applications (see Gershwin, Akella and Choong (1985) and Kimemia(1982)).

The optimal value function is shown in Kimemia (1982) to be convex and continuous. To approximate the convexity of the value function, Kimemia and Gershwin proposed a quadratic approximation to the value function. They showed that such an approximation yielded very good system behavior. When actually applying the control, the exact value function seemed to matter less than its gross characteristics. In general, the experience of Gershwin and his co-workers seems to show that the behavior of the system is relatively insensitive to changes in value function parameters.

Several features of this optimal control derivation are needed in Chapter 3 when we introduce delay. The first is that the optimal control is determined from (2.17) by substituting a differential equation for x(t). Also, only a differential equation that is linear in u(t) yields the linear program (2.20). Having a differential equation representation of the evolution of the system state is necessary if we are to apply this same analysis to other systems. This is an important consideration for systems with delay, because the evolution of the state of a delay systems is not trivially described using simple differential equations.

2.2 EXTENDING THE BASIC MODEL

The Kimemia-Gershwin control has proved very effective in detailed simulations of actual manufacturing systems (Akella, Choong and Gershwin 1984). It turns out, however, that some of the assumptions made in developing the model, namely the assumption that the processing time is small and the assumption that the instantaneous capacity be strictly satisfied, limit the application of the control law.

These assumptions are true for an FMS with a small number of machines (typically less than ten machines). To extend the model to larger systems, these two assumptions must be reexamined. Extending this model and its solution to larger systems is the main objective of the remaining chapters of this thesis.

CHAPTER 3 - THE FMS MODEL WITH DELAY

In this chapter we examine the effects of adding delay to the FMS model of Chapter 2. A simple analysis shows that an FMS that works on a large number of parts should be modeled with a process delay. We examine the effects of delay within work-stations and on the FMS as a whole.

Delay in individual workstations can be tolerated within the framework of the basic FMS model. However, doing so requires that internal inventories be maintained between workstations to keep the workstations supplied during changes in the production rate. These inventories are proportional to the delay so for large delays, the inventory levels will be excessive. Subdividing the control of the FMS may therefore be necessary to reduce the level of work in progress (WIP).

A process delay in an isolated FMS with constant demand is a relatively simple extension of the basic model. Delay in an interconnected FMS is less straightforward.

3.1 FUNDAMENTAL SOURCES OF DELAY IN MANUFACTURING SYSTEMS

To understand why delay is an important consideration in a continuous flow model, the basic assumptions behind a flow model must be reexamined. Figure 3.1 shows a simple workstation. This

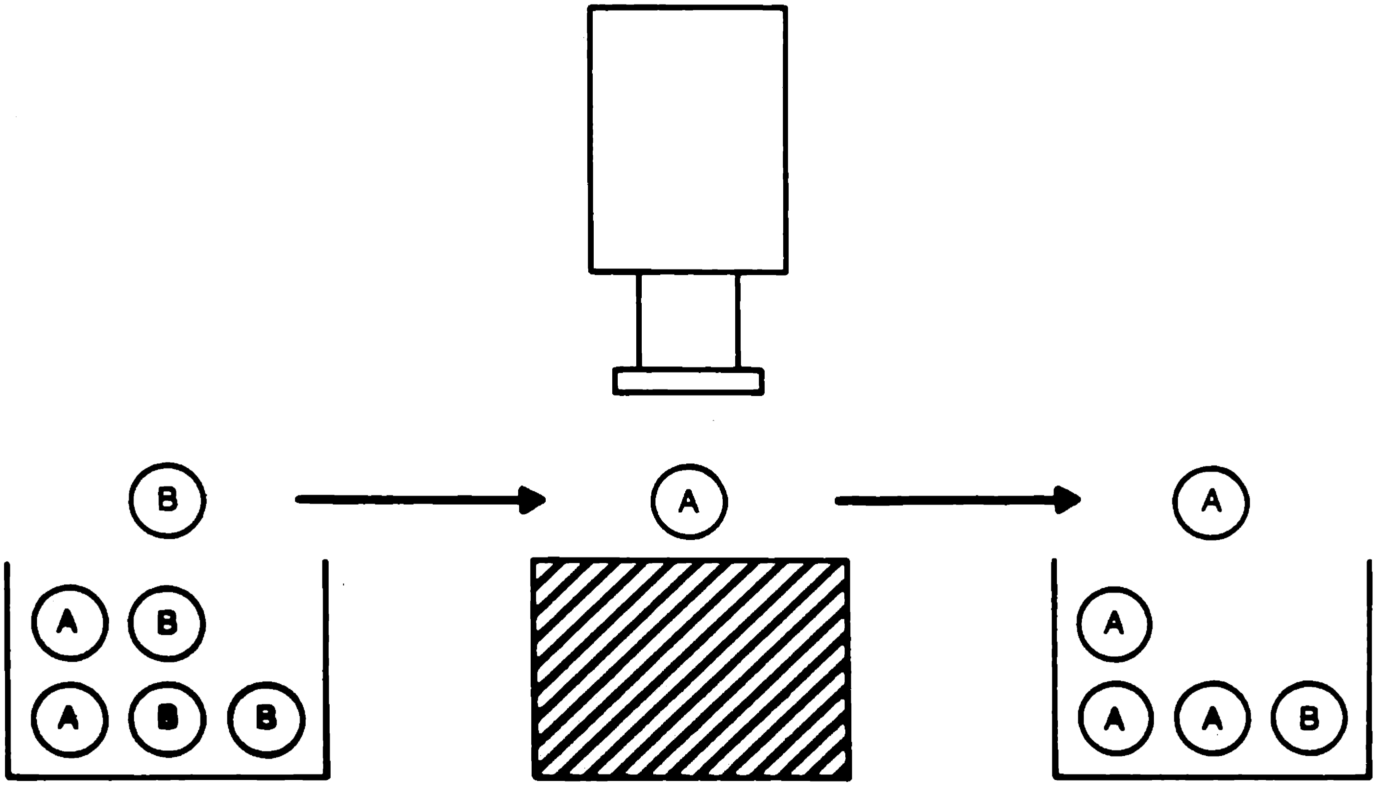


FIGURE 3.1 WORK STATION WITHOUT DELAY

station consists of a single machine producing two part types. The operation time is T.

To accurately model the material flow as a continuous process, it is reasonable to measure the rate of flow over a time period that is much greater than T. For example, if T=1 min., then we could measure, at least approximately, a continuum of "parts/hour" rates, but we could not really measure a continuum of "parts/min.". (i.e. We would see 1 part or 0 parts only.)

We define the time scale of parts production for this process to be of order T, meaning that during intervals of time less than T very few part production events can take place. The basis for the assumption that the processing time is negligible in the Kimemia-Gershwin model is, therefore, that the time parts spend in the system is of order T or less.

Increasing the processing time, T, does not in itself introduce delay. We can see this by noting a change in the production rate at the input of the system in Figure 3.1 still results a change in the output within T time units. It does not make sense, therefore, to define a delay in the "flow" within this time period, since we require a period of at least several T to define a flow in the first place.

For example, in Figure 3.1, a change in production from all Type A production to all Type B production, would look like an instantaneous change in the rate of flow if measured over a time span much larger than T, even though T may be several hours. It is therefore reasonable to model flow rates at the input and output of this processes as identical at any given time. What has changed by increasing T is the time scale of parts production.

This non-delay model is, however, not appropriate for all systems. For example, Figure 3.2 shows a conveyer belt dryer process. In this process, parts enter the system, pass under the dryer and then exit.

The time between the loading of individual parts may be much less than the time spent in the dryer, especially if the conveyer belt process is slow. For example, suppose parts are loaded once a minute and the drying process takes 1 hour. If we have been producing Type A's all morning and then start loading Type B's at 12:00 noon, then from 12:00 to 1:00 we will measure a flow rate of 60 B's per-hour at the input, while at the output, we will still measure 60 A's per-hour. There is a true delay in the rate.

It is not the processing time that introduces this delay but the processing time relative to the interarrival time of parts. For example, if the conveyer belt is short or very fast and therefore holds only one or two parts, then the production rate at the input and output would be the same. If the belt typically holds many parts at a time, the output rate will be delayed relative to the input rate.

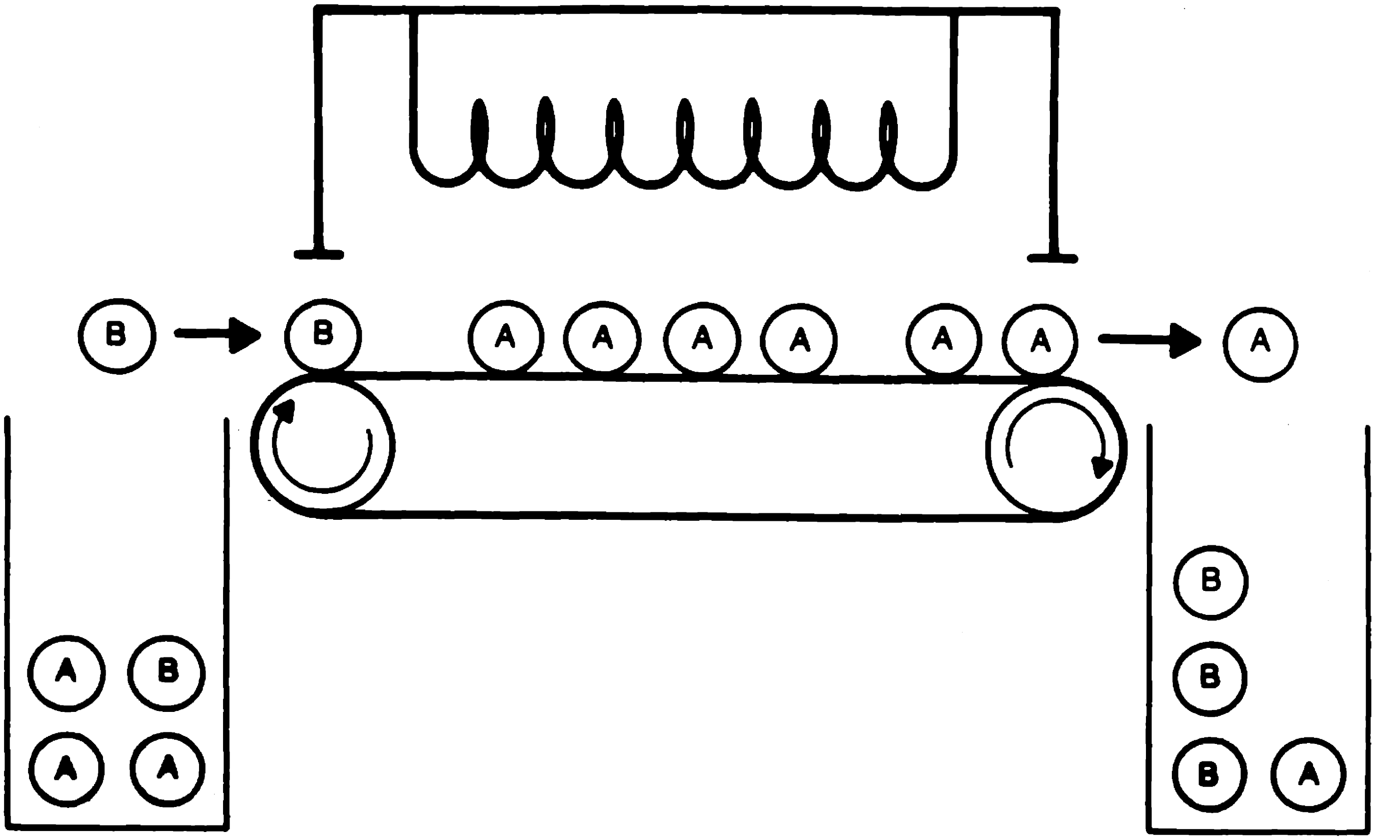


FIGURE 3.2 SERIAL DELAY PROCESS

To see how delay may arise for more general systems, consider a workstation with N machines of the type we saw in Figure 3.1. Such a system is shown in Figure 3.3.

Parts in this system can arrive N times faster than in the Figure 3.1 system but the processing time for each part, T, is still the same. The time scale of the flow in the Fig. 3.3 system will thus be 1/N'th the time scale of the flow in the Figure 3.1 system.

A change in loading rate in the Figure 3.2 system does not change the output rate immediately because parts currently in the machines must complete their operations before the rate of finished parts at the output is equal to the new loading rate. The process flow speeds up as the number of machines increases but the processing time remains fixed. Therefore, the processing time becomes an increasingly larger multiple of the interarrival time as the number of parts in the system increases. For large enough N, the processing time becomes a significant delay.

This phenomenon is a simple consequence of Little's Theorem (Little 1961). Little's Theorem states that if N is the mean number of parts in a system, T is the average time spent in the system and λ is the arrival rate, then

$$N = \lambda T \qquad (3.1)$$

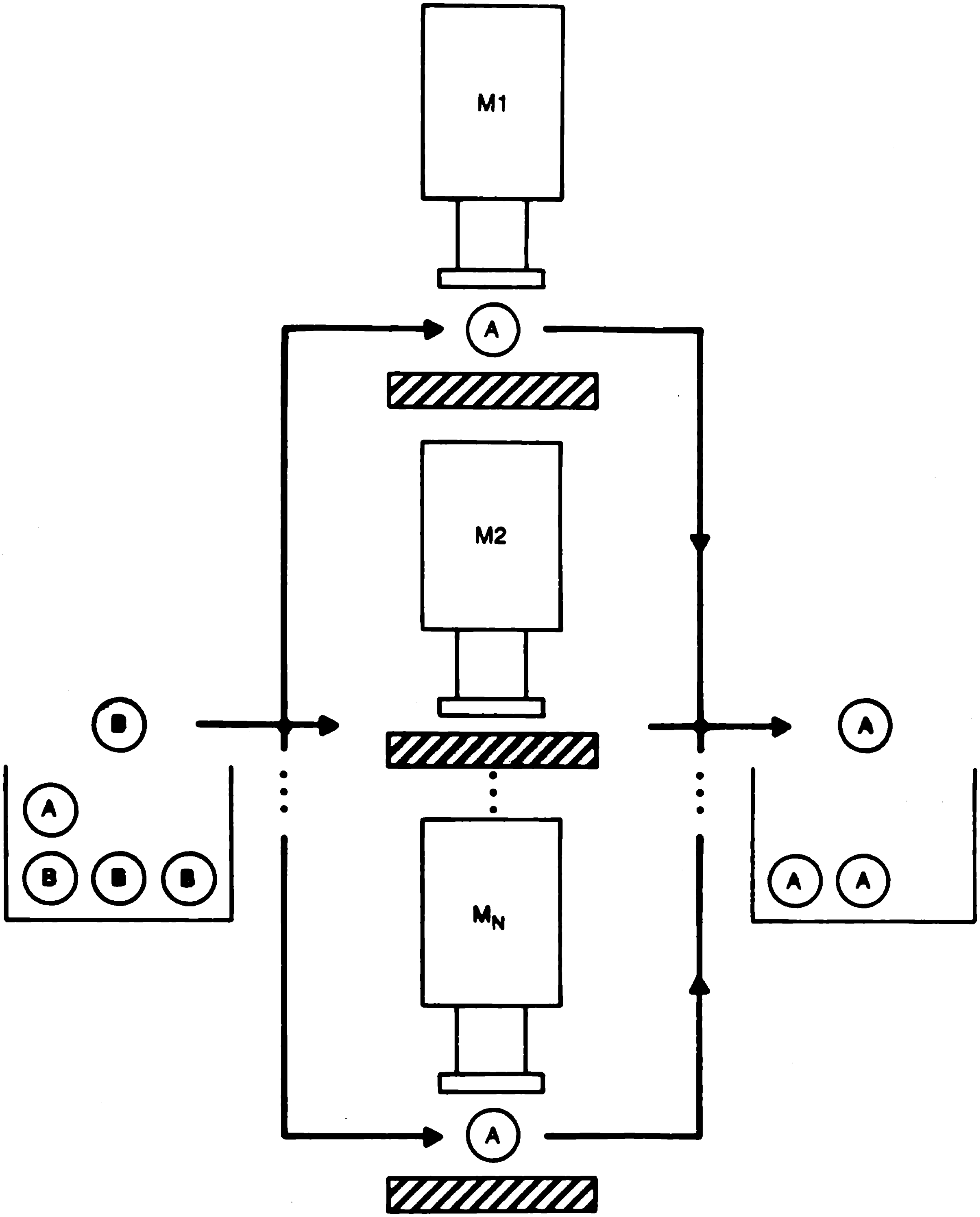


FIGURE 3.3 PARALLEL DELAY PROCESS

Writing (3.1) as $T=N(1/\lambda)$, we see that the system time, T, is N times the average interarrival time. For N large, T is large compared to the interarrival time and thus it represents a significant delay.

This observation suggests that any manufacturing work-station or cell that operates on many parts concurrently should be modeled with a delay between input and output.

This result has significant consequences for hierarchical strategies. In a hierarchical control system, higher levels will typically deal with more highly aggregated portions of the factory, which will typically involve many machines. Thus, high level controllers will typically need to consider the effects of delay.

In particular, the model of Chapter 2 is subject to this same delay phenomenon. A workstation may be a conveyer belt or other serial process, or a workstation may consist of many identical machines in parallel (e.g. the number of machines, L_m, is much greater than one for some workstation m). In either case there will be flow rate delays in the process.

3.2 WORK STATION DELAY IN A FMS

We first examine delay within the workstations of an FMS. Delays within workstations can be accommodated within the basic model, though at the expense of requiring more internal inventory than may otherwise be desirable.

In implementing a u(t) in an FMS that satisfies the instantaneous capacity constraint $\Omega(\alpha)$, the assumption is that all the workstations have material available to work on. A $u(t) \epsilon \Omega$ is considered feasible because each workstation has capacity to produce at rate u(t) at time t. However, we must also guarantee that all workstations have a sufficient supply to implement a production rate.

How much material is enough? The answer, roughly speaking, is enough to ensure that any transition from a rate u' to a rate u'' does not leave a workstation starved. More precisely, let us assume that material produced by workstation m accumulates in a single downstream buffer reserved only for workstation m. Define

$$B_{nm}(t) = \text{the number of type n parts in the buffer downstream of workstation m at time t} \quad (3.2)$$

and

$$B_m(t) = [\ B_{1m}\ B_{2m} \ldots B_{Nm}\ ]^T \tag{3.3}$$

Let the delay of workstation m be denoted s_m and $u^m(t)$ be the partial vector of production rates, $u_n(t)$, at workstation m. That is, $u^m(t)$ only contains components, u_n^m, that correspond to part types produced by workstation m. Since the rate u(t) is the same throughout the system, we have

$$\dot{B}_m(t) = u^m(t - s_m) - u^m(t) \tag{3.4}$$

If we require $u(t) \epsilon \Omega(\alpha)$, there will be a fixed amount of material contained in a workstation and its downstream buffer because the production rate at the input to workstation m equals the rate at which material is removed from the buffer. Let C_m be this fixed amount of material. Symbolically,

$$C_m = \int_{t-s_m}^{t} u^m(\sigma)\ d\sigma + B_m(t) \tag{3.5}$$

Using this definition we obtain,

$$B_m(t) = C_m - \int_{t-s_m}^{t} u^m(\sigma)\ d\sigma \tag{3.6}$$

Now let U^m be the component-wise, least upper bound on u^m due to the finite machine capacities. That is,

$$U^m - [\ U_1^m\ U_2^m\ \dots\ U_N^m\ ]^T \qquad (3.7)$$

$$u^m{}_n(t) \leq U_n^m \text{ for all } t \qquad (3.8)$$

The bound, U_n^m, is the maximum rate at which part type n can be produced by workstation m. Then $B_m(t) \geq 0$ ⇒

$$C_m \geq \int_{t-s_m}^{t} u^m(\sigma)\, d\sigma \qquad \text{for all } t \qquad (3.9)$$

which means C_m must satisfy

$$C_m \geq \int_{t-s_m}^{t} U^m\, d\sigma - U^m s_m \qquad (3.10)$$

$U^m s_m$ is therefore the smallest value of C_m that guarantees that any $u(t) \epsilon \Omega(\alpha)$ will never empty a buffer. Letting C_m take this minimum value and noting that the average amount of material in workstation m is $\bar{u}^m s_m$, where $\bar{u}^m$ is the average production rate at station m, we get

$$E[\ B_m(t)\ ] = s_m\ (\ U^m - \bar{u}^m\ ) \tag{3.11}$$

Equation (3.11) gives the minimum average buffer level for each workstation when operating the FMS according to the instantaneous capacity rule, $u(t) \in \Omega(\alpha)$.

Note that for $s_m \rightarrow 0$, the average buffer level is zero. The average buffer level is also small if $U^m \approx \bar{u}^m$, which would be the case for a highly reliable FMS operated very close to capacity. In general the higher the average production rate, $\bar{u}^m$, the smaller the average inventory level needed to ensure no-starvation. This is somewhat counterintuitive, but recall that the reason we keep this inventory is to support a workstation during <u>changes</u> in the production rate. A system that is heavily loaded will tend to use its inventory more often and thus the average level of the inventory will be lower under heavy loading.

Because the minimum average buffer level is proportional to s_m for this class of policies, long delays within workstations will require large inventories. This suggests that subdividing the FMS into several blocks may be necessary to reduce the level of the internal workstation buffers. We will examine the control of a subdivided system in Chapter 5. In the next section, we examine the impact of delay on single FMS.

3.3 A SINGLE FMS WITH DELAY

If we choose to maintain the minimum internal material levels, $U^m s_m$, given in the previous section, we must still consider the effect of the end-to-end delay through the FMS. We show the resulting FMS delay problem for a single FMS is a simple extension of the problem in Chapter 2.

Consider the system described in Section 2.1 but with a new delay state equation due to internal workstation delays, namely

$$\dot{x}(t) = u(t-s) - d \, . \tag{3.12}$$

We still have the same constraint $u(t) \epsilon \Omega(\alpha(t))$.

The state of the system now includes not only $x(t)$ and $\alpha(t)$ but also the past controls, $\{u(\sigma) | t-s \leq \sigma \leq t\}$. We will denote this set $\mathit{u}(t)$. The value function then becomes

$$J(x,\mathit{u},\alpha) = \lim_{T\to\infty} \frac{1}{T} E\{ \int_0^T g(x(t)) \, dt \mid x(0) = x \, , \, \alpha(0) = \alpha \, , \mathit{u}(t) = \mathit{u} \} \tag{3.13}$$

Note, however, that $x(t)$ is completely determined over the interval $0 \leq t \leq s$ by the initial conditions x and u, that is

$$x(t) = x + \int_0^t u(-\sigma)\, d\sigma - td \qquad 0 \le t \le s \tag{3.14}$$

This means we can write the value function as

$$\begin{aligned} J(x,u,\alpha) = & \lim_{T\to\infty} \frac{1}{T} \int_0^s g(x(\sigma))d\sigma \\ & + \lim_{T\to\infty} \frac{1}{T} E\{ \int_s^T g(x(t))\, dt \mid x(s) = \hat{x}\ ,\ \alpha(0) = \alpha \ \} \end{aligned} \tag{3.15}$$

where

$$\hat{x} = x + \int_0^s u(-\sigma)\, d\sigma - sd \qquad 0 \le t \le s. \tag{3.16}$$

The first limit in (3.15) is zero. By comparing the second term to (2.16), we see this problem is equivalent to the Chapter 2 problem if we use $\hat{x}$ in place of x in (2.16). The strategy is, therefore, to calculate $\hat{x}$, the future value of x, watch the current α and apply the control (2.20) using $\hat{x}$ in place of x. The minimum buffer stocks between workstations guarantee that the rate will be feasible.

3.4 A PROBLEM WITH INTERCONNECTED FMS'S AND DELAY

In the previous section, we saw by simply computing the future value of $x(t)$ and using it in the control law for a non-delay problem, that we could get an optimal control for a single FMS with delay. Unfortunately, this simple manipulation does not work for interconnected FMS's. In this section we will examine an example of a simple interconnection of FMS's, namely the two FMS system shown in Figure 3.4.

The system in Figure 3.4 consists of two FMS's in tandem. Each FMS is of the type presented in Section 2.1. There are two independent machine states, α_1 and α_2, and two capacity sets, Ω_1 and Ω_2. There is a buffer between the machines whose level we denote $B_1(t)$.

The system equations are

$$\dot{B}_1(t) = u_1(t-s) - u_2(t) \qquad (3.17)$$

$$\dot{x}_2(t) = u_2(t) - d \qquad (3.18)$$

Because $B_1(t)$ is a physical buffer level, it is positively constrained

$$B_1(t) \geq 0 \text{ for all } t. \qquad (3.19)$$

$x_2(t)$ is a production surplus as in the single FMS case.

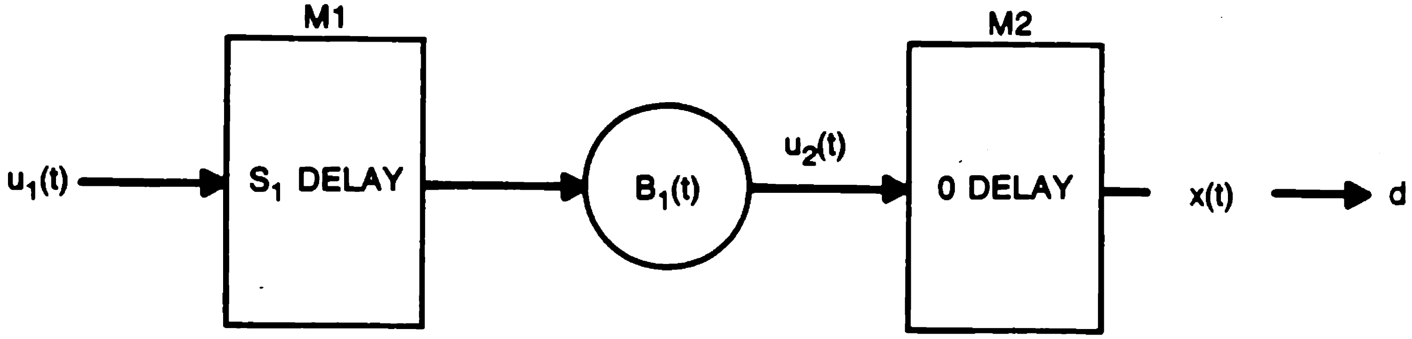


FIGURE 3.4 TANDEM FMS SYSTEM WITH DELAY

We have assumed that only the first system has delay, since we can advance the surplus, x_2, as shown in the previous section to put the system equations in the above form.

We now have a cost function

$$g(B_1,x_2) = g_1(B_1) + g_2(x_2) \qquad (3.20)$$

where g_1 and g_2 are functions of the form described in Section 2.1.

Note that trying to advance the state equation to get B(t+s) yields

$$B(t+s) = B(t) + \int_0^s u_1(t-\sigma)\, d\sigma - \int_0^s u_2(t+\sigma)\, d\sigma \qquad (3.21)$$

The first two terms are known since they involve only the current B and the past u_1. The third term, however, is the integral of the future u_2. This term would be computable if we knew the value of α on the interval [t,t+s], but α is random.

The problem arises because the differential equations are now functions of both the present and past control and not just the past control, as was the case for the single FMS with delay. The interconnection problem yields a value function that is a general function of the past u_1. That is

$$J = J(B_1, x_2, \mathcal{U}_1, \alpha). \tag{3.22}$$

The evolution of B_1 and x_2 is described by the delay differential equations, but the evolution of the set $\mathcal{U}_1$ is difficult to describe. If a differential equation does not exist to describe the evolution of $\mathcal{U}_1$, it is not clear, a priori, that the linear program control form

$$\min_{u \in \Omega} \nabla J^{*} u \tag{3.23}$$

will be valid for interconnected FMS problems with delay.

Whether the optimal control still satisfies a linear program of this form and what that linear program looks like will be addressed in the next chapter.

CHAPTER 4 - A LINEAR SYSTEM APPROXIMATION TECHNIQUE FOR SYSTEMS WITH DELAY

In Chapter 3 we saw that a network of FMS's with delay could not be solved by simple translation in time. We also observed that the evolution of the past controls is not easily described. In this chapter, a technique that allows us to approximate the evolution of the past controls by using a system of ordinary differential equations is developed.

The technique is based on the work of Repin (1965). Hess (1972) and Hess and Hyde (1973) used the same approach for approximating delay in a linear-quadratic problem. Hess and Hyde were able to use the approximation to get an ordinary differential system that could then be solved using the steady state Ricatti equation. They then showed that the approximation yielded the same matrix-integral equation form derived by Ross and Flugge-Lotz (1969).

We reverse this order and use the approximation technique to generate the form of the control law for a system of interconnected FMS's directly. This approach, however, requires that we know the solution to the non-delay problem, which, as we mentioned in Chapter 1, is only available for scalar problems. Therefore, we must approximate the value function.

We examine the form of the control for a delay problem using a quadratic approximation of the value function. Quadratic approximations were used successfully by Akella, Choong and Gershwin (1984) for non-delay systems and provide a reasonable approximation to the convex value function within a small neighborhood of a nominal operating point (e.g. hedging point).

The gradient of the quadratic approximation yields a convolution term involving the past controls and a weighting function. A heuristic predictor interpretation of the resulting control is then suggested.

4.1 A LINEAR SYSTEM APPROXIMATION FOR A DELAY LINE

In this section, we develop the delay approximation using a simple first order differential equation, or link. Bounds on the error introduced by this approximation are then developed. We then improve this approximation by using multiple links in series. However, the approximation is only good for functions that satisfy certain smoothness conditions. It turns out, however, that for an FMS problem only the integral of the error must approach zero. Based on this fact, an important lemma, Lemma 4.1, is developed to justify the use of the linear system approximation for a FMS.

4.1.1 Single Differential Equation Approximation

We begin by considering a general delay line defined by

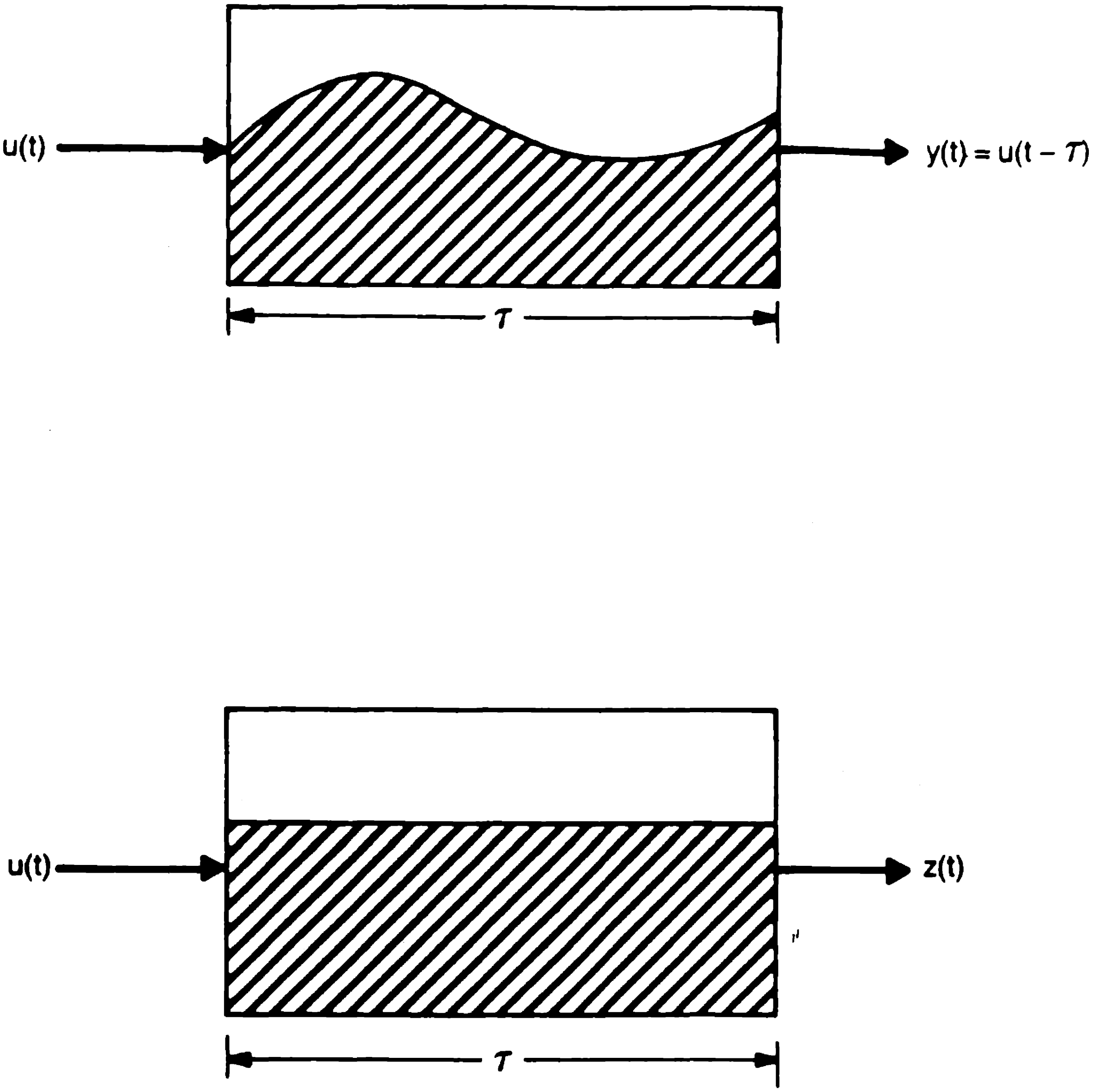
$$y(t) = u(t-\tau). \qquad (4.1)$$

We will assume that u(t) is bounded and absolutely integrable and that $u(t)=0$ for $t\leq 0$. In the model presented in Chapter 2, this delay line would represent the delay through a workstation. The output, y(t), would feed a buffer and the input, u(t), would be the production rate.

To see how one might approximate a delay line by a linear system, consider the two "snapshots" of a delay line shown in Figure 4.1. The top figure shows the actual distribution of material in the line at time t. The bottom figure shows the same amount of material but averaged over the entire delay line. Let z(t) be the average value of u(t) in the line. Then $\tau z(t)$ is the total amount of material in the line.

If we now let time advance by a small increment, δt, u(t) will enter and $u(t-\tau)$ will leave. The value z(t) will therefore change according to

$$\tau\ z(t+\delta t) = \tau\ z(t) + u(t)\delta t - u(t-\tau)\delta t. \qquad (4.2)$$



$$\tau\dot{z}(t) + z(t) = u(t)$$

FIGURE 4.1 "SNAPSHOT" OF MATERIAL IN A DELAY LINE

We now make a simplifying assumption. Suppose we approximate the output of the delay line, $y(t)=u(t-\tau)$, by the average value variable $z(t)$. That is assume

$$u(t-\tau) \approx z(t). \tag{4.3}$$

We will establish error bounds on this approximation shortly. Using this approximation in (4.2) we get

$$\tau\ z(t+\delta t) = \tau\ z(t) + u(t)\delta t - z(t)\delta t \tag{4.4}$$

or

$$\tau\ \frac{z(t+\delta t) - z(t)}{\delta t} + z(t) = u(t). \tag{4.5}$$

Letting $\delta t \rightarrow 0$ gives

$$\tau\ \dot{z}(t) + z(t) = u(t). \tag{4.6}$$

If the distribution of material in the original delay line is smooth, $z(t)$ can be a reasonable approximation to the output rate, $y(t)$, of the delay line. The error depends on how smooth $u(t)$ is. To evaluate the approximation, we consider the case where $u(t)$ is twice differentiable with $\ddot{u}(t) \le K$. This condition can be relaxed to include inputs with bounded first derivatives as shown in Appendix A. The proof is more tedious but follows an argument very similar to the one presented below.

We begin by defining the error

$$e(t) \triangleq z(t) - y(t) . \tag{4.7}$$

We will assume that z(0)=0. To evaluate |e(t)| we can write

$$\dot{e}(t) = \dot{z}(t) - \dot{y}(t) = \frac{u(t)-z(t)}{\tau} - \dot{u}(t-\tau) \tag{4.8}$$

$$= -e(t)/\tau + \phi(t) \tag{4.9}$$

where

$$\phi(t) = \frac{u(t)-u(t-\tau)}{\tau} - \dot{u}(t-\tau). \tag{4.10}$$

Note that for some $\bar{t}\epsilon[t-\tau,t]$ we have, by the mean value theorem,

$$\dot{u}(\bar{t}) = \frac{u(t)-u(t-\tau)}{\tau} . \tag{4.11}$$

Therefore,

$$|\ \phi(t)\ | = |\ \dot{u}(\bar{t}) - \dot{u}(t-\tau)\ | \tag{4.12}$$

$$= |\ \int_{t-\tau}^{\bar{t}} \ddot{u}(\xi)\ d\xi\ | \tag{4.13}$$

$$\leq |\ \bar{t} - (t-\tau)\ |\ K \leq K\tau. \tag{4.14}$$

We show in Appendix A, Fact 1, that (4.9) has a simple bounded-input, bounded-output condition, namely

$$| e(t) | \le \tau | \phi(t) |. \tag{4.15}$$

Therefore, by combining (4.15) and (4.16) we have the following bound on the error e(t):

$$| e(t) | \le K\tau^2. \tag{4.16}$$

4.1.2 m'th Order Approximation

By splitting the delay line into more than one segment, as shown in Figure 4.2, the differential equation approximation can be improved. In Figure 4.2, the output of link j, $z_j(t)$, is the input to link j+1. Each link is a differential system like that in Figure 4.1 but with delay τ/m. Therefore, the evolution of these segments, or links, is described by

$$(\tau/m)\dot{z}_1(t) + z_1(t) = u(t) \tag{4.17}$$

$$(\tau/m)\dot{z}_2(t) + z_2(t) = z_1(t)$$

$$\cdot$$

$$\cdot$$

$$\cdot$$

$$(\tau/m)\dot{z}_m(t) + z_m(t) = z_{m-1}(t)$$

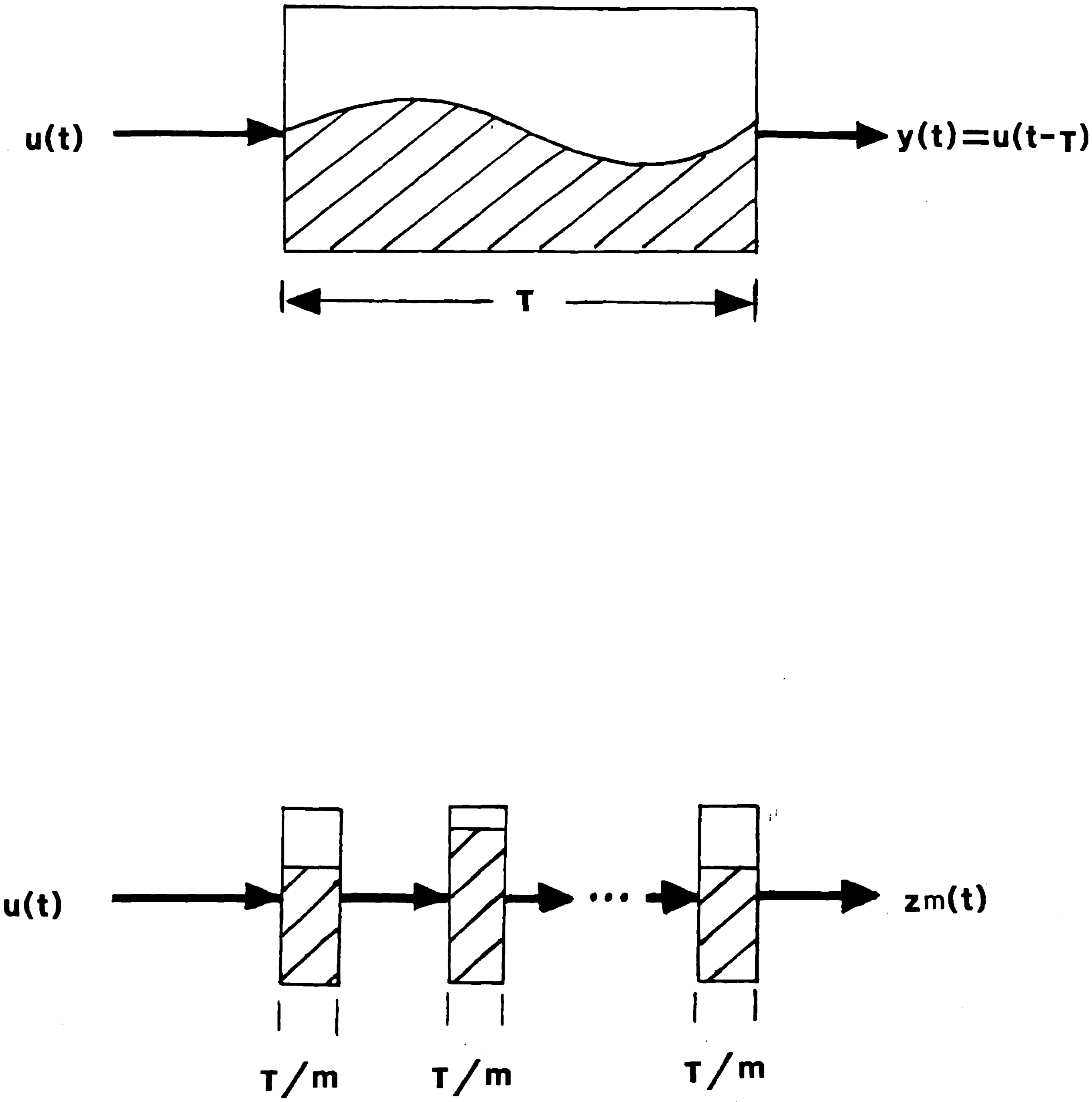


FIGURE 4.2 m-TH ORDER APPROXIMATION

where we assume $z_j(0)=0$, $j=1,2..m$.

The variables $z_j(t)$ approximate $u(t-j\tau/m)$ with an error which we define by

$$e_j(t) = z_j(t) - u(t-j\tau/m). \tag{4.18}$$

The input to link $j+1$ is $z_j(t)$, which consists of two terms, $e_j(t)$ and $u(t-j\tau/m)$. We show in Appendix A that the error due to these two terms contribute to $e_{j+1}(t)$ according to

$$| e_{j+1}(t) | \le | e_j(t) | + K (\tau/m)^2 . \tag{4.19}$$

By (4.16), the error on the first link, $e_1(t)$, is bounded by $K(\tau/m)^2$. Therefore we have

$$| e_m(t) | \le m K(\tau/m)^2 = K \tau^2/m \tag{4.20}$$

Recalling that $e_m(t)=z_m(t)-u(t-\tau)$, (4.20) implies

$$z_m(t) \rightarrow u(t-\tau) \tag{4.21}$$

as $m\rightarrow\infty$.

Appendix A contains the complete proof of this fact following the work of Repin (1965). This result is also shown to hold for a u(t) that has only a bounded first derivative.

To motivate the approximation in the frequency domain, note that the transfer function for each of each equations in (4.17) (from the input z_{j-1} to the output z_j) is given by

$$\frac{1}{1+\Delta t s} \tag{4.22}$$

Therefore the transfer function from u to z_m is

$$\frac{Y_m(s)}{U_1(s)} = \left(\frac{1}{1+\Delta t s}\right)^m \tag{4.23}$$

which can be written

$$\frac{Y_m(s)}{U_1(s)} = \frac{1}{\left(1+\frac{\tau s}{m}\right)^m} \tag{4.24}$$

Letting $m \to \infty$, the denominator approaches $e^{\tau s}$ so we get

$$\frac{Y_m(s)}{U_1(s)} \to e^{-\tau s} \tag{4.25}$$

as $m \to \infty$.

The right hand side of (4.25) is simply the transfer function of a delay line. Since u(t) is the input to this line we expect that $z_m(t)$ approaches $u(t-\tau)$. This is not completely correct if u(t) has discontinuities, however, since the linear system cannot "pass" a discontinuous u(t).

4.1.3 Linear System Approximation for a FMS

In a FMS problem, the capacity constraint, Ω, is polyhedral. The input u(t) may jump from one extreme point to another and, therefore, may not be continuous. However, it is bounded. In Appendix A, we show that although the values of $z_m(t)$ do not converge pointwise to $u(t-\tau)$, the integral of $z_m(t)$ converges to the integral of $u(t-\tau)$. That is

$$\int_0^t z_m(\xi)\ d\xi \rightarrow \int_0^t u_l(\xi-\tau)\ d\xi \tag{4.26}$$

as $m \rightarrow \infty$.

We show below that the convergence of the above integrals is sufficient to prove the approximation holds for a continuous flow FMS model. First, however, we outline the proof of (4.26).

To prove (4.26), we interchange integration at the output of the m links with integration at the input. The integral of u(t) has a bounded first derivative since u(t) is bounded. Therefore, the integral of u(t) satisfies a bounded first derivative condition. If we consider the integral of u(t) as the input to the system, convergence follows from our previous argument. A complete proof is contained in Appendix A. We summarize this result from Appendix A in the following important lemma.

Lemma 4.1

Let u(t) be absolutely integrable with $u(t)=0$ for $t\le 0$ and $|u(t)|\le B$ for all $t>0$. If u(t) is applied to both the delay element

$$y(t) = u(t-\tau) \tag{4.27}$$

and the m'th order system defined by (4.17) with initial condition $z_j(t)=0$, $j=1,2,..m$, then

$$\int_0^t (z_m(\xi) - y(\xi)) \, d\xi \rightarrow 0 \tag{4.28}$$

uniformly as $m\rightarrow\infty$.

■

Note that convergence of the integral is all we need to establish an equivalence between a control problem generated from a delay system and a control problem generated using the m'th order approximation. This follows from the fact that the production state variable, x(t), is a function of the integrals of the controls only. Therefore the states of the original delay system and the m'th order approximate system will be the same for $m\rightarrow\infty$, if they follow the same control policy and have the same machine states.

It thus follows that an optimal control policy for the m'th order approximate system will converge to an optimal control policy for the original system as m→∞. We summarize this observation in the following lemma.

Lemma 4.2

Consider Figure 4.3. Let system D be a multiple FMS system with delay and let system A be an identical system to D with each delay element replaced by the m'th order linear system (4.17). Let z_j^k be the j'th approximation variable of FMS k=1,2..K, s_k be the delay of FMS k and $u^k(t)$ be the input to FMS k. If

$$\mu^*(\alpha , z_j^k , j=1,2,..m, k=1,2,..K) \tag{4.29}$$

is the optimal control for system A, then, for m→∞,

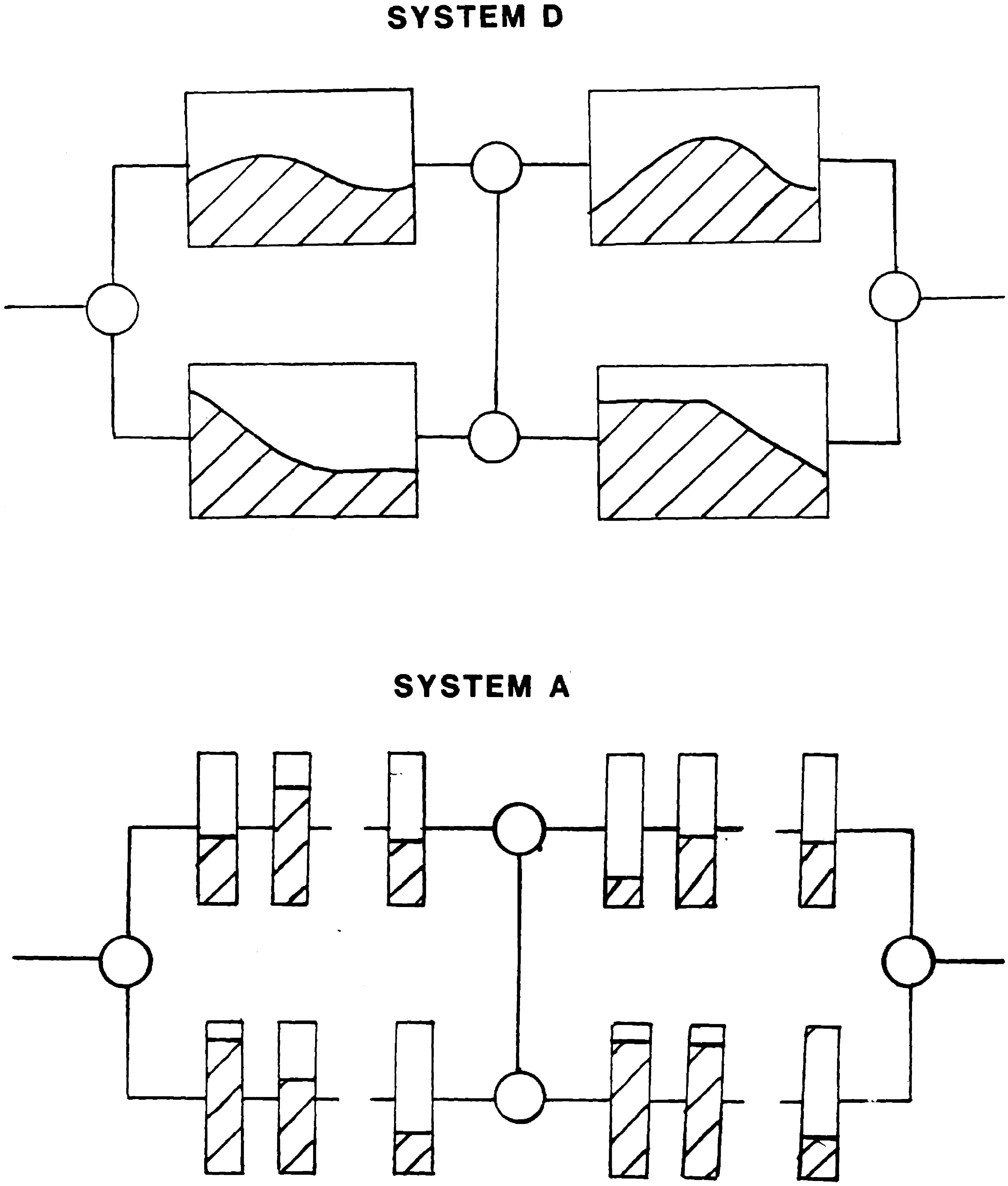
$$\mu^*(\alpha , u^k(t-j\Delta t_k/m), j=1,2,...m, k=1,2,..K) \tag{4.30}$$

where

$$\Delta t_k = \frac{s_k}{m} \tag{4.31}$$

is also optimal for system D.

■



SYSTEM D

SYSTEM A

FIGURE 4.3 DELAY AND APPROXIMATE SYSTEMS

Lemma 4.2 allows us to avoid the difficulty of describing the evolution of the delay elements in an FMS system by using ordinary, albeit infinitely large, systems of differential equations in place of the delay elements. The differential system can then be used in the optimal control law of Kimemia and Gershwin to derive the optimal control for system with delay.

In the next section we give a simple example of how the linear system approximation can be applied to the FMS model.

4.2 AN EXAMPLE OF DELAY APPROXIMATION

In this section, we apply the linear approximation to a simple, single-workstation system. We show how the approximation can be used to generate a theoretical control for a system with delay. Consider the two-machine system described in Chapter 3.4 and shown in Figure 3.4 where the system produces only a single part type.

We suppress the subscripts on B_1 and x_2 in this section to simplify the notation. The system equations are therefore given by

$$\dot{B}(t) = u_1(t-s) - u_2(t) \tag{4.32}$$

$$\dot{x}(t) = u_2(t) - d \tag{4.33}$$

We make the m'th order approximation to this system:

$$\dot{\tilde{B}}(t) = \frac{z_m(t)}{\Delta t} - u_2(t) \tag{4.34}$$

$$\dot{x}(t) = u_2(t) - d \tag{4.35}$$

$$\dot{z}_1(t) = u_1(t) - \frac{z_1(t)}{\Delta t} \tag{4.36}$$

$$\dot{z}_2(t) = \frac{z_1(t) - z_2(t)}{\Delta t}$$

$$\vdots$$

$$\dot{z}_m(t) - \frac{z_{m-1}(t) - z_m(t)}{\Delta t}$$

where

$$\Delta t - \frac{s}{m} \tag{4.37}$$

Note that the equation for z_1 is different from the equation presented in Section 4.1.2. We have multiplied u(t) by Δt at the input and divided $z_m(t)$ by Δt at the output.

This manipulation does not alter the approximation (cf. Corollary in Appendix A). We do this to give the variables a physical meaning. Note that in the previous section, $z_1(t)$ was approximately equal to u(t-iΔt) (except in the neighborhood of a discontinuity) and thus has units of material per unit time. The variables, z_i, after the above manipulation, now have the following interpretation:

$$z_j(t) \approx u(t-j\Delta t)\Delta t \approx \int_{t-(j-1)\Delta t}^{t-j\Delta t} u_i(\xi)\ d\xi \qquad (4.38)$$

In words, $z_j(t)$ is approximately the amount of material that was loaded during the time interval $[t-(j-1)\Delta t, t-j\Delta t]$. Each z_j now has units of material. The last variable, z_m, enters the the equation for B(t) as $z_m(t)/\Delta t$ and thus becomes a rate again.

Note that the Σz_j is approximately the total amount of material in the delay element. This physical interpretation will be helpful when we approximate the value function.

We can rewrite the m'th order system as

$$\dot{X}(t) = \frac{1}{\Delta t} A\ X(t) + B\ U(t) - C\ d \qquad (4.39)$$

where

$$X(t) = \begin{bmatrix} B(t) \\ x(t) \\ z_1(t) \\ . \\ . \\ . \\ z_m(t) \end{bmatrix} \qquad U(t) = \begin{bmatrix} u_1(t) \\ u_2(t) \end{bmatrix} \tag{4.40}$$

and

$$A = \left[\begin{array}{ccccccc} 0 & 0 & 0 & 0 & 0 & \ldots & 1 \\ 0 & 0 & 0 & 0 & 0 & \ldots & 0 \\ 0 & 0 & -1 & 0 & 0 & \ldots & 0 \\ 0 & 0 & 1 & -1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & -1 & 0 & \ldots & 0 \\ . & . & & . & . & & \\ . & . & & & . & . & \\ . & . & & & & & \\ 0 & 0 & & & . & 1 & -1 \end{array}\right] \tag{4.41}$$

$$B = \begin{bmatrix} 0 & -1 \\ 0 & 1 \\ 1 & 0 \\ 0 & 0 \\ . & . \\ . & . \\ . & . \\ 0 & 0 \end{bmatrix}$$

We now form the optimization problem for the m'th order system as we did in Chapter 2 and get the value function

$$J_u(x,B,z_1,\ldots z_m,\alpha) = J_u(X,\alpha) \tag{4.42}$$

$$= \min_{u\in\Omega(\alpha)} \lim_{T\to\infty} \frac{1}{T} E\left[\int_t^{t+T} (g_1(B)+g_2(x)) \; dt \mid x(t),B(t),z_1(t),\ldots,z_m(t),\alpha \right]. \tag{4.43}$$

As before, the optimal J satisfies

$$0 = \min_{u\in\Omega(\alpha)} \{ g_1(x) + g_2(B) + \nabla_x J_u^* \, (AX+Bu-Cd) + \sum_\beta \lambda_{\alpha\beta} \, J_u^*[X,\beta] \} \tag{4.44}$$

and thus the optimal u is given by

$$\min_{u\in\Omega(\alpha)} \nabla_x J_u^* \; Bu \tag{4.45}$$

or,

$$\min_{u\in\Omega(\alpha)} \frac{\partial J_u^*(X,\alpha)}{\partial z_1} u_1 + \left[\frac{\partial J_u^*(X,\alpha)}{\partial x} - \frac{\partial J_u^*(X,\alpha)}{\partial B} \right] u_2 \tag{4.46}$$

Equation (4.46) gives the optimal control for the m'th order approximation. From the observation in Lemma 4.2, this control becomes optimal for the delay system as m→∞. The result shows that the optimal control does satisfy a linear program. The coefficients of the linear program are functions of the set of approximation variables, X, and α. In the next section, the form of these coefficients for a quadratic approximation to the value function is examined.

4.3 DELAY APPROXIMATION WITH A QUADRATIC VALUE FUNCTION

If analytical techniques existed for solving (4.46), then a finite m could be selected and the function

$$J^*(x,B,z_1,z_2,..z_m)$$

could be generated. Interpolating between the values of z_j would yield a reasonable approximation to the optimal value function. One could then increase m to get an increasingly accurate approximation.

An analytical solution to (4.46) does not exist, however, so approximations must be used. Rishel (1973) shows that the optimal value function (4.42) is both continuous and convex. We therefore try a quadratic approximation of the form.

$$\mathfrak{J}^*(X,\alpha) = \frac{1}{2} X^T Q(\alpha) X + R(\alpha)^T X \tag{4.47}$$

where $Q(\alpha) \geq 0$.

Proceeding informally, we use this approximation in (4.46) and get

$$\min_{u \in \Omega(\alpha)} (Q(\alpha) X + R(\alpha)) B u \tag{4.48}$$

which yields

$$\min_{u \in \Omega(\alpha)} c_1(X,\alpha) u_1 + c_2(X,\alpha) u_2 \tag{4.49}$$

where

$$c_1(X,\alpha) = \sum_{i=3}^{m+2} q_{3i} z_i(t) + q_{31} B(t) + q_{32} x(t) + r_3 \tag{4.50}$$

$$\begin{aligned} c_2(X,\alpha) = \sum_{i=1}^{m+2} (q_{2i}-q_{1i}) z_i(t) + (q_{21}-q_{11}) B(t) + \\ +(q_{22}-q_{12}) x(t) + (r_2-r_1) \end{aligned} \tag{4.51}$$

Defining,

$$f_1(i\Delta t) = q_{3i+2} \qquad (4.52)$$

$$f_2(i\Delta t) = q_{2i+2} - q_{1i+2} \qquad (4.53)$$

and recalling

$$z_i(t) \approx u_1(t-i\Delta t)\Delta t \qquad (4.54)$$

we get

$$c_1(X,\alpha) \approx \sum_{i=1}^{m} f_1(i\Delta t)\ u_1(t-i\Delta t)\ \Delta t + q_{31}\ B(t) + q_{32}\ x(t) + r_3 \qquad (4.55)$$

$$c_2(X,\alpha) \approx \sum_{i=1}^{m} f_2(i\Delta t)\ u_1(t-i\Delta t)\ \Delta t + (q_{21}-q_{11})\ B(t) + (q_{22}-q_{12})\ x(t) + (r_2-r_1) \qquad (4.56)$$

Letting $\Delta t \rightarrow 0$ and recalling that $m\Delta t = s$ we get

$$c_1 = \int_0^s f_1(\xi)\ u_1(t-\xi)\ d\xi + q_{31}\ B(t) + q_{32}\ x(t) + r_3 \qquad (4.57)$$

$$c_2 = \int_0^s f_2(\xi)\ u_1(t-\xi)\ d\xi + (q_{21}-q_{11})\ B(t) + (q_{22}-q_{12})\ x(t) + (r_2-r_1) \qquad (4.58)$$

The coefficients, c_1 and c_2, of the linear program thus have a linear form, which is not surprising given the quadratic assumption. This form is similar to the form given by Ross and Flugge-Lotz (1969) for a LQ problem where the optimal value function is truly quadratic.

This linear form can be verified by noting that the gradient of a quadratic form is linear and using the following theorem from Oguztoreli (1966, Theorem 3.9)

Theorem 4.1

Every linear real-valued functional $J(u)$ in $L_1(0,s)$ can be represented by a Lebesgue integral

$$J(u) = \int_0^s f(\xi)\, u(\xi)\, d\xi \qquad (4.59)$$

where $f(\xi)$ is bounded almost everywhere and is uniquely determined by the functional $J(u)$.

■

The functions $f_1(\xi)$ and $f_2(\xi)$ in (4.57) and (4.58) are functions that weight the value of the material in the system. If they are constant, the integrals in the coefficients c_1 and c_2 simply represent the total amount of material in the delay element (recall the physical interpretation of Σz_i). An increasing $f(\xi)$ would weight material far from the buffer (i.e. loaded later) less that material close the the buffer (i.e. loaded earlier). A decreasing $f(\xi)$ would do the opposite.

In the next section, we look at the results of a simulation experiment based on the LP control with linear form coefficients and then suggest a heuristic interpretations of the weighting functions of the control.

4.4 SIMULATION OF DELAY SYSTEM CONTROLS

To test that the quadratic approximation to the value function and the resulting control law perform reasonably well, we simulated the two simple systems shown in Figure 4.4 and Figure 4.5. A discretized version of the differential equations describing each systems was constructed. The simulations are not detailed part simulations. They simply show the behavior of the differential equations in an FMS model.

4.4.1 Example 1: Single Machine with Delay and Random Demand

The system in Figure 4.4 is the simplest of the two examples. It consists of a single, reliable machine with delay. The machine produces at a maximum rate of 2 parts per unit time. The demand, d(t), on the machine is stochastic. Let $\alpha(t)$ be the state of the demand. If the demand is in the high state, $\alpha(t)=1$, the demand rate is 4 parts per unit time. If demand is in the low state, $\alpha(t)=0$, the demand is 1 part per unit time. The surplus, x(t), is therefore described by

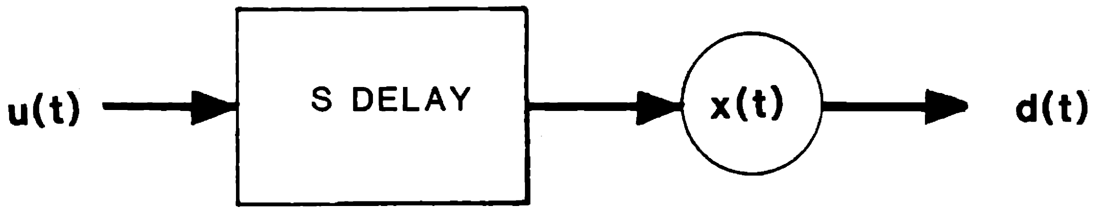


FIGURE4.4 SIMULATION EXAMPLE 1

$$\dot{x}(t) = u(t) - (1 + 3\alpha(t)) \tag{4.60}$$

The transitions between demand states are Markovian with

$$P[\alpha(t+\delta t)=1 \mid \alpha(t)=0] = 0.05 \delta t \tag{4.61}$$

$$P[\alpha(t+\delta t)=0 \mid \alpha(t)=1] = 0.10 \delta t \tag{4.62}$$

The control loads the machine based on the current demand state. However, the state may change by the time the material exits the system. This is in contrast to the single unreliable machine with constant demand discussed in Chapter 3. There, the ability to load, the capacity, is uncertain but the demand after material is loaded is known and hence the value of x(t) when material exits the system is known.

Two controls were used. The controls takes the general form

$$\min_{0 \le u \le 2} c\ u \tag{4.63}$$

The form of the coefficient, c, is different for each control. Control Z used a simple hedging point policy:

CONTROL Z: $$c = x(t) - x_h \tag{4.64}$$

This corresponds to a weighting-function, $f_1(\xi)$ in (4.57) and (4.58), that is zero. Control C uses a constant-weighting-function coefficient:

$$\text{CONTROL C:} \qquad c = x(t) + \int_0^s u(t-\sigma)\, d\sigma - x_h \qquad (4.65)$$

The constant-weighting-function was chosen because of its simplicity. Physically, it makes the integral in (4.65) equal to the total amount of material in the system.

The system was simulated for 10,000 time steps for delays of 10, 30 and 50 time units. The hedging point, average inventory, INV, and backlog, BCK, for the two controls are shown in Table 4.1. The values shown are for the hedging point, x^h, that yielded the lowest INV+BCK cost. As Table 4.1 shows, the constant-weighting- function control resulted in both lower backlog and inventory. Note that the hedging point value is higher for Control C because the hedging point in measured against the control plus the material in the line.

DELAY	CONTROL Z			CONTROL C		
	INV	BCK	X_h	INV	BCK	X_h
10	5.2	6.2	3	4.9	5.5	15
30	6.7	7.1	4	5.6	5.9	48
50	7.2	8.7	4	5.8	6.1	74

TABLE 4.1 - Example 1 Simulation Results for Controls Z and C

The system was animated so that qualitative behavior could be observed. In side-by-side operation, Control Z did not react as well to changes in the demand. If demand changed from low to high, it did not increase production until after the line emptied. When demand changed from high to low, it tended to overshoot the hedging point. Control C, by contrast, tended to keep the value of x(t) more stable, which is apparent from the lower INV and BCK levels in Table 1.

4.4.2 Example 2: Two Machines Feeding One Buffer

Figure 4.5 show the second simulation example. This system consists of two machines which produce identical parts. Machine 1 (M1) has no delay. Machine 2 (M2) has delay. The total number of parts produced by both machines is measured against a constant demand to determine the scalar production surplus, x(t). The demand rate is 1 part per unit time. x(t) is therefore described

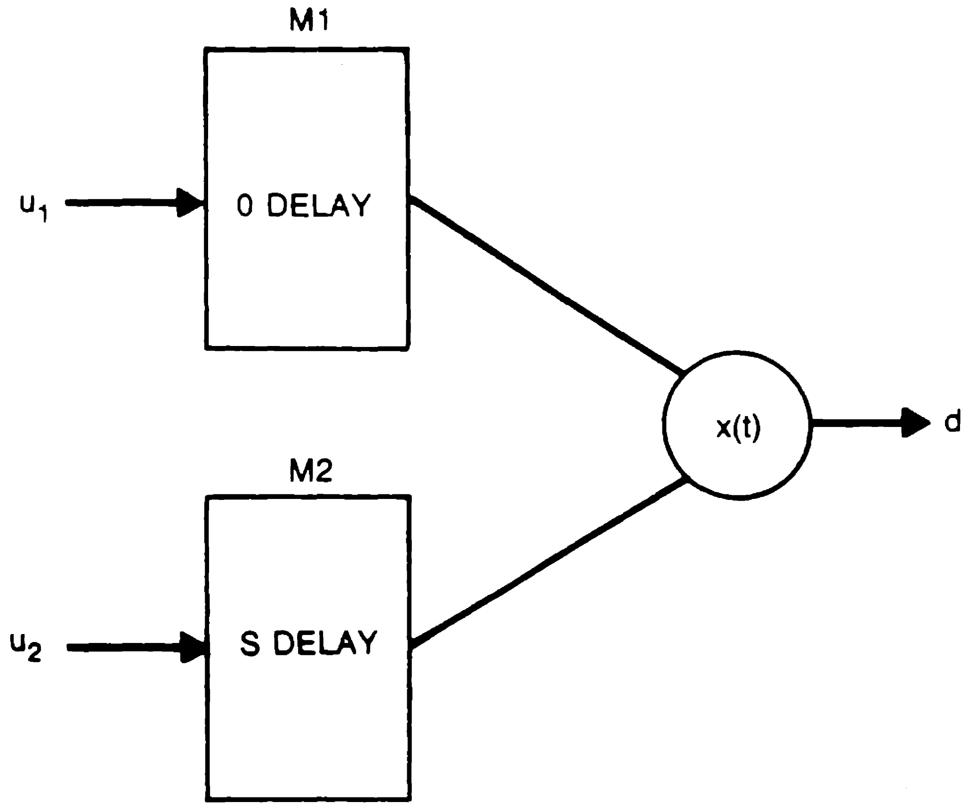


FIGURE 4.5 SIMULATION EXAMPLE 2

by

$$\dot{x}(t) = u_1(t) + u_2(t-s) - 1 \tag{4.66}$$

The maximum production rate of each machine is 2 parts per unit time. Both machines are unreliable. We denote the state of machine i by $\alpha_i(t)$, where $\alpha_i(t)=1$ if Machine i is working and $\alpha_i(t)=0$ if Machine i is broken. The probability of repair and failure are both 0.1 per unit time. Therefore,

$$P[\ \alpha_i(t+\delta t)=1 \ |\ \alpha_i(t)=0\ ] = 0.1\ \delta t \qquad i=1,2 \tag{4.67}$$

$$P[\ \alpha_i(t+\delta t)=0 \ |\ \alpha_i(t)=1\ ] = 0.1\ \delta t \qquad i=1,2 \tag{4.68}$$

The general form of the control is given by

$$\min_{\substack{0\le u_1\le\alpha_1 \\ 0\le u_2\le\alpha_2}} c_1u_1 + c_2u_2 \tag{4.69}$$

The following two policies were tested initially:

CONTROL Z: $c_1 = x(t) - x^1_h$

$c_2 = x(t) - x^2_h$

CONTROL C: $c_1 = x(t) - x^1_h$

$c_2 = \int_0^s u_2(t-\sigma)\, d\sigma + x(t) - x^2_h$

Again, Control Z corresponds to a zero-weighting-function policy. Parts are loaded if the x(t) is below the hedging point value x_h. Control C is a constant weighting function policy. The integral term is added to the coefficient c_2 and the sum is compared to the hedging point value x_h.

The controls were simulated over 10,000 time steps for a range of values of hedging points to determine the best values for each control. Delays of 10, 30 and 50 time units were used. Table 4.2 shows the simulation results. Again, the average inventory (INV) and backlog (BCK) as well as the parameter values that yielded the lowest total value of INV+BCK are shown.

DELAY	CONTROL Z				CONTROL C			
	INV	BCK	x^1_h	x^2_h	INV	BCK	x^1_h	x^2_h
10	2.2	4.3	2	2	2.0	2.2	1	8
30	3.9	6.2	2	2	2.8	3.2	2	19
50	5.0	8.8	2	2	4.6	3.6	4	31

TABLE 4.2 - Example 2 Simulation Results for Controls Z and C

As Table 4.2 shows, the constant weighting function control consistently outperformed the zero weighting function control. The average inventory levels are approximately the same for the two controls, but the backlog of control Z is significantly greater. Note that the relative INV and BCK difference for the two policies increases with the length of the delay.

This system was also animated graphically so that qualitative behavior of the controls could be observed. Control Z tended to react slowly to failures and also tended to overproduce upon recovery. The slow responsiveness of Control Z can be seen in Table 4.2 where the backlog, BCK, is greater for Control Z even though average inventory, INV, is also larger. The tendency for overproducing can be seen from large value of INV relative to the hedging point.

Again, one might expect this type of behavior. Control Z does not use any information on the state of material in the delay line. It is therefore slow to respond as the line empties and can over produce when the line is full. Control C measures the amount of material in the line and therefore less likely to under or over produce.

These results are not meant to be conclusive, but they do indicate that substantial improvements are possible when using even a simple weighting function control. In the next section, we suggest a more methodical approach to constructing delay controls.

4.5 A PREDICTOR HEURISTIC FOR DELAY SYSTEM CONTROL

In this section, a heuristic interpretation of the weighting functions of the quadratic approximation control is suggested. The heuristic is based on interpreting the terms in the coefficients of the linear program as predictions, or estimates, of the future values of the production states. The predictor is then used in a hedging point policy.

As a motivation for accepting such a heuristic, recall that the single FMS with delay that we examined in Section 3.3 had an optimal control that involved using the future production state, $x(t+s)$, in the optimal control for the non-delay system. Of course, in that case the future state was known exactly. Nevertheless, it does lend some credence to a predictor interpretation that it is truly optimal in at least one simple FMS case. Predictor controls are also common suboptimal strategies in the classical control literature (Smith 1957).

A predictor heuristic is analogous to the certainty equivalence principle of LQG control. In certainty equivalence controls, the problems of estimation and control are separated. The problem is solved as if perfect state information were known, and then an estimate of the state is used in the resulting control. A predictor heuristic differs in that one estimates a future state given the current state and a "noisy" (i.e. uncertain) future.

Two predictors were simulated. The first predictor uses a Markov model to approximate the production state. The second predictor uses the exact expected value of the production state.

4.5.1 Markov Predictor

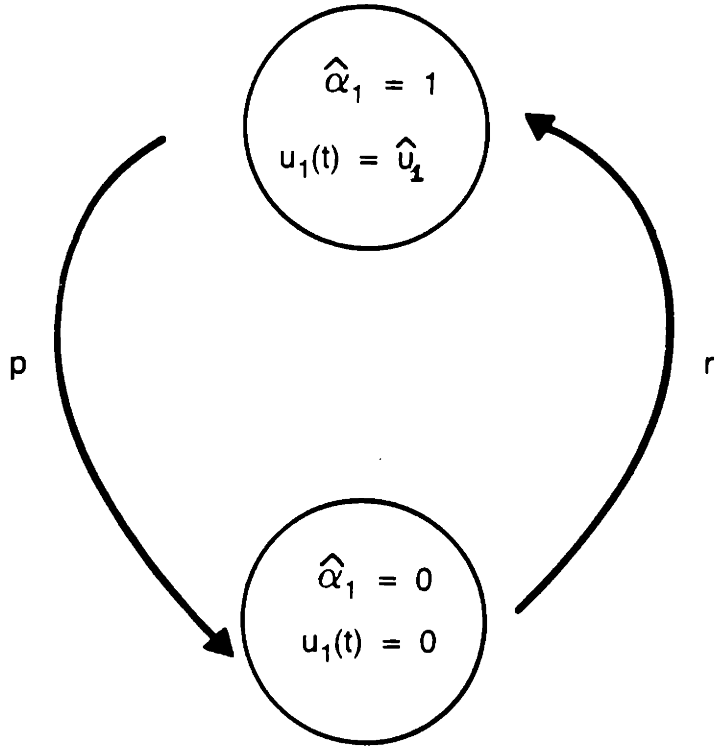
The two-FMS system shown in Figure 4.5 is used to demonstrate our predictor ideas. The current state of $x(t), \alpha(t)$ and u is used to predict the future state $x(t+s)$. We denote this estimate $\hat{x}(t+s)$. The values $x(t)$ and $\alpha(t)$ are then used in a simple hedging point policy.

To predict the value $x(t+s)$, note that its expected value is given by

$$E\ x(t+s) = x(t) + \int_0^s u_2(\xi - s)\ d\xi + E[\int_0^s u_1(t+\xi)\ d\xi] - sd \qquad (4.70)$$

The first two terms, which involve the current surplus and the past $u_2(t)$, are known. The third term is the expected total production of Machine 1 over the interval $[t, t+s]$, which is random and depends on future machine states and controls.

Now consider the simple Markov model of the first machine's behavior shown in Figure 4.6. When the first machine, M_1, is down, it produces nothing. When M_1 is operational, it produces at some average rate, which we define as $\hat{u}_1$.



$$\hat{u}_1 = \frac{p + r}{r} d$$

FIGURE 4.6 MARKOV ESTIMATOR

Let p_1 be the failure rate, r_1 is the repair rate of Machine 1 and β be the fraction of the demand we expect M_1 to satisfy (in the simulation, the machines are identical so we let $\beta=1/2$). We can then evaluate $\hat{u}_1$ by noting that the total long term production of Machine 1 is βd and the fraction of time it is up is r_1/r_1+p_1. Therefore, if we want our model to have the same average production, we must have

$$\hat{u}_1 \frac{r_1}{r_1+p_1} = \beta\ d \tag{4.71}$$

which yields

$$\hat{u}_1 = \beta\ d\ \frac{r_1+p_1}{r_1} \tag{4.72}$$

In reality, M_1 produces at a variety of rates which depend on $x(t)$, $\alpha(t)$ and the control parameters used. The above approximation ignores these details and uses only $\hat{u}_1$, the average production of M_1 when it is up. This approximation simplifies the calculation of M_1's expected production considerably.

Let $\hat{\alpha}_1(t)\epsilon\{0,1\}$ be the state of the Markov approximation. Define the usual probability distribution of this new process by

$$\pi(t)=[\pi_0(t) \quad \pi_1(t)] \tag{4.73}$$

where

$$\pi_1(t) = P[\ \hat{\alpha}_1(t) = 1\ ] \tag{4.74}$$

The infinitesimal generator for this process is

$$Q - \begin{bmatrix} -r & r \\ p & -p \end{bmatrix} \tag{4.75}$$

and therefore

$$\pi(t+\xi) - \pi(t)\ e^{Q\xi} \tag{4.76}$$

The expected total production of this new process over $[t,t+s]$ is therefore given by

$$\hat{E}(\hat{\alpha}_1) = \pi(t) \left[\int_0^s e^{Q\xi}\, d\xi \right] \begin{bmatrix} \hat{u}_1 \\ 0 \end{bmatrix} \tag{4.77}$$

where $\pi(t)=[1 \quad 0]$ if M_1 is down at time t and $\pi(t)=[0 \quad 1]$ otherwise.

Using (4.77) with $\hat{\alpha}_1$ replaced by α_1 for the expected total production of M_1 over $[t,t+s]$ gives the following estimate for $x(t+s)$:

$$\hat{x}(t+s) - x(t) + \int_0^s u_1(t-\xi)\, d\xi + \hat{E}(\alpha_1) - sd\ . \tag{4.78}$$

This estimate is crude but is very easy to compute and it reflects the change in expected production of the second FMS over [t,t+s] for different machine states. Note that the estimator $\hat{E}$ is a function of α_1 only. $\hat{E}$ can be computed off-line and therefore (4.78) is trivial to implement. This estimator can then be used in a simple hedging point policy, as demonstrated below in the Section 4.5.3.

4.5.2 Expected Value Predictor

A more sophisticated predictor can be constructed for the Figure 4.5 example by numerical methods. Given a hedging point, the possible sample paths for a given state surplus, x(t), and machine state, α(t), can be enumerated. The expected value of x(t+s), as given by (4.), can be computed exactly from this enumeration. We call this type of predictor an <u>expected-value predictor</u>

For complicated systems, such a predictor would not be practical. It requires a combinatorial calculation that must be performed on-line for each x(t) and α(t) or else performed off-line and stored in a large table. Even though it is not practical, we can use the expected-value predictor as a benchmark for evaluating the simpler Markov predictor of the previous section.

4.5.3 Simulation of Predictor Controls

The Markov estimator was tried on the example in Figure 4.2. We also ran simulations using the expected-value predictor for comparison. As before, the simulation was tested for a range of hedging point values and the hedging point that yielded the lowest total inventory and backlog (INV+BCK) was selected.

The Markov-predictor control takes the form,

$$\text{CONTROL M: } c_1 = x(t) - x^1_h$$

$$c_2 = x(t) + \int_0^s u_2(t-\sigma)\, d\sigma$$

$$+ \hat{E}(\alpha_1) - sd - x^2_h$$

while the expected-value control takes the form

$$\text{CONTROL E: } c_1 = x(t) - x^1_h$$

$$c_2 = x(t) + \int_0^s u_2(t-\sigma)\, d\sigma$$

$$+ E(\int_0^s u_1(t+\sigma)\, d\sigma) - sd - x^2_h$$

$\hat{E}(\alpha_1)$ is the estimate of Machine 1's production over the interval [t,t+s] based on the Markov predictor, given by (4.77). The integral $E(\int_0^s u_1(t+\sigma)d\sigma)$ is the true expected value computed by sample path enumeration and c_1 and c_2 are the coefficients of in the minimization (4.69) as before. The results of the simulations are shown in Table 4.3 below.

DELAY	CONTROL M				CONTROL E			
	INV	BCK	x^1_h	x^2_h	INV	BCK	x^1_h	x^2_h
10	2.8	3.1	4	3	1.6	2.9	2	2
30	2.6	3.9	3	4	3.4	2.5	3	2
50	3.9	3.0	4	4	3.5	3.0	3	3

TABLE 4.3 - Simulation Results for Controls M and E

The data in Table 4.3 show that for both Control M and Control E, the average inventory and backlog values do not increase much when the delay is increased. Both controls also performed better than Control Z and Control C for the 50 unit delay case. The difference between the Markov predictor and the actual expectation appears to be minimal. Considering the computational burden of computing the actual expectation, this small difference suggests that even a crude predictor can yield good results.

One problem with the Markov estimator is that it does not consider the location of material in the delay line. If material in the the first FMS is clustered at the beginning of the delay line (i.e. most of the material has been loaded recently) then we would expect that much of this material would still be in the buffer at time $t+s$.

Conversely, if most of the material is clustered toward the end of the delay line, then we would expect that more of the material will be processed by time t+s. By this reasoning, it may be better to choose a weighting function $f(\xi)$ that is decreasing in ξ. The estimate would then take the form,

$$\hat{x}(t+s) = x(t) + \int_0^s f(\xi)\, u_2(t-\xi)\, d\xi + \hat{E}(\alpha_l) - sd \tag{4.79}$$

Choosing such a weighting function may be difficult, however, and, as shown in Table 4.3, even an exact expected value does not improve the performance significantly. As in any heuristic strategy, fine tuning via simulation and practical experience will be necessary to adjust this estimator.

4.6 DISCRETE TIME DYNAMIC PROGRAMMING AND DELAY APPROXIMATIONS

We mentioned in Section 4.1 that the delay approximation technique would generate good estimates of the optimal value function if analytical or computational techniques existed for solving the m'th order problem. One technique that can be used to solve this problem is the value iteration algorithm of discrete-time dynamic programming (Bertsekas 1976). In this section, we show how the delay approximation technique can be used to reduce the number of states needed in a discrete-time dynamic programming solution.

Consider the one-machine, random demand example of Section 4.4.1. Assume that time has been discretized so that the delay through the system is T units. Suppose also that there are n discrete levels of control, u, and k total (x,α) states. The total number of states for this discretization is then

$$n^T k \tag{4.80}$$

The number of delay states can get very large. For example, if n=4 and T=16 then the number of states is

$$4.29\times10^9 k \tag{4.81}$$

which is too large to store in a computer.

Consider, however, a first order delay approximation to this system. Recalling that the approximation variable, z_1, approximates the average amount of material in the system, the number of states needed for m=1 is only of order nT. For the numbers give above, this gives 4x16=64 states, a considerably smaller number.

Of course, one would not expect a first order approximation to be sufficient in many cases. In general, a m'th order approximation would yield

$$\left[\frac{nT}{m}\right]^m k \tag{4.82}$$

total states.

The dimensionality of all but the lowest order approximations quickly become too large to be useful. There may be cases, however, where first or second order approximations may yield acceptable controls. For example, the constant weighting function predictor of the last section could be generated from a first order approximation by using policy iteration techniques.

In practice, the large number of buffer level and machine states required for realistic problems would make solving even a first order approximation impractical. We must, therefore, devote our attention to understanding the heuristic strategies of the previous section if truly useful controls are sought. To develop the heuristic strategy further, the next chapter examines the problem of control in interconnected systems without delay.

CHAPTER 5 - INTERCONNECTED WORKSTATIONS

In this chapter, we investigate the control of a tandem, two-machine system without delay. There are three primary reasons for doing so: First, as we show in Section 5.1, the instantaneous capacity restriction can limit throughput, so subdividing a system may be desirable. Second, as demonstrated in Chapter 3, systems with delay may also have to be partitioned in order to reduce internal inventories. And third, the problem is interesting on its own because it leads to valuable insights into the control of more complex systems.

In Section 5.2, the notion of Production Surplus (PS) in interconnected systems is developed. The PS is simply an extension of the production variable $x(t)$ we have been using thus far. Representing interconnected systems in a production surplus space allows for a more intuitive understanding of the behavior of an interconnected system. The analysis of the two-machine system in the following sections is described using the production surplus.

Numerical calculations of the optimal control policy for a simple, tandem two-machine, one-part-type system are analyzed in Section 5.3. The data suggest that for the second machine, a policy based only on a production surplus hedging point is optimal. For the first machine, the optimal policy can be characterized by two regions in surplus space. In one region, the system behaves approximately like a transfer line with a finite internal buffer. In the other region, the system behaves approximately like two isolated machines both independently following a PS hedging point policy.

Based on these observations, a sub-optimal strategy is suggested in Section 5.4 that use a buffer hedging point in one region of the surplus space and a surplus hedging point in the other region. We call this policy a Two Point control because it can be completely characterized by the two hedging point values. A method for calculating the parameters of the control is developed. For an important class of problems, namely for systems that are heavily utilized, the control regions of this sub-optimal policy are very close to the optimal control regions.

5.1 THE INSTANTANEOUS CAPACITY RESTRICTION

An important assumption in the Kimemia-Gershwin model is that no material is allowed to build up between work stations. This assumption is the basis of the instantaneous capacity set constraint. Requiring the control to satisfy the instantaneous capacity $\Omega(\alpha)$ at all times can reduce a system's capacity.

For example, if the manufacturing system consists of a series of in-line machines and some part type must visit all the machines, then if any machine is down, the production of that part-type must be stopped. If we allow internal storage, parts can be loaded even though downstream machines are not operational. The parts then accumulate in internal buffers while downstream machines are under repair. Internal buffers also allow a machine to work while its upstream neighbor is broken. It is therefore possible to increase the utilization of machines and hence the capacity by using internal buffers and relaxing the instantaneous capacity restriction.

There is, however, an absolute maximum capacity that must always be satisfied. Suppose part types are denoted n=1,2,..N as in the Kimemia-Gershwin model of Chapter 2. As in Chapter 2, let τ_{nm} be the processing time of part n on work station m and α_m be the number of operational machines in work station m. Let $\bar{u}$ be the N-vector of average production rates. In the Chapter 2 formulation, we would have $\bar{u}$=d. The average number of parts being processed by work station m is therefore

$$\sum_n \bar{u}_n \tau_{nm}. \tag{5.1}$$

The average number of parts being processed by work station m can not exceed the average number of working machines at workstation m. Therefore, if we let $\bar{\alpha}_m$ denote the average number of working machines at work station m, we must have

$$\sum_n \bar{u}_n \tau_{nm} \le \bar{\alpha}_m . \tag{5.2}$$

Equation (5.2) must be true for all work stations in the system. Therefore, we define the <u>absolute capacity</u> of a FMS by

$$\bar{\Omega} = \{ \bar{u} \mid \sum_{n} \bar{u}_n \tau_{nm} \le \bar{\alpha}_m, \bar{u} \ge 0 \quad \text{for all } m \}. \qquad (5.3)$$

The set $\bar{\Omega}$ defines production rates that can be achieved if machines can be completely utilized, that is, if some machines are never allowed to be idle. This is why be call $\bar{\Omega}$ the absolute capacity.

There are control policies which satisfy $\bar{u}\epsilon\bar{\Omega}$ but which do not satisfy $u(t)\epsilon\Omega(\alpha(t))$ for all t. For example, take the case of two workstations in series, each of which has only one machine. This system is shown if Figure 5.1. Each machine is capable of a maximum production rate of one part per unit time and the machines have identical failure and repair statistics. Failures and repairs are Markovian. Let p be the failure rate for the machines and r be the repair rate. The maximum throughput of each machine is then

$$\frac{r}{r+p} \, . \qquad (5.4)$$

Therefore, if a large internal buffer is inserted between the machines, the maximum throughput of the FMS approaches r/(r+p).

If the instantaneous capacity restriction is imposed, the FMS can only produce while both machines are up. Assuming machines only fail while processing parts, the fraction of time both machines are up is (Buzacott 1982)

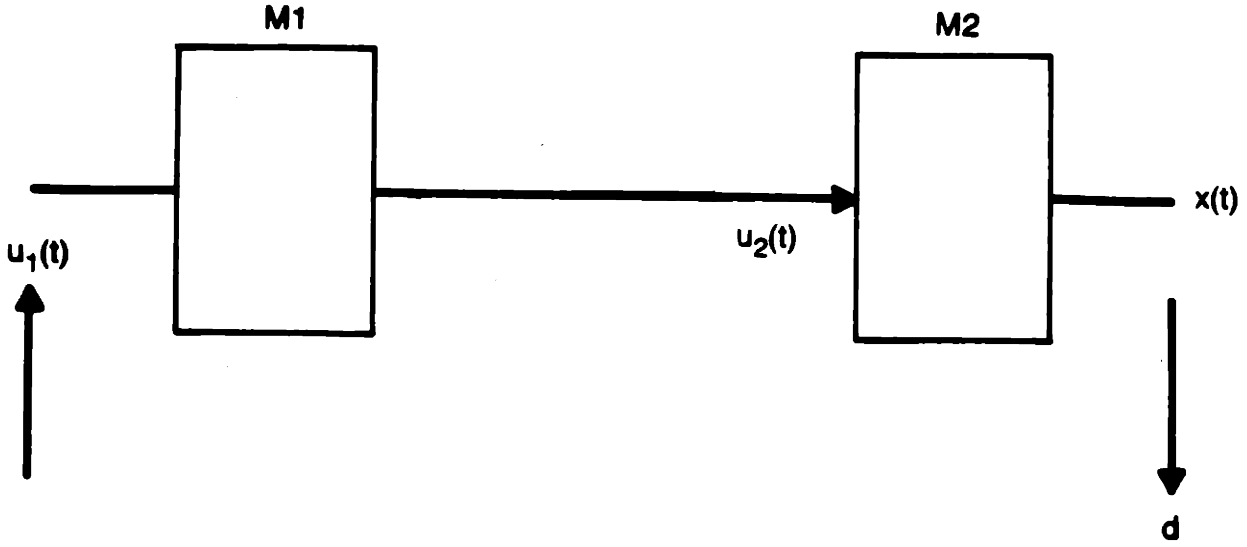


FIGURE 5.1 TWO MACHINE FMS

$$\frac{r}{r + 2p} \cdot \qquad (5.5)$$

which is less than r/(r+p). Therefore a demand d satisfying

$$\frac{r}{r + 2p} < d < \frac{r}{r + p} \qquad (5.6)$$

is feasible because it is contained in $\bar{\Omega}$ but cannot be achieved if the instantaneous capacity restriction is imposed.

The instantaneous capacity restriction, however, does guarantee that no material builds up in a FMS. Because u(t) is feasible at each work station in the system, the production rate at the input of any work station will equal the production rate at its downstream neighbor and therefore material will not accumulate between workstations.

The fact that material is prevented from building up when the instantaneous capacity restriction is imposed makes the control especially desirable when congestion in the transportation system and workstation buffers is critical. For example, it would be appropriate for an FMS whose work stations are interconnected by an automated materials handling system with limited space (e.g. a fixed number of pallets) or for any system where the work in progress (WIP) is to be kept at a minimum. Indeed, these are precisely the types of systems that the basic model of Chapter 2 was intended for.

For large systems where some inventory between workstations is acceptable, the instantaneous capacity constraint may unnecessarily restrict throughput. Subdividing the control of the system may then be necessary. In the remaining sections we investigate the control of a simple system with internal buffers as a first step toward understanding the control of a subdivided FMS.

5.2 PRODUCTION SURPLUS SPACE

5.2.1 Definition of Production Surplus

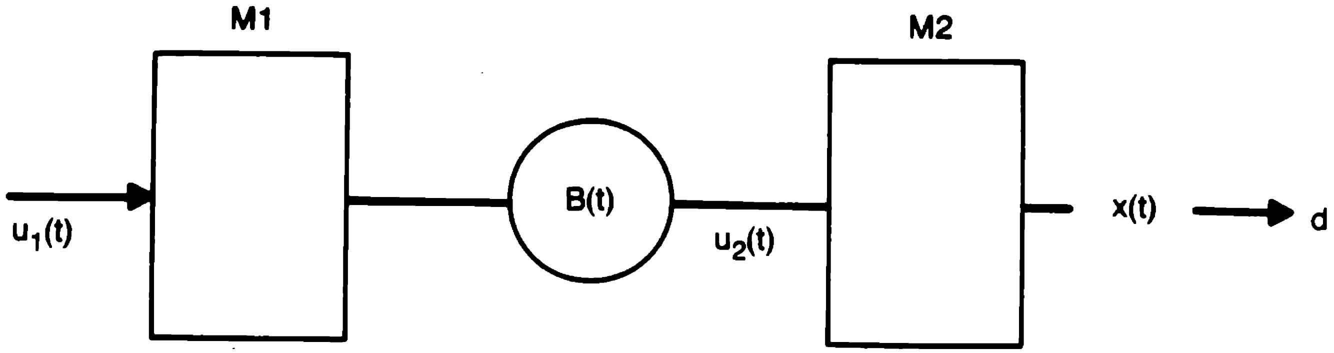
In this section, we introduce the concept of Production Surplus Space to describe the control of interconnected systems. We will use the single-part-type, two-machine system shown in Figure 5.2 to develop our production surplus ideas.

Description of Figure 5.2 System

The system of Figure 5.2 consists of two machines connected in series. The machines are separated by a physical buffer, whose level, $B(t)$, is constrained to be non-negative:

$$B(t) \geq 0 \tag{5.7}$$

The production rate of Machine i is denoted $u_i(t)$. There is a constant demand, $d=1$ part per unit time, and both machines can produce at a maximum rate of 2 parts per unit time. Machines are subject to Markovian failures and repairs. The failure rate, p, is the same for both machines. The repair rate, r, is also the



FAILURE RATE = p
REPAIR RATE = r
DEMAND = 1
MAXIMUM PRODUCTION =2

FIGURE 5.2 NUMERICAL EXAMPLE

same for both machines. As before, we denote the state of Machine i by $\alpha_i(t)$, where $\alpha_i(t)=1$ if Machine i is operational and $\alpha_i(t)=0$ if Machine i is not operational.

Desirable behavior for the system is to meet the demand rate, d, without overproducing, underproducing or carrying large quantities of material in the buffer. Our goal is to come up with a control that produces such behavior. In Section 5.3.1 we give exact definitions of cost functions designed to produce this type of behavior.

Production Surplus

We define the production surplus (PS) at Machine i by

$$\dot{x}_i(t) = u_i(t) - d \qquad i = 1,2 \tag{5.8}$$

where $u_i(t)$ is the production rate of Machine i at time t.

The PS at a machine is the difference between what a machine has produced up to time t minus the cumulative demand up to time t. Note that the production surplus at any stage is the state we would be using if we isolated each machine and analyzed it as in Chapter 2.

We make the following zero-initial-condition assumptions concerning $x_i(t)$ and $B(t)$,

$$x_i(0) = 0 \qquad i=1,2 \tag{5.9}$$

$$B(0) = 0$$

The buffer level, B(t), is given by

$$\dot{B}(t) = u_1(t) - u_2(t) \tag{5.10}$$

From (5.8), (5.10) and the initial condition assumptions, (5.9), we have

$$B(t) = x_1(t) - x_2(t), \tag{5.11}$$

and also

$$x_1(t) \geq x_2(t) \ . \tag{5.12}$$

The second relation is true because of the physical buffer constraint (5.7).

5.2.2 Interpreting Controls in Surplus Space

We can completely represent the state of the system using $x_i(t)$ space and the machine states. An example of the PS space for the two-machine system of Figure 5.2 is shown in Figure 5.3. This figure shows the state space and control regions when both machines are operational $(\{\alpha_1,\alpha_2\}=\{1,1\})$. There is a different division of the control space for all four combinations of machine states (α_i=0 or 1 for i=1,2). We will examine the regions for other machine states in detail in Section 5.3.2 when we discuss the numerical examples.

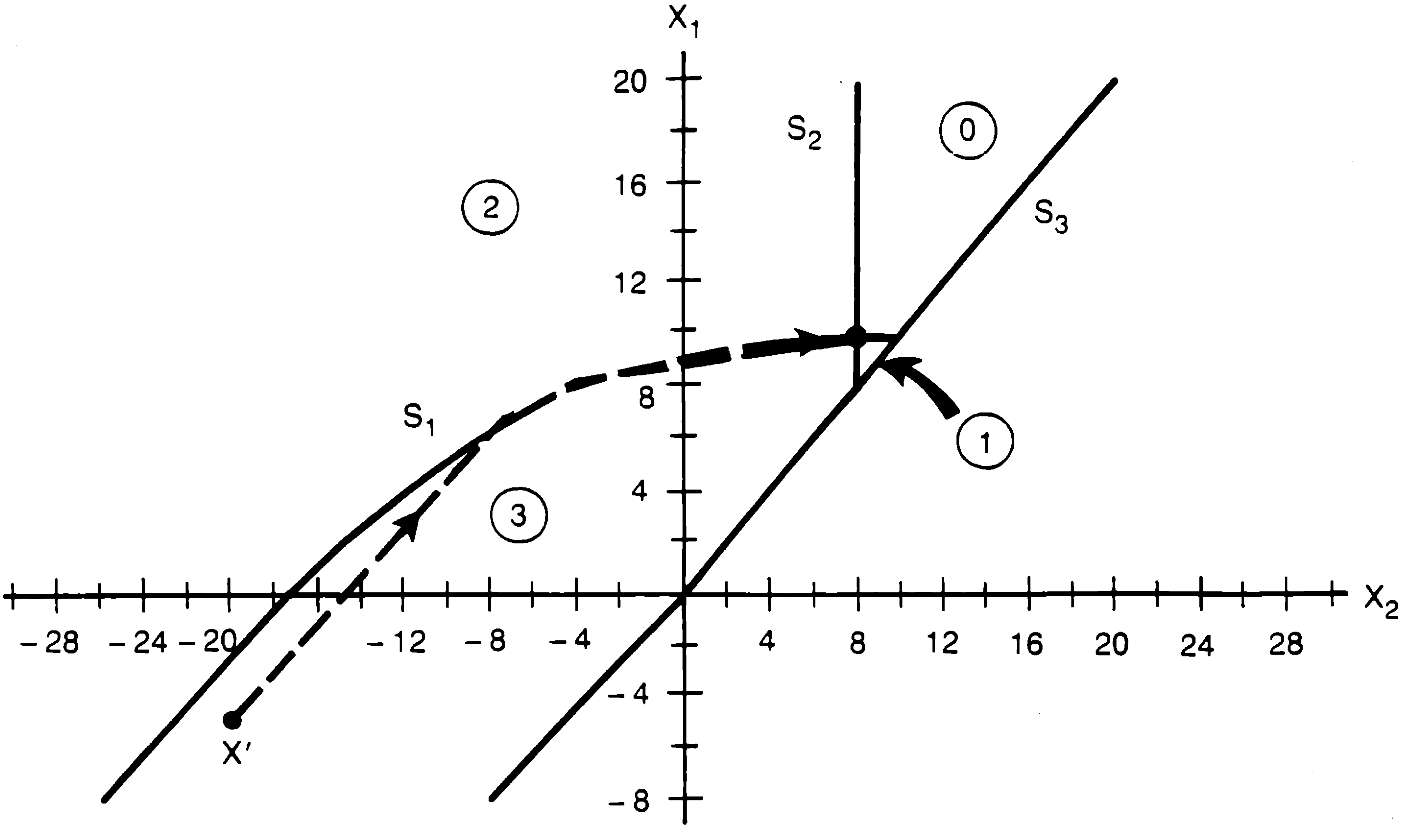


FIGURE 5.3 EXAMPLE SURPLUS SPACE ($\alpha_1 = \alpha_2 = 1$)

In the next section, we show how these regions are generated. For now, we concentrate on understanding the the features in Figure 5.3.

The regions are subsets of the space where a particular control action is optimal. In Region 1, the optimal control is for Machine 1 to produce at its maximum rate of 2 parts per unit time, which is greater than the demand of 1 part per unit time, and for Machine 2 to produce nothing. In Region 2, the optimal control is for Machine 2 to produce at the maximum rate, while Machine 1 produces nothing. In Region 3, both machines should produce at the maximum rate, while in Region 0 neither machine should produce.

The boundary S_3 is the $x_1=x_2$ line and is due to the physical buffer constraint. From (5.7) only states with $x_1(t)\geq x_2(t)$ are allowed and thus only the states above S_3 are physically possible. A state on the S_3 boundary corresponds to a state with a zero internal buffer level. In general, the buffer level is the vertical distance from a point to the S_3 boundary.

The boundaries S_1 and S_2 separate the feasible region of x-space. Below the boundary S_1 (Regions 1 and 3), it is optimal for Machine 1 to produce. Above S_1 (Regions 0 and 2), it is optimal for Machine 1 not to produce. Likewise, to the left of S_2 (Regions 2 and 3) it is optimal for Machine 2 to produce, while to the right of S_2 (Regions 0 and 1) it optimal for Machine 2 not to produce.

On a boundary, the control has a more complex behavior. Akella, Choong and Gershwin (1984), studying a simpler problem, show that along boundaries like S_1 and S_2, which are called attractive, the optimal control is to follow the boundary. To do otherwise results in trajectories that chatter back and forth from one region to the other. We assume a similar behavior takes place along the boundaries S_1 and S_2, and give an example of a trajectory below. The interested reader should refer to Akella, Choong and Gershwin (1984) for a more detailed discussion of chattering and boundary behavior.

The intersection of S_1 and S_2 is the hedging point. Perturbing the system at this point will put it into a region that drives the system back to the hedging point. It is therefore a stable point toward which the system tries to work. It represents the optimum buffer and surplus levels for hedging against future failures.

As an example of how the boundaries affect a trajectory, consider a trajectory starting at x' in Figure 5.3. If both work stations have sufficient capacity, the trajectory will move toward the positive orthant (both Machine 1 and Machine 2 producing at rate 2) as shown. In this example, the direction initially keeps the internal buffer level constant, because both machines are producing at rate 2, which is greater than the demand of 1.

When the trajectory encounters the S_1 boundary it changes directions, allowing the internal buffer level to decline. The boundary S_1 determines whether or not M1 builds buffer stock. If x(t) is below S_1, M1 produces parts which get added to the buffer. If x(t) is below S_1, M1 stops replenishing the buffer. We will, therefore, call the vertical distance from S_1 to S_3 at some value of x_2 the optimal buffer level at x_2.

When the trajectory reaches the hedging point, the system produces at the demand rate ($x_1(t)$ and $x_2(t)$ are constant) until a machine fails and the regions change.

5.2.3 Examples of Two Heuristic Policies in PS Space

We now consider how two possible sub-optimal strategies for controlling the system in Figure 5.3 would appear in PS space. Figure 5.4 shows a pure PS hedging point policy for the case where both machines are operational ($\alpha_1=\alpha_2=1$). This policy corresponds to treating the two machines as independent and with constant demand as in Section 2.1. The effect of the buffer is ignored and a single surplus-hedging-point value is used for each machine. The policy could be refined further by allowing the hedging points to be functions of α, but we assume for illustration in Figure 5.4 that the hedging points are independent of α.

Note that the optimal buffer level in this policy increases without bound as x_2 decreases. Near the hedging point, the optimal buffer level is smallest. Far to the left of the hedging point, the optimal buffer level is large. Note that decisions in

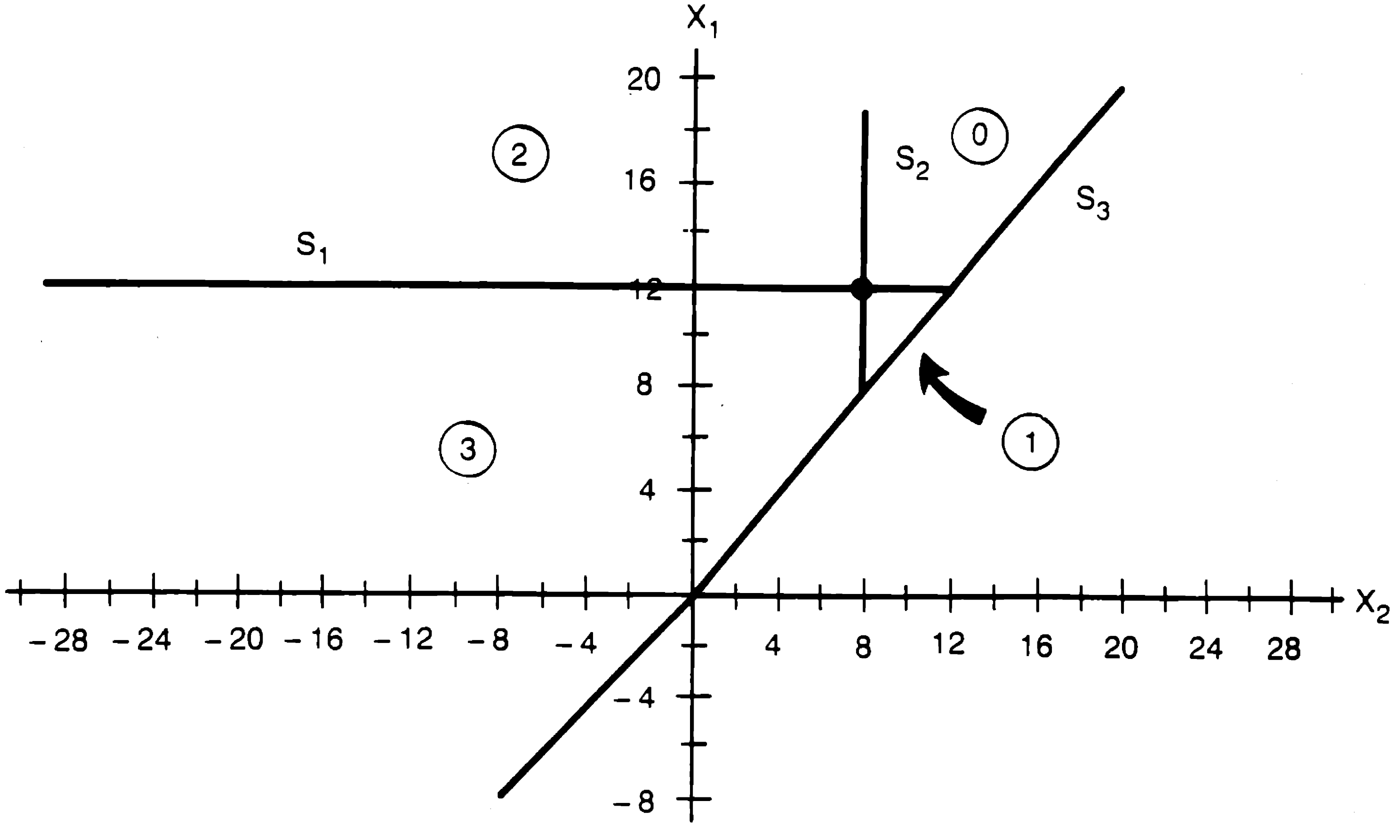


FIGURE 5.4 PURE PS POLICY ($\alpha_1 = \alpha_2 = 1$)

this policy are independent: Each machine is loaded based only its own surplus. The buffer level does not affect the control, except at the boundary S_3.

Another possible policy is shown is Figure 5.5. This policy is a pure buffer policy. The decision of the first machine depends only on the buffer level and not on the values of x_1 or x_2. The decisions of the second machine depend only on x_2 as before. Again, the policy could be further refined to allow the hedging points to be functions of α, but we assume in Figure 5.5 that this is not the case. In this policy, the optimal buffer level is independent of the production surplus, $x_1(t)$.

5.3 NUMERICAL CALCULATIONS OF OPTIMAL CONTROLS

In the previous section we examined the characteristics of a PS space representation and examined a pure PS policy and a pure buffer policy. We now examine the results of some numerical calculations of optimal policies for the two-machine, one-part-type system of Figure 5.2.

5.3.1 Discrete Dynamic Programming Formulation and Solution

To compute true optimal controls for the simple two-machine, one-part-type system, the value iteration algorithm of dynamic programming (DP) was used on a discretized version of the

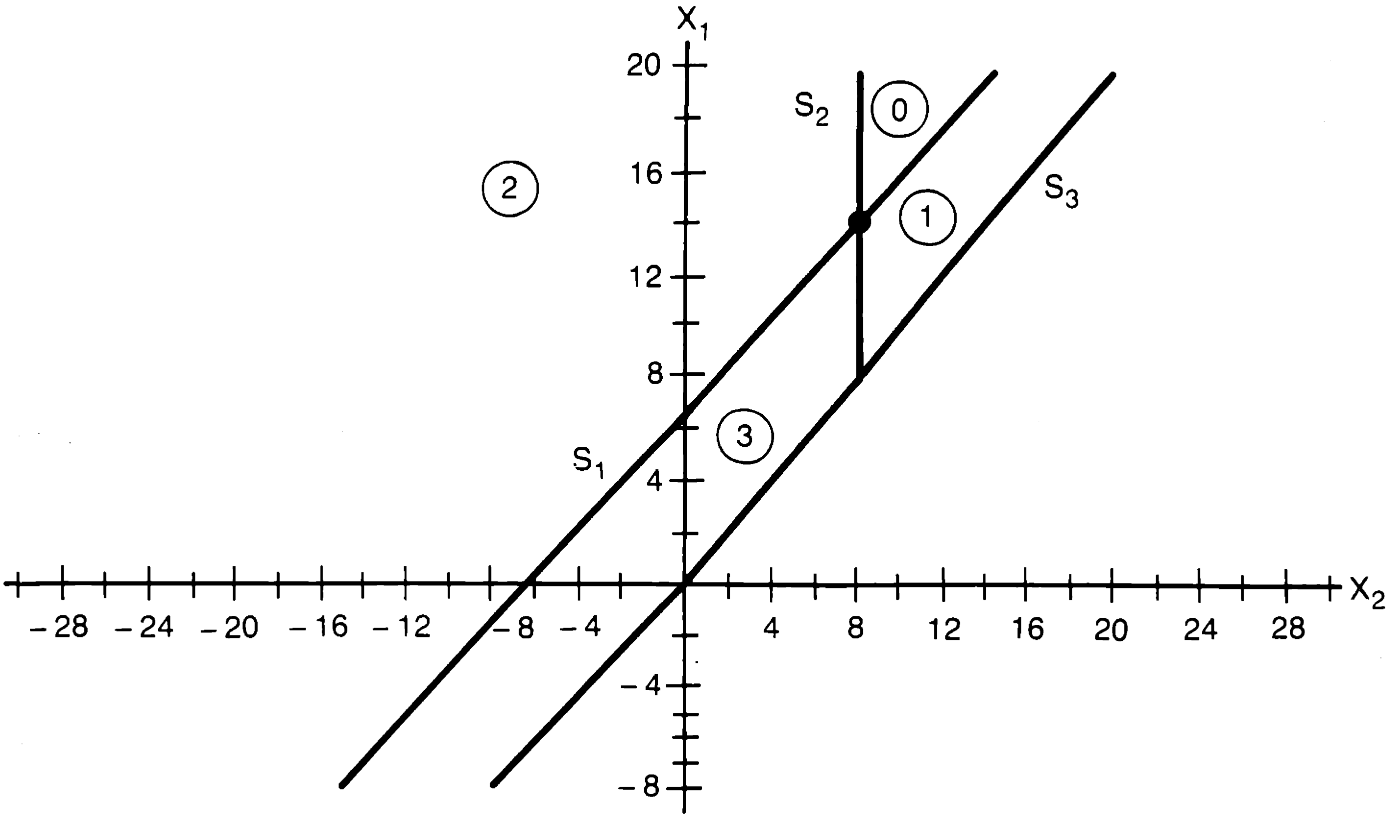


FIGURE 5.5 PURE BUFFER POLICY ($\alpha_1 = \alpha_2 = 1$)

problem in Section 5.2. The source code and a brief description of the algorithm are given in Appendix B. A complete discussion of the details of the algorithm is presented in Bertsekas (1976).

A discretized version of the surplus equation (5.8) for the system in Figure 5.2 is given by

$$x_1[t+1] = u_1[t] - d \tag{5.13}$$

$$x_2[t+1] = u_2[t] - d \ , \tag{5.14}$$

where the demand $d=1$.

The buffer level is

$$B[t] = x_1[t] - x_2[t]. \tag{5.15}$$

The failure and repair transitions are

$$P[\ \alpha_i[t+1] = 0 \mid \alpha_i[t] = 1\ ] = p, \qquad i=1,2 \tag{5.16}$$

$$P[\ \alpha_i[t+1] = 1 \mid \alpha_i[t] = 0\ ] = r, \qquad i=1,2\ . \tag{5.17}$$

The constraints on $u_i[t]$ are then

$$0 \le u_i[t] \le 2\alpha_i[t], \qquad i=1,2. \tag{5.18}$$

The cost function used in the algorithm is the piece-wise linear cost,

$$g_1(B) = C_1^+ B \tag{5.19}$$

$$g_2(x_2) = C_2^+ x_2^+ + C_2^- x_2^- . \tag{5.20}$$

where $x_2^+ = \max\{0, x_2\}$ and $x_2^- = \max\{0, -x_2\}$. This cost function penalizes the system for overproducing ($x_2 > 0$) or underproducing ($x_2 < 0$), and also penalizes the system for carrying internal buffer stock. The penalty thus enforces the desirable behavior mentioned above, namely to meet the demand rate of 1 part per unit time, without a surplus and without carrying material in the buffer. Failures of the machines mean that some compromise in this ideal behavior is necessary.

The value iteration algorithm requires that we use the discounted, infinite-horizon cost

$$\int_0^\infty e^{-\beta t} [\ g_1(B(t)) + g_2(x_2(t))\ ]dt. \tag{5.21}$$

Since our objective is to get desirable behavior out of the system and not to control actual physical costs, a discounted cost is justifiable. Also, for $\beta \rightarrow 0$, (5.21) does approach the total cost. We let $\beta = 0.1$ and then minimized the cost (5.21) subject to the constraints (5.13)-(5.18). Various values of p, r, C_1^+, C_2^+ and C_2^- were tested.

The algorithm was implemented in PASCAL on a micro-computer. A single example takes on the order of five hours of processing time to converge on a 6 Mhz IBM AT computer.

Table 5.1 shows the three series of examples. In Series A, the parameters p and r, are fixed as are the inventory and backlog cost C_2^+ and C_2^-. The penalty, C_1^+, on the internal buffer stock is varied. In Series B, the repair rate, r, is varied for the case where the penalty on the internal buffer, C_1^+, is less than the penalty on the inventory, C_2^+. In Series C, the repair rate is again varied, but for the case where $C_2^+ > C_1^+$. In the next section, we discuss these cases in detail.

TABLE 5.1 Cases For Numerical Optimal Controls

CASE	p	r	C_1^+	C_2^+	C_2^-
A1	0.1	0.18	0.1	1.0	10.0
A2	0.1	0.18	0.5	1.0	10.0
A3	0.1	0.18	1.0	1.0	10.0
A4	0.1	0.18	2.0	1.0	10.0
B1	0.1	0.5	2.0	1.0	10.0
B2	0.1	0.2	2.0	1.0	10.0
B3	0.1	0.15	2.0	1.0	10.0
B4	0.1	0.12	2.0	1.0	10.0
C1	0.1	0.5	1.0	2.0	10.0
C2	0.1	0.2	1.0	2.0	10.0
C3	0.1	0.15	1.0	2.0	10.0
C4	0.1	0.12	1.0	2.0	10.0

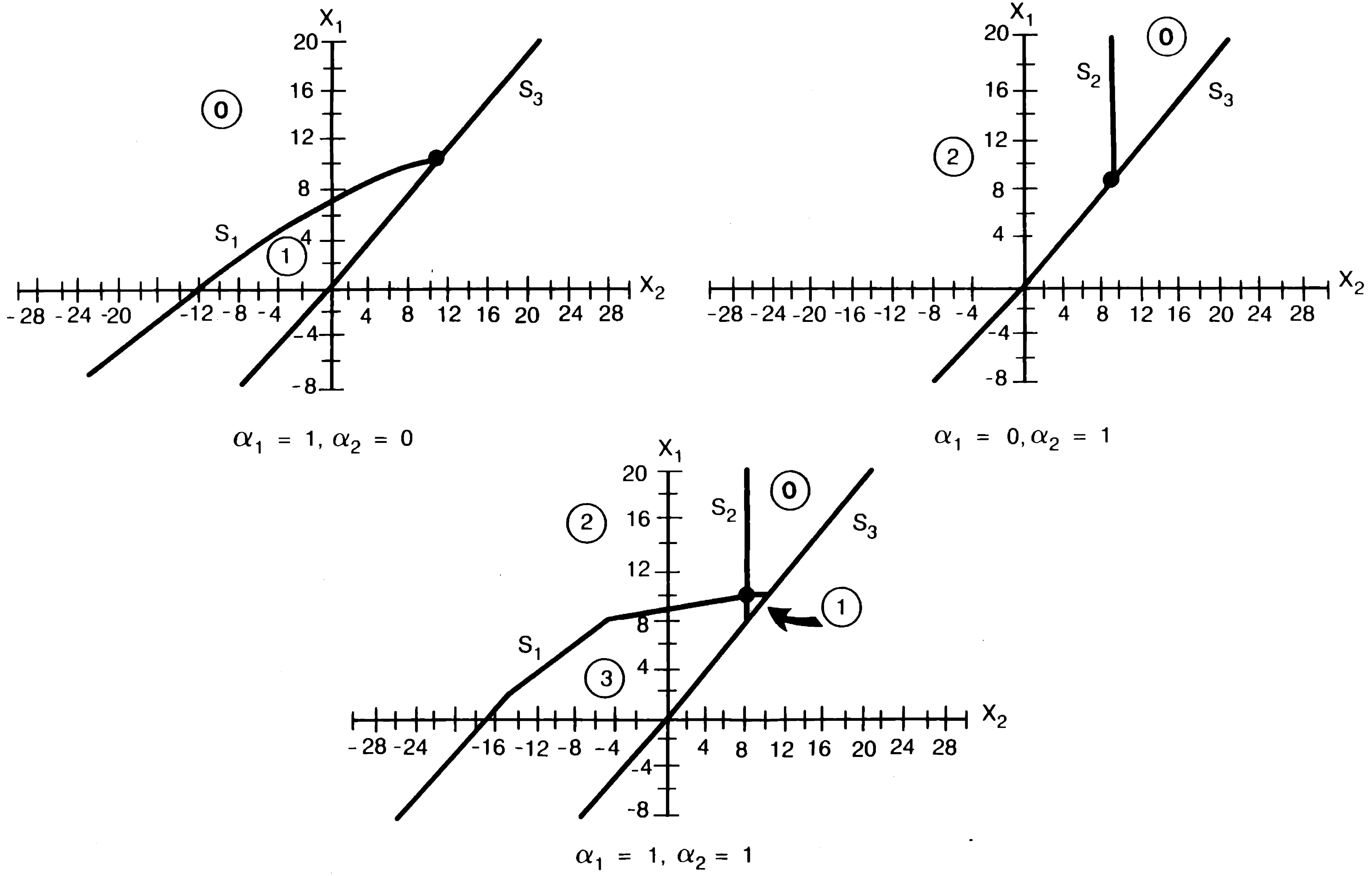
5.3.2 Control Regions for Different Machine States

Figure 5.6 shows the complete set of control space regions for all machine states of example A2. This is the example that generated the regions given earlier in Figure 5.3. We omitted the graph for $\alpha_1=\alpha_2=0$ (both machines down) since no control action is possible in this case: The whole state space is Region 0.

Note that the hedging point, the dark dot at the intersection of S_1 and S_2, moves out away from the origin only slightly when either of the machines is down. The S_2 boundary also shifts only slightly (from $x_2=8$ to $x_2=9$) when $\alpha_1=0$, $\alpha_2=1$. In most of the examples, the hedging point and S_2 boundary change only slightly for different machine states.

The S_1 boundary does move closer to the S_3 boundary when $\alpha_1=1, \alpha_2=0$. This too was true of all the examples. Recalling that the vertical distance from S_1 to S_3 is the optimal buffer level, this suggests that we actually want less material in the buffer when Machine 2 is down than when it is up.

For example, if both machines are up ($\alpha_1=1$, $\alpha_2=1$ in Figure 5.6), at the point $x_1=8$, $x_2=0$ and Machine 2 suddenly fails, then we are immediately transported to the graph $\alpha_1=1$, $\alpha_2=0$ with the same x values. Machine 1 will then stop producing. When x_1 and x_2 decline enough to hit the S_1 boundary, Machine 1 will begin producing and only then will the buffer begin to rise.



$\alpha_1 = 1, \alpha_2 = 0$

$\alpha_1 = 0, \alpha_2 = 1$

$\alpha_1 = 1, \alpha_2 = 1$

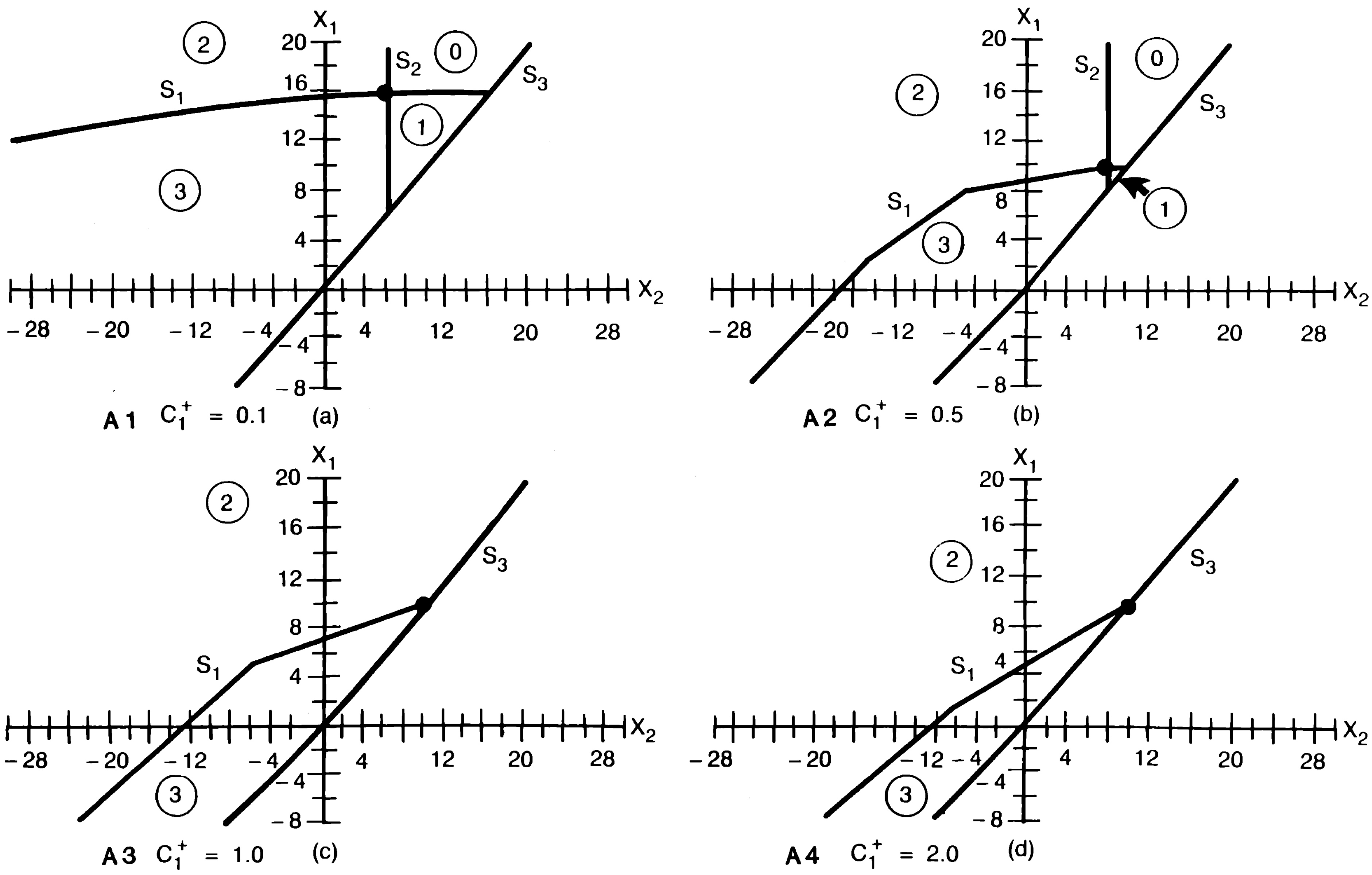
FIGURE 5.6 EXAMPLE A2

Intuitively, this makes sense if we consider that the most likely future when Machine 1 is up and Machine 2 is down is for the machines to stay in their current states. If Machine 2 does suddenly come up, it is likely that Machine 1 will still remain up and able to supply it. If Machine 1 fails, it is likely that Machine 2 will still be down and the buffer stock will not be necessary. Only if Machine 2 comes up shortly followed by Machine 1 going down, an unlikely sequence, will the buffer stock be critical. By contrast, if both machines are up, the buffer will be needed any time Machine 1 fails.

For the remainder of this chapter, we will concentrate on characterizing the control regions for the case where both machines are operating ($\alpha_1 = \alpha_2 = 1$). We do this because our primary interest is in the gross shape and approximate position of the boundaries, which do not seem to change significantly from machine state to machine state. Also, when $\alpha_1 = \alpha_2 = 1$, all the boundaries are present. The general changes in the location of boundaries for different machine states outlined above should, however, be kept in mind as we examine the numerical cases below.

5.3.3 Control Regions for Series A

The control regions for examples A1-A4 are shown in Figure 5.7. Only the graphs for $\alpha_1 = \alpha_2 = 1$ are shown. The boundaries for the other machine states varied slightly from those in Figure 5.7, as discussed above. In this series of examples, only the



A 1 $C_1^+ = 0.1$ (a)

A 2 $C_1^+ = 0.5$ (b)

A 3 $C_1^+ = 1.0$ (c)

A 4 $C_1^+ = 2.0$ (d)

FIGURE 5.7 SERIES A

cost of the internal buffer, C_1^+, was varied. This series reveals some interesting insights into the optimal hedging point and S_2 boundary.

Note that as C_1^+ is increased S_1 becomes closer to being parallel to S_3 and the hedging point moves closer to S_3. This suggests that at the hedging point, the optimal buffer level is essentially zero unless C_1^+ is significantly less than C_2^+. That is, unless the buffer penalty is low, the buffer is only used while the system is behind.

If we think of the buffer stock as a resource that increases the output of the system, this means it is optimal to use the buffer resource to increase output only while the system is catching up, but once the system catches up, this resource is too expensive to continue using. We see that only for examples A1 and A2, where C_1^+ is much less than C_2^+, is the buffer level at the hedging point significantly greater than zero.

The boundary S_2, when it exists, is vertical. (We examine the occurrence and disappearance of this boundary shortly.) In all the cases examined that have an S_2 boundary, it is always vertical. This indicates that Machine 2 follows a pure PS-hedging-point policy.

This makes intuitive sense. The buffer stock exists only to supply Machine 2, not to meet any demand. Therefore, if Machine 2 "needs" to produce, it seems reasonable that preserving the buffer level is not a consideration. We have not been able to prove this, however, so we state it as conjecture.

CONJECTURE 5.1

In the optimal policy for the two machine system of Figure 5.2, the optimal control for the second machine is a function of $x_2(t)$ only, and the optimal policy is a pure PS hedging point policy.

■

Conjecture 5.1 says that all the information about the effect of Machine 1's behavior on the decisions Machine 2 takes can be summarized in a single hedging point value for each of the machine states.

Another observation from Figure 5.7 is that the boundary S_2 is not always present. That is, Regions 0 and 1 disappear. Note in example A3 and A4, that in the whole feasible region above S_3, the optimal decision if for Machine 2 to produce. Machine 2 is, therefore, acting as if it is in a transfer line, producing whenever it is able to. Note also that in these two cases, the cost of the internal buffer, C_1^+, is equal to or greater than the penalty on $x_2 \geq 0$, C_2^+.

The intuitive reason for this behavior is as follows: Given we have material in the buffer and the second FMS is able to produce, it is no more expensive to advance material than it is to leave it in the buffer. Further, the finished material is also more "reliable" in terms of meeting the demand than is the material in the buffer, because it does not have to be processed first. We again conjecture that this property is true for all cases.

CONJECTURE 5.2

In the two-machine system of Figure 5.2, if the cost of carrying material in the buffer is greater than the cost of carrying material at the output,

$$c_1^+ \geq c_2^+$$

then the optimal policy for the second FMS is to always produce if possible.

■

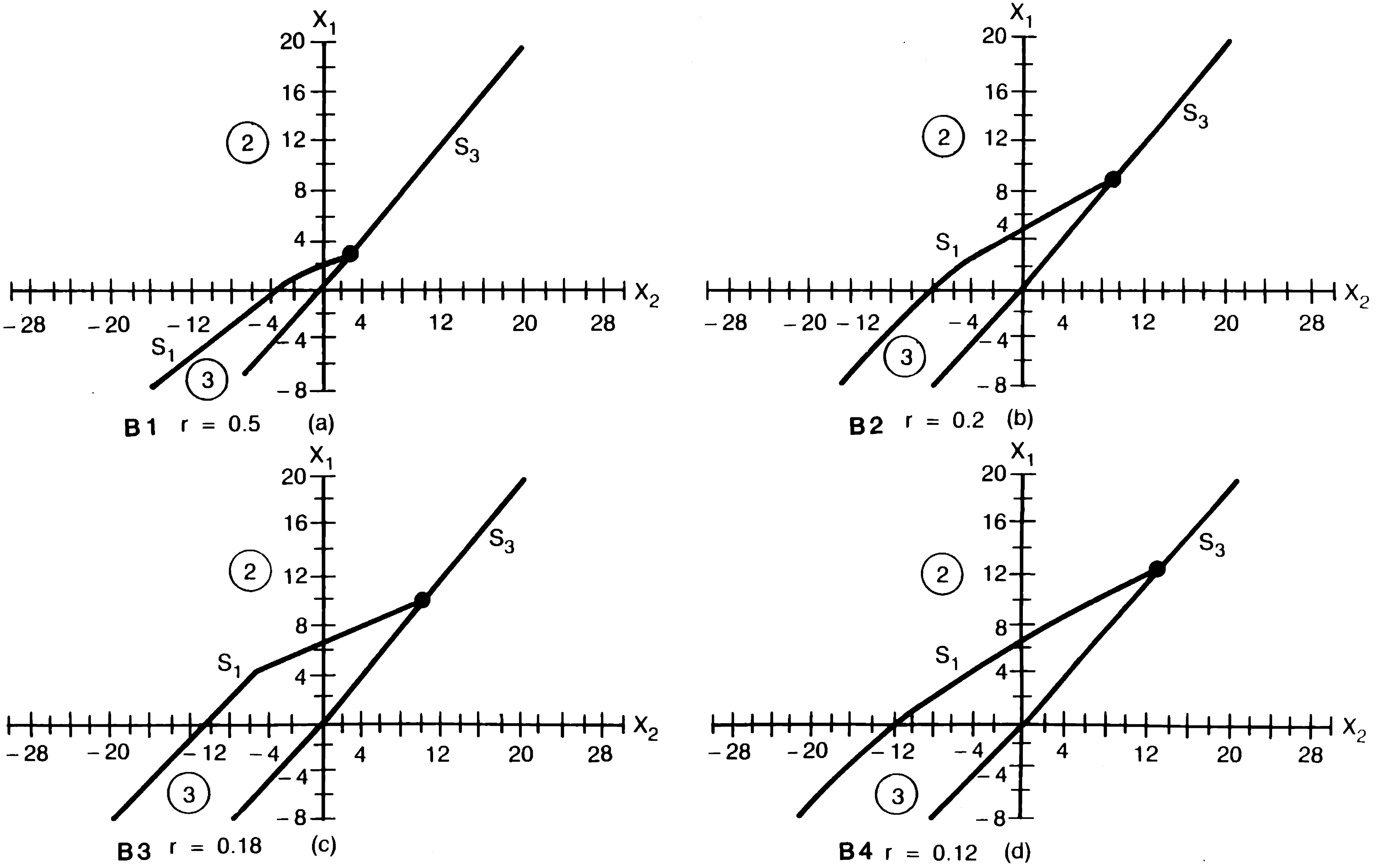
In practice, the cost of material in early stages is typically less than the cost of material is later stages because less material and labor (value added) has been expended per part. This is not always the case, however. For example, in perishable food processing (e.g. fish canneries), value added costs are low, yet the cost of in-process inventory (e.g. refrigeration) is high relative to warehouse costs (e.g. canned goods storage). When finished parts inventories are cheaper than in-processes inventories, Conjecture 5.2 says that the only decision that must be made is when to first load a part. Parts in the system then follow a "hot potato" policy, sometimes referred to as a "pull" policy, where each machine tries to process its waiting supply as fast as it can.

5.3.4 Control Regions For Series B

Figure 5.8 shows the control regions for the case $C_1^+ > C_2^+$ where the repair rate was varied. Again, these graphs are for the case $\alpha_1 = \alpha_2 = 1$. The boundaries for the other machine states paralleled the changes seen in the $\alpha_1 = \alpha_2 = 1$ cases.

Note that the S_2 boundary is not present, as Conjecture 5.2 predicts. As the repair rate is reduced, the hedging point moves out along S_3 and also S_1 moves away from S_3. This corresponds to a greater surplus hedging point and larger optimal buffer levels respectively. This seem intuitively correct, as we expect that an unreliable system will need to carry more buffer stock and surplus.

Also note that the S_2 boundary becomes parallel to the S_3 boundary as x_2 becomes negative. This indicates that the optimal buffer level reaches a maximum value for states where the surplus, x_2, is very negative. For these states, the control looks approximately like the pure buffer policy of Figure 5.5. Closer to the hedging point, the S_1 boundary becomes more nearly horizontal, becoming more like the PS surplus hedging point policy of Figure 5.6. We examine this characterization further in Section 5.4.



B1 r = 0.5 (a)

B2 r = 0.2 (b)

B3 r = 0.18 (c)

B4 r = 0.12 (d)

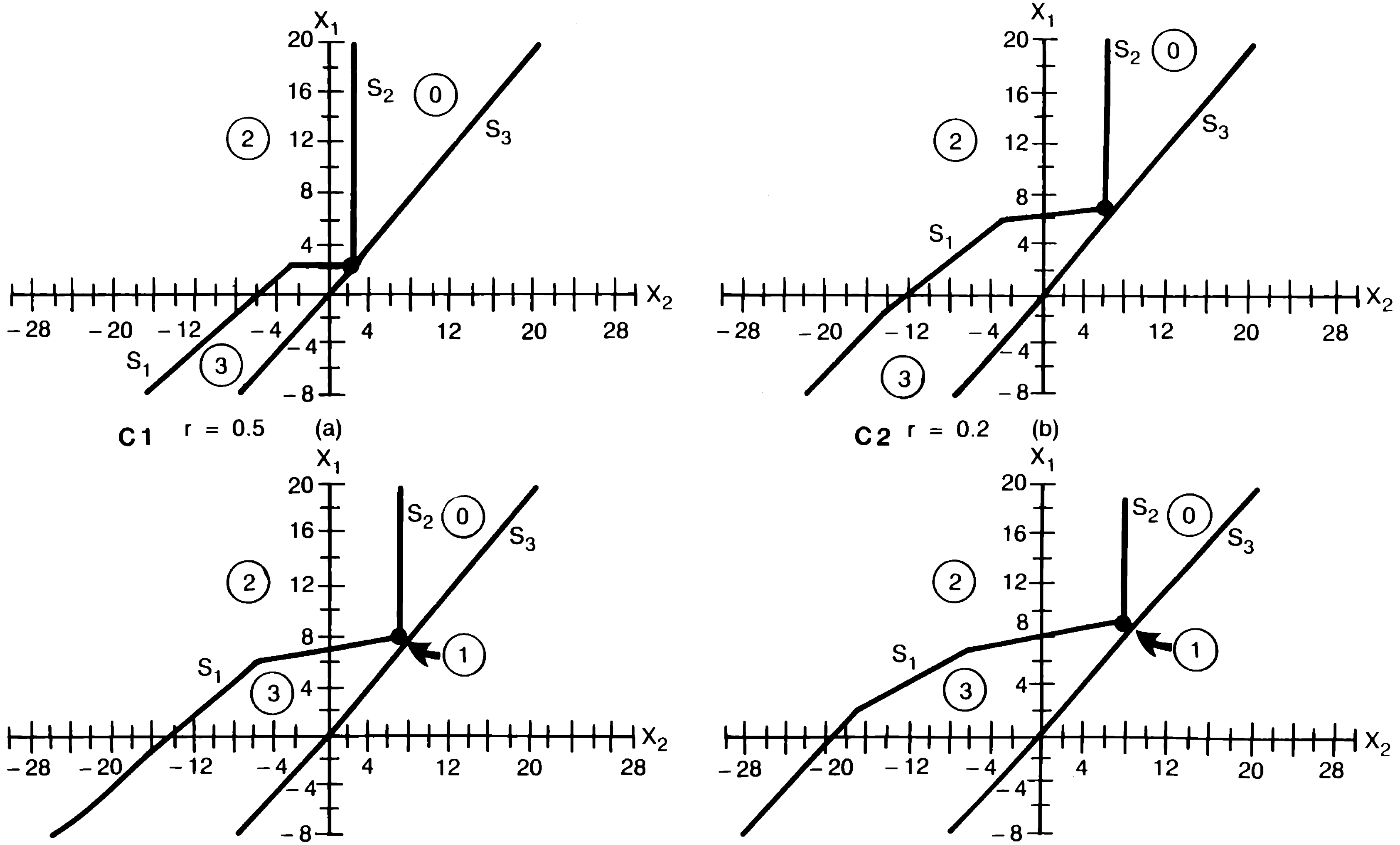
FIGURE 5.8 SERIES B

5.3.5 Control Regions For Series C

Figure 5.9 shows the graphs for Series C, where $C_1^+ < C_2^+$ and the repair rate is varied. The graphs are for $\alpha_1 = \alpha_2 = 1$. Again, the boundaries for other machine states paralleled the changes in the $\alpha_1 = \alpha_2 = 1$ case.

Boundary S_2 appears in these examples and is absolutely vertical, as Conjecture 5.1 predicts. Again we note the same increase in the hedging point value and optimal buffer level seen in Series B as the machines are made more unreliable. Also note that, again, the optimal buffer level, the vertical distance between S_1 and S_3, appears to reach a maximum for x_1 very negative. The tendency of the S_1 boundary to become horizontal near the hedging point is even more pronounced in this series than in Series B.

Note that the optimal buffer level near the hedging point is essentially zero (i.e. the hedging point lies on S_3) and that Region 1 is small or non-existent in all cases. This again follows the intuition presented in Section 5.3.3 about series A, namely that the buffer is an expensive resource that is only used when the system is far behind (x_2 negative).



C1 r = 0.5 (a)

C2 r = 0.2 (b)

C3 r = 0.18 (c)

C4 r = 0.12 (d)

FIGURE 5.9 SERIES C

5.4 TWO-POINT CONTROL

In all the cases of the previous section, the optimal control for the first machine is characterized by an increasingly horizontal S_1 close to the hedging point and an optimal buffer level that appears to reach a maximum far from the hedging point.

The control boundaries can roughly be characterized by dividing the production surplus space into two regions as shown in Figure 5.10, which shows case A2 when $\alpha_1=\alpha_2=1$. In Region II, where x_2 is negative, the optimal policy is approximately a pure buffer policy. In Region II, near the hedging point, the optimal policy is approximately a pure PS policy.

If we combine this characterization with Conjecture 5.1, that Machine 2 follows a pure PS hedging point policy, we get a policy that uses a pure PS hedging point for Machine 2, a buffer hedging point policy for Machine 1 in Region I and a PS hedging point for Machine 1 in Region II. Symbolically, this approximation takes the form

$$\min_{u\in\Omega(\alpha)} c_1(\alpha)\ u_1 + c_2(\alpha)\ u_2 \tag{5.22}$$

where

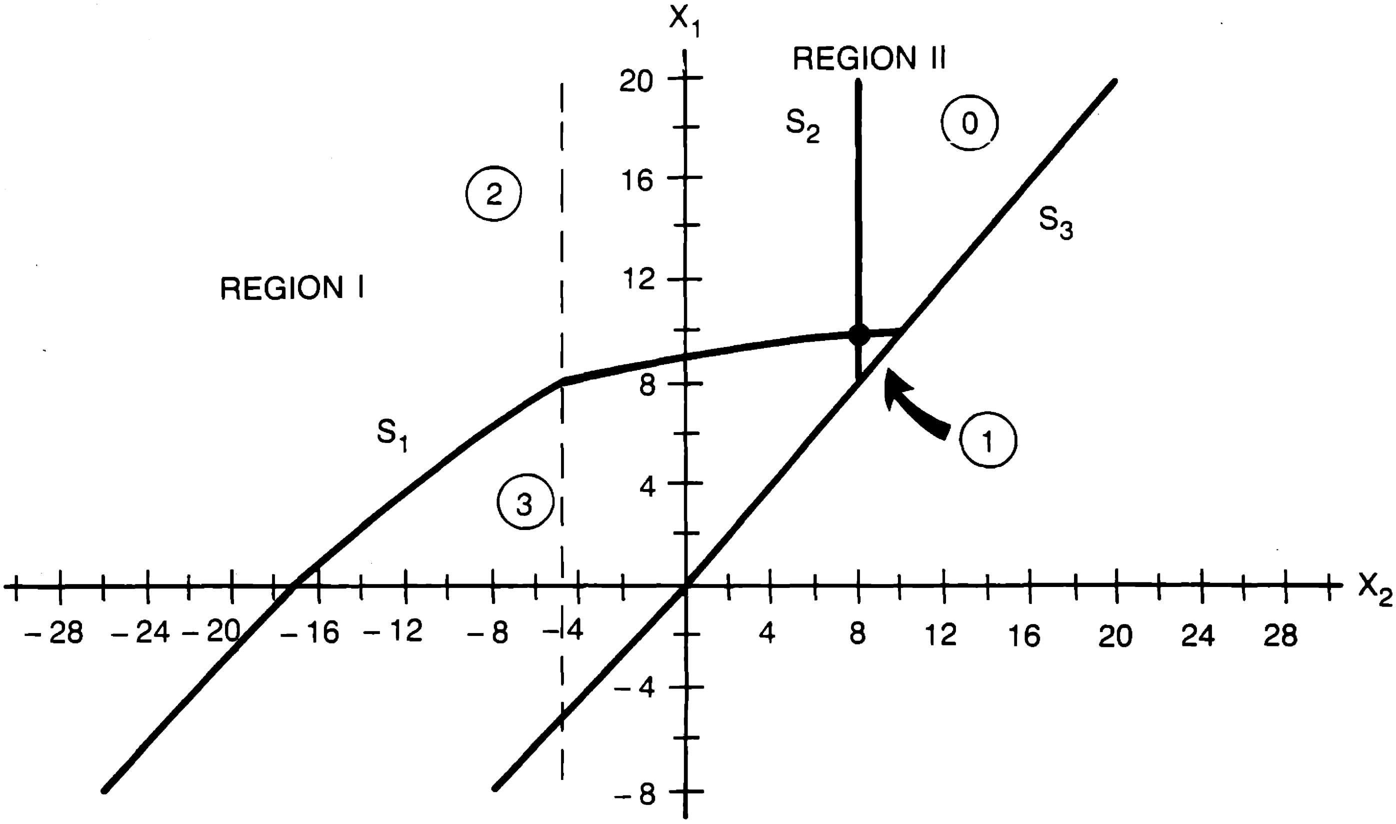


FIGURE 5.10 REGIONS I AND II

4357-87-18-4/24/87-1

$$c_1(\alpha) = \max \{ x_1(t) - x_1^H(\alpha), B(t) - B^H(\alpha)) \} \tag{5.23}$$

$$c_2(\alpha) = x_2(t) - x_2^H(\alpha) \tag{5.24}$$

We call this control a <u>two-point</u> control because it involves both PS hedging points and a buffer hedging point. Note that in this characterization, the coefficients c_1 and c_2 are not linear in x and B, and, therefore, this does not correspond to a quadratic value function approximation.

The two-point control is parameterized by $x_1^H(\alpha)$, $x_2^H(\alpha)$, and $B^H(\alpha)$. Only the values for $(\alpha_1,\alpha_2)=(1,0)$, (0,1) and (1,1) are needed. The number of parameters we need to choose is therefore small, yet the the basic characteristics of the control regions match those of the optimal control.

5.4.1 Determining The Parameters in a Two-Point Control

In this section, we suggest a method of determining the parameters $x_2^H(\alpha)$ and $B^H(\alpha)$. Our analysis is based on a steady state approximation of the behavior of the system in Regions I and II. We will, therefore, simplify the two-point control further and assume that the hedging points are not functions of α. That is

$$x_2^H(\alpha) = x_2^H \qquad \text{for all } \alpha \tag{5.25}$$

$$B^H(\alpha) = B^H \qquad \text{for all } \alpha \tag{5.26}$$

Determining B^H

To determine the value of the parameter B^H, note that in Region I, where x_2 is very negative, the buffer limit, B^H, determines the behavior of the system. Because we have assumed that B^H is not a function of α, the system, operating in Region I, behaves like a transfer line with a finite buffer of size B^H: Machine 2 always produces if $B(t)>0$ and Machine 1 always produces if $B(t)<B^H$.

Since the system is behind production in Region I, consider the finite horizon approximation to the optimization problem illustrated in Figure 5.11. The current production surplus is $x_2(t)$, the buffer level is $B(t)$ and the horizon is T. Assume that $x_2(t)$ is so negative that the $x_2(t+T)$ can never be greater than zero, even if both machines are up for the entire interval. This will be true if

$$x_2(t) < T\bar{U} \tag{5.27}$$

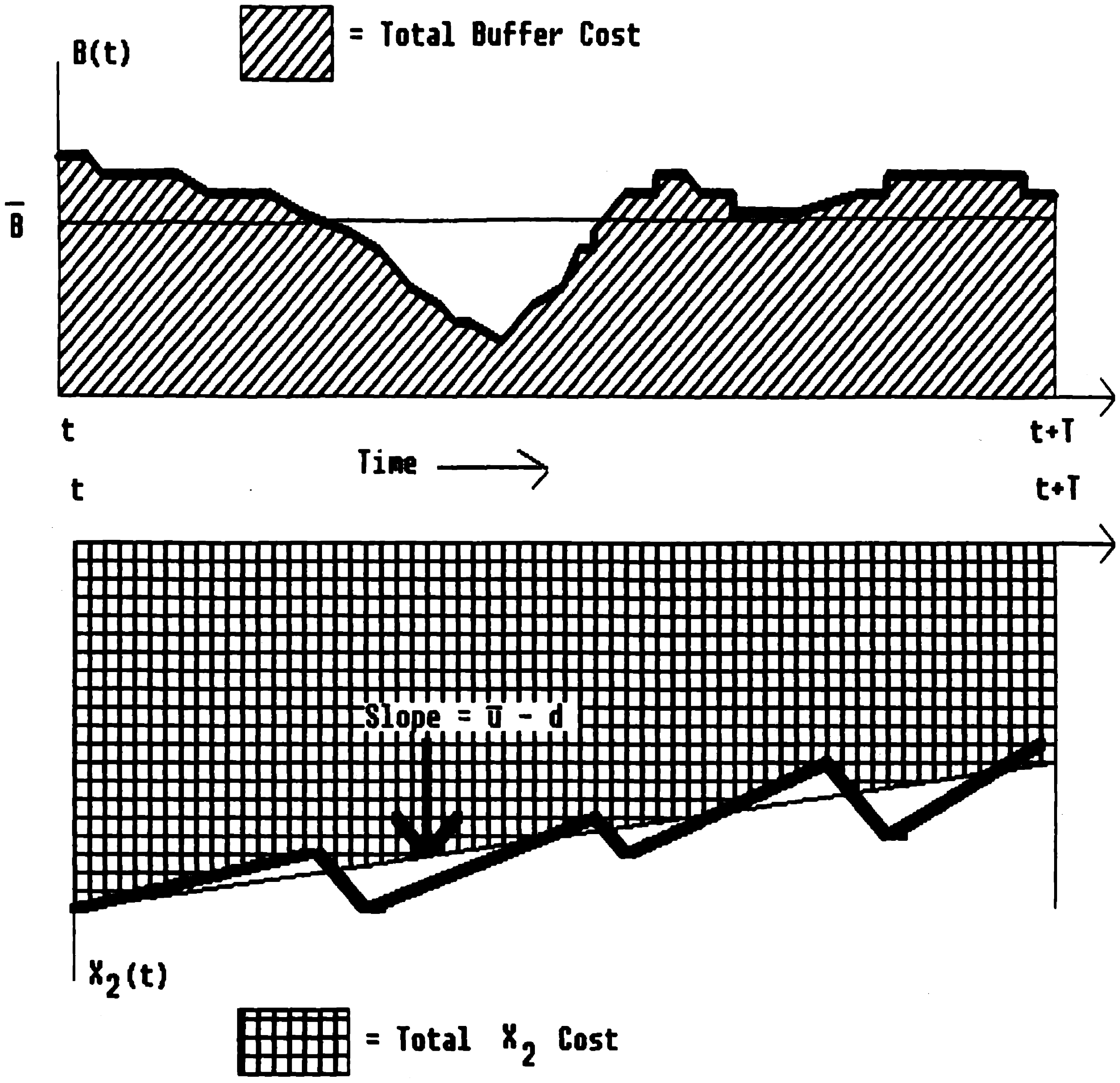


FIGURE 5.11 REGION I APPROXIMATION

where $\bar{U}$ is the maximum production rate of the two machines when both are operational. ($\bar{U}$=2 in the numerical examples of this chapter.)

If we evaluate the cost (5.21) over the finite interval [t,t+T], $x_2(t)$ is always negative so (5.21) becomes

$$\int_0^T e^{-\beta t} \left(C_1^+ B(t) - C_2^- x_2(t) \right) dt \qquad (5.28)$$

Let us further assume that the system is in steady state in [t,t+T] and that we are following the Region I buffer hedging point policy with parameter B^H. Let $\bar{B}(B^H)$ be the average buffer level during [t,t+T] and $\bar{u}(B^H)$ be the average rate of production during [t,t+T] for hedging value B^H.

We can approximate the x_2-trajectory, as shown in Figure 5.11, by a line of slope $\bar{u}$-d, which is the average rate of increase in $x_2(t)$ on the interval:

$$x_2(t) \approx x_2(0) + \left(\bar{u}(B^H) - d \right) t \qquad (5.29)$$

We can also approximate the B-trajectory by its average value as shown by the horizontal line at the top of Figure 5.11. That is

$$B(t) \approx \bar{B}(B^H) \qquad (5.30)$$

Evaluating (5.28) using these approximations yields

$$\int_0^T e^{-\beta t} \left[C_1^+ \bar{B}(B^H) - C_2^- \left(x(0) + \left(\bar{u}(B^H) - d \right) t \right) \right] dt \qquad (5.31)$$

If we now minimize (5.31) with respect to B^H, the constant terms due to $x(0)$ and d drop out, and the minimization becomes

$$\min_{B^H \geq 0} \int_0^T e^{-\beta t} \left[C_1^+ \bar{B}(B^H) - C_2^- \, t \, \bar{u}(B^H) \right] dt \qquad (5.32)$$

Letting $T \rightarrow \infty$ and evaluating the integrals in (5.32) implies the optimal B^H satisfies

$$\min_{B^H \geq 0} \left[\frac{C_1^+}{\beta} \bar{B}(B^H) - \frac{C_2^-}{\beta^2} \bar{u}(B^H) \right. \qquad (5.33)$$

The solution to (5.33) will give the buffer hedging point, B^H, that optimally balances high throughput, $\bar{u}$, and low buffer level, $\bar{B}$ under the Region I policy when the system is infinitely far behind.

Since we assumed the system is operating in steady state and that it is following a policy based on a fixed buffer hedging point, the system behaves exactly like a transfer line with a finite buffer of size B^H. $\bar{B}$ and $\bar{u}$ are, therefore, the average buffer level and average throughput of a two-machine, one-buffer transfer line with buffer size B^H.

Analytical solutions for the throughput, $\bar{u}$, and average buffer level, $\bar{B}$, as a function of the buffer size, B^H, for a two-machine transfer line with a finite buffer can be found in Gershwin and Schick (1980). Their solution is for a more general problem where the processing time on the two machines can be different. A simple one-dimensional optimization can be used to solve for the optimal B_1^H using (5.33). The problem statement and analytical expressions of Gershwin and Schick (1980) are given in Appendix C.

Determining x_1^H and x_2^H

To complete the two-point control, we need to determine values for x_1^H and x_2^H. We do this by recalling that Conjecture 5.1 says that Machine 2 follows a pure x_2-hedging-point policy. If we let the value of the hedging point be independent of α, then Machine 2 behaves as if it is in isolation, like a simple one-machine, one-part-type FMS.

The effect of the buffer can be incorporated by modifying the failure rate of Machine 2. From the analytical results of Gershwin and Schick, the probability of the buffer being empty while Machine 2 is operational, a condition called starvation, can be computed. We will denote this probability, p_s.

Suppose we modify the machine state α_2 as shown in Figure 5.12. We now consider the machine to be in a failed state if Machine 2 is either failed or starved. Note that Machine 2 goes from being starved to not being starved when Machine 1 is repaired (Machine 2 has to be working if it is starved). Therefore, the transition rate out of the starved "state" takes place at rate r. Therefore, Machine 2 always changes from the "failed or starved" state to the operational state at rate r. Of course, the process is not Markovian any longer. However, we will assume that it can be approximated by a Markov process as shown in Figure 5.12.

The transition into the "failed or starved" state takes place at a rate p', which is not known. We get p' by noting that the probability of the new process being in the "failed or starved" state must be equal to the probability that, in the original process, Machine 2 is either starved or broken. (This assumes that a machine that is starved cannot fail.) That is

$$\frac{p}{r+p} + p_s = \frac{p'}{r+p'} \tag{5.34}$$

or

$$p' = \frac{r\left(\frac{p}{r+p} + p_s \right)}{1 - \left(\frac{p}{r+p} + p_s \right)} \tag{5.35}$$

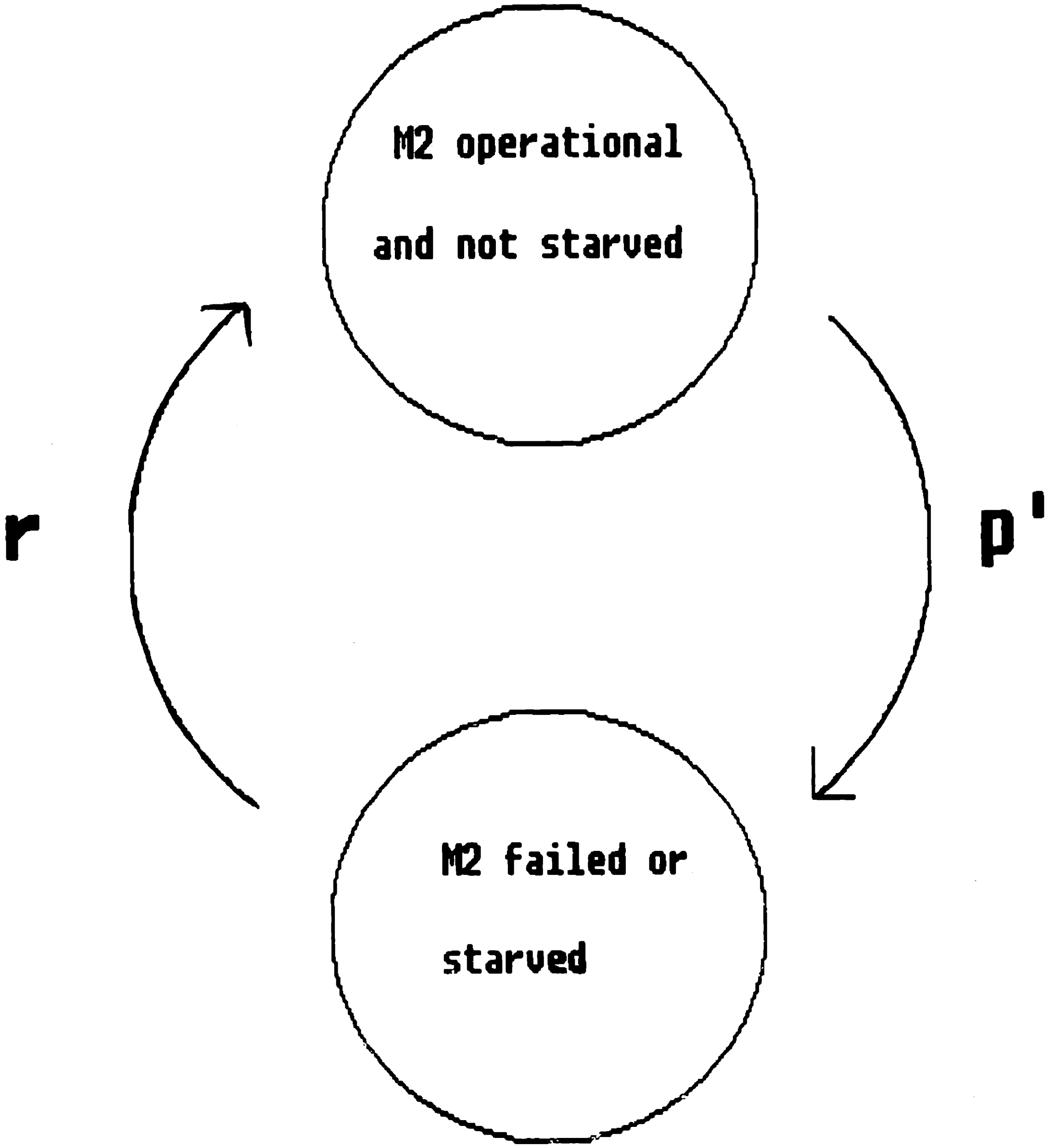


FIGURE 5.12 APPROXIMATION OF BUFFER EFFECTS

A hedging point, x_2^H, can then be generated for the second machine based on the analytical results for the one-machine, one-part type problem of Akella and Kumar (1986) using the parameters r and p'. A description of this problem and the analytical results are given in Appendix C as well.

Finally, by our previous observations, the hedging point is typically on the zero buffer boundary, S_3. Therefore we set $x_1^H=x_2^H$.

The steps for computing the parameters of the Two-Point control are summarized below:

STEP 1 - Compute the value of B_1^H by solving the minimization (5.33) using the analytical expressions for the average buffer level, $\bar{B}(B^H)$, and throughput, $\bar{u}(B^H)$, given in Appendix C.

STEP 2 - Compute the probability p_e from Appendix C.

STEP 3 - Generate the hedging point x_2^H by considering the second machine in isolation but with the new failure rate p' from (5.35) using the expressions in Appendix C.

STEP 4 - Set $x_1^H=x_2^H$.

In the next section, we look at the regions generated by the two-point control for some of the numerical examples.

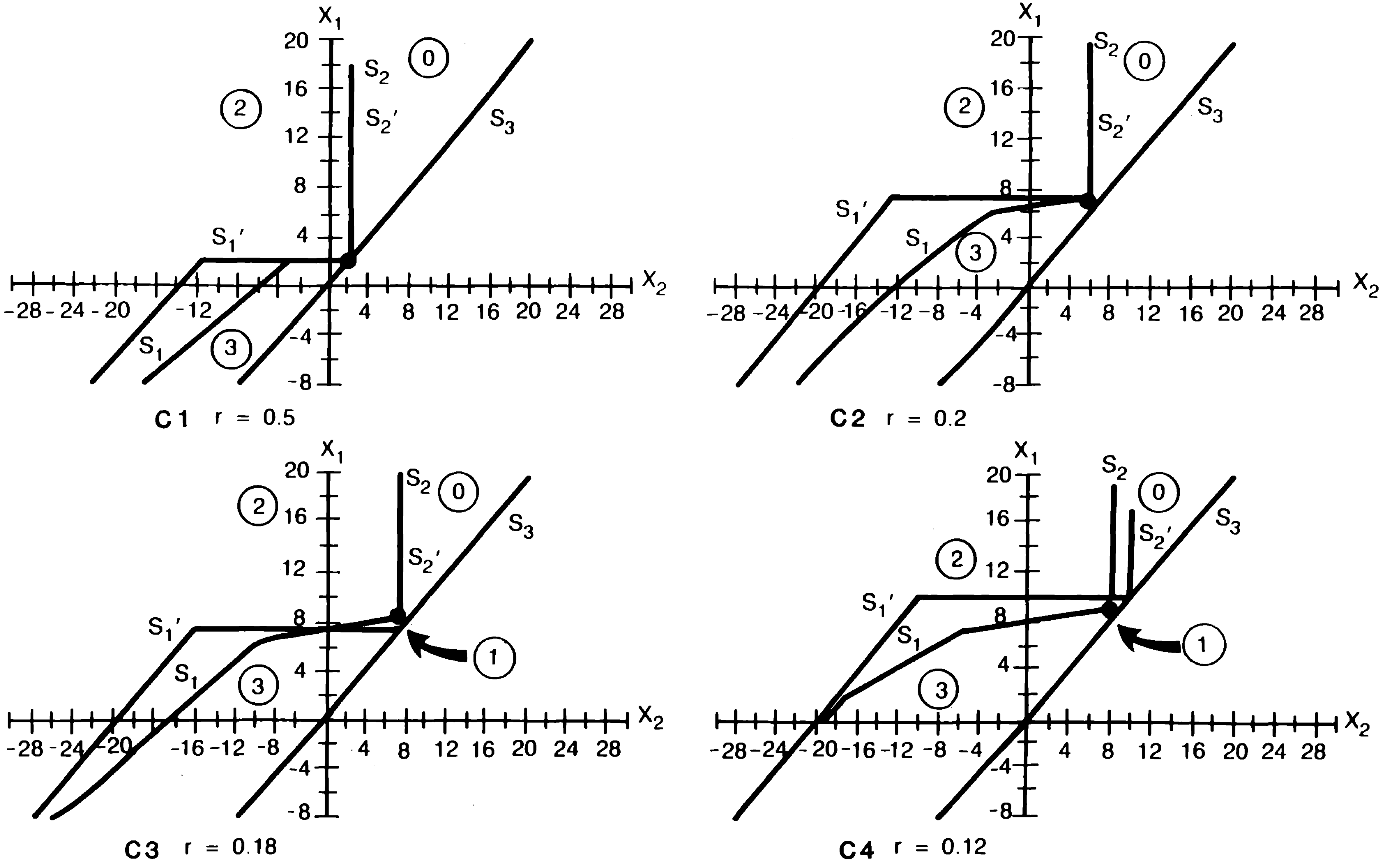
5.4.2 Example Regions of Two-Point Control

Figure 5.13 shows the regions of the two-point control superimposed on the optimal regions for Series C. The two-point control boundaries, labeled S_1', S_2' and S_3', are the darker boundaries. Calculating these boundaries involves only a one-dimensional search to solve (5.29) and took only seconds to perform on a PC. The optimal control takes over 5 hours to compute using the same machine.

Because we assumed the hedging points are not functions of α, the two-point boundaries will not change as α changes. The optimal boundaries do change somewhat with α as we observed in Section 5.3.2.

Note that the location of boundary S_2' is nearly the same as S_2, except for case C4 where it is slightly to the right of S_2. This suggests that adding p_s to the failure rate of Machine 2 is a very close approximation of the effect of the upstream process on Machine 2.

The S_1' boundary is not close to S_1 in C1 and C2, but is closer is C3, while in C4, S_1 approaches S_1' almost exactly. That S_1' becomes closer to S_1 as the repair rate is decreased is reasonable given our approximation. Recall that to get the buffer limit represented by S_1', we assumed the system was far behind



C1 r = 0.5

C2 r = 0.2

C3 r = 0.18

C4 r = 0.12

FIGURE 5.13 TWO-POINT CONTROL SERIES C

($x_2 << 0$) and would spend a long time catching up ($T \rightarrow \infty$ in (5.28)). This lead to the infinite horizon, steady state optimization (5.29). Systems that are heavily utilized, like C4, have very little excess capacity. They therefore take a long time to build up a surplus once they have fallen behind. The infinite horizon approximation of Section 5.4.1 is, therefore, quite good for heavily loaded systems.

Systems with a lot of excess capacity can build surpluses quickly and hence the assumption that they will be behind for a long time when $x_2 < 0$ is not as good. We conjecture, however, that if we looked at the state space for $x_2 << 0$, S_1 would eventually reach the limit S_1.

It is encouraging that the approximation works best for heavily loaded systems. Most manufacturing systems represent a substantial capital investment and, therefore, tend to be run close to capacity. Optimal use of machines and stocks is especially critical in such cases. The two-point control can generate close to optimal regions for this important class of problems. It is also simple to compute, easy to implement and has an intuitively satisfying interpretation.

5.5 SUMMARY

In this chapter, we saw that the instantaneous capacity restriction can limit the throughput of a manufacturing system. This led us to examine the control of a simple two-machine system with an internal buffer, where the instantaneous capacity restriction was not imposed. Using dynamic programming, we generated optimal controls for this system.

Two conjectures were made based on observations of the optimal controls. The first is that Machine 2 always follows a policy based only on a surplus hedging point. The second conjecture is that if the cost of positive surplus at the output is greater than the cost of material in the buffer, then Machine 2 always produces if it can. The second conjecture, in particular, can be used to greatly simplify the control of systems with downstream storage costs that are less than upstream storage costs. A proof of either of these conjectures would be a good future research goal.

We then introduced the two-point control characterization of the two-machine, one-part-type system. A method for computing the parameters of this control was suggested that provides a close approximation to the optimal control boundaries, especially for systems that are highly utilized. Additional research will be needed to see if this characterization can be extended to lines with machines of different speeds and reliabilities, or to lines with more than two machines. An extension to an true FMS, with multiple part types would also we desirable.

CHAPTER 6 - SUMMARY AND SUGGESTIONS FOR FURTHER RESEARCH

6.1 SUMMARY

We mentioned in Chapter 1 that the long-term goal of manufacturing research is the development of models and techniques for handling general manufacturing systems. In this thesis, we have taken three preliminary steps toward this goal.

The first is an analysis of how delay arises in manufacturing systems. In Chapter 3, a straight forward examination of continuous flow models using Little's Theorem revealed that large aggregations of machines, such as one might find in a large work station or sub-module of a hierarchical system, often have to modeled with delay. Delay is, therefore, a fundamental characteristic of many manufacturing systems and should be incorporated in manufacturing system models.

The effects of flow rate delay on the Kimemia-Gershwin FMS model were also examined in Chapter 3. We showed that while work station delay can be accommodated in the Kimemia-Gershwin model, the necessary internal inventories are proportional to the delay. To reduce these inventories, a manufacturing process with delay will require a more complex control than the one presented in Chapter 2 .

As a step toward understanding the form of such a control, we developed a linear system approximation technique in Chapter 4 that allows the direct application of the theoretical results of

Kimemia and Gershwin to systems with delay. The approximation allowed us to side-step the analytical difficulties involved in working with the original delay state. Using this technique, a linear program form for the control of delay systems was generated. Using quadratic approximations, the coefficients of this linear program turned out to involve linear terms in the current inventories and past controls.

Unfortunately, systematic methods for analyzing a general linear system with Markov-jump disturbances do not exist. We therefore gave a heuristic predictor interpretation of the terms involved in the linear program in Section 4.5. Simulation examples showed that even crude forms of the predictor control substantially lowered backlog and inventory costs when compared to simple hedging point policies.

To investigate the implications of sub-dividing an FMS, our final effort was a preliminary investigation in Chapter 5 of the optimal control for a simple interconnected system. Using dynamic programming, optimal controls were generated and plotted using a surplus space representation.

We observed in Conjecture 5.1 that the second machine always followed a policy based on a surplus hedging point. The first machine followed a policy that can be characterized by two regions. In one region, the control is based on a production volume hedging point. In the other region, the control is based on a physical buffer hedging point. A simple heuristic, called the two-point control, was developed in Section 5.4 to approximate this behavior. A simple method of computing the parameters of the control was then presented.

6.2 SUGGESTIONS FOR FURTHER RESEARCH

There are both short-term and long-term extensions to the basic work presented in this thesis. An immediate extension could be an investigation of the weighting function controls presented in Chapter 4. For example, should the functions be increasing, decreasing or constant? Heuristic strategies, like the predictor control of Section 4.5, for choosing functions and coefficients are also needed. Extensions to systems with more complicated flows, such as reentrant flow systems, are needed as well.

A more indepth investigation could include work on multiple part type systems with delay. For example, what effect does the different processing time of parts have on the delay model and the form of the control?

Immediate extensions following the work in Chapter 5 could include a proof on Conjectures 5.1 and 5.2. Conjecture 5.1, that the second machine follows a pure surplus hedging point policy, is appealing because it says that the effect of the upstream process can be summarized in a single hedging point value. Conjecture 5.2, that the "hot potato" policy is optimal if buffer costs are decreasing along the line, seems intuitively correct and would greatly simplify the control for systems with decreasing buffer costs.

The buffer limit in the two-point control is also worth investigating in more detail. For example, can one prove this limit exists? Does it become unbounded for an average rather than a discounted cost criterion. It would also be worth investigating how the parameters in the two-point control vary as a function of the system parameters for a large class of systems. Finally, a simulation study of a realistic system to determine the performance of the two-point control against common (i.e. shortest processing time) heuristics is needed.

Several long-term research goals are also suggested. For example, by examining the effects of delay on the continuous flow model of Kimemia and Gershwin, the question of how to define criteria for applying the model was raised in Chapter 3. We also saw in Chapter 5 that the instantaneous capacity constraint restricts the potential capacity of an FMS if the FMS is too large. More work is required to understand the precise capacity vs. inventory tradeoffs involved in subdividing the control of an FMS.

For example, in developing the minimum inventory level in Chapter 3, we assumed that the loading rate at the input to an FMS was observed at every workstation. This is certainly feasible given the instantaneous capacity restriction, but it is not a necessary implementation. One could easily imagine a policy that loads according to the instantaneous capacity restriction but follows a different (e.g. first come, first served) internal policy. Such an approach would be similar to flow control in a network of queues.

Finally, as an ultimate goal, one could extend the ideas on both delay and interconnected systems to general network configurations.

APPENDIX A: PROOF OF LEMMA 4.1

Introduction

In this appendix, we prove the result stated earlier in Lemma 4.1, Chapter 4. The Lemma is repeated for convenience below:

Lemma 4.1

Let $u(t)$ be absolutely integrable with $u(t)=0$ for $t \leq 0$ and $|u(t)| \leq B$ for all $t>0$. If $u(t)$ is applied to both the delay element,

$$y(t)=u(t-\tau) \tag{A-1}$$

and the m'th order system,

$$\begin{aligned}(\tau/m)\dot{z}_1(t) + z_1(t) &= u(t) \qquad \text{(A-2)}\\ (\tau/m)\dot{z}_2(t) + z_2(t) &= z_1(t)\\ &\vdots\\ (\tau/m)\dot{z}_m(t) + z_m(t) &= z_{m-1}(t)\end{aligned}$$

with initial condition $z_j(0)=0$, $j=0,1,..m$, then

$$\int_0^t (\ z_m(\xi)-y(\xi)\ )\ d\xi \rightarrow 0 \tag{A-3}$$

uniformly as $m \rightarrow \infty$.

The proof of Lemma 4.1 will proceed as follows: First, we will establish a bounded-input bounded-output condition on a simple first order differential equation similar to (A-2). We will then prove the intermediate result that $z_m(t) \rightarrow u(t-\tau)$ as $m \rightarrow \infty$ if u(t) is twice differentiable with a bounded second derivative. Using this result, we show that $z_m(t)$ converges to $u(t-\tau)$ under weaker conditions, namely when u(t) has a bounded first derivative. Finally, we prove Lemma 4.1 by integrating at the input rather than at the output of the linear system and using the intermediate results above.

First, we need to establish the following fact:

Fact 1

If the input, f(t), to the differential equation,

$$\dot{x}(t) + \frac{x(t)}{a} = f(t) \qquad \text{(A-4)}$$

is bounded, $|f(t)| \le B$, and $x(0) = 0$, then

$$|\ x(t)\ | \le aB \text{ for all } t \ge 0. \qquad \text{(A-5)}$$

Proof

The Laplace transform of A-4 is $\frac{1}{s+1/a}$ which has an inverse transform $e^{-t/a}$. Therefore

$$| x(t) | = | \int_0^t e^{-\xi/a} f(t-\xi) \, d\xi | \tag{A-6}$$

$$\leq \int_0^t | e^{-\xi/a} | \, | f(t-\xi) | \, d\xi$$

$$\leq B \int_0^t | e^{-\xi/a} | \, d\xi$$

$$\leq B \int_0^\infty | e^{-\xi/a} | \, d\xi$$

$$= a B \tag{A-7}$$

■

Twice Differentiable u(t) with Bounded Second Derivative

We now prove the pointwise convergence of $z_m(t)$ to $y(t)$ for an input $u(t)$ that is twice differentiable with $|\ddot{u}(t)| \leq K$ for all t. We will begin by taking $m=1$ in (A-2). Define the error, $e(t)$, by

$$e(t) \triangleq z_1(t) - y(t) \tag{A-8}$$

We recall that $z_1(0)=0$. To evaluate $|e(t)|$, we can differentiate (A-8) and use (A-2) and (A-1) to get

$$\dot{e}(t) = \dot{z}_1(t) - \dot{y}(t) = \frac{u(t)-z_1(t)}{\tau} - \dot{u}(t-\tau) \tag{A-9}$$

or

$$\dot{e}(t) = -e(t)/\tau + \phi(t) \tag{A-10}$$

where

$$\phi(t) = \frac{u(t)-u(t-\tau)}{\tau} - \dot{u}(t-\tau). \tag{A-11}$$

Note that for some $\bar{t}\varepsilon[t-\tau,t]$ we have by the mean value theorem,

$$\dot{u}(\bar{t}) = \frac{u(t)-u(t-\tau)}{\tau}. \tag{A-12}$$

Therefore,

$$| \phi(t) | = | \dot{u}(\bar{t}) - \dot{u}(t-\tau) | \tag{A-13}$$

$$= | \int_{t-\tau}^{\bar{t}} \ddot{u}(\xi)\, d\xi | \tag{A-14}$$

$$\leq | \bar{t} - (t-\tau) | K \leq K\tau. \tag{A-15}$$

Applying Fact 1 to (A-10) we have

$$| e(t) | \leq \tau | \phi(t) |. \tag{A-16}$$

Therefore, by combining (A-15) and (A-16) we have,

$$| e(t) | \leq K\tau^2 \qquad \text{(A-17)}$$

We now consider a system of m such links as defined in A-2. We assume $x_j(t)=0$ $j=1,2,..m$ and define the j'th error, $e_j(t)$, by

$$e_j(t) \triangleq z_j(t) - y_j(t) \qquad \text{(A-18)}$$

where

$$y_j(t) \triangleq u(t-j\tau/m). \qquad \text{(A-19)}$$

We can write the output of the first link as

$$z_1(t) = y_1(t) + e_1(t) \qquad \text{(A-20)}$$

Now let

$$z_2(t) \triangleq z_2^1(t) + z_2^2(t) \qquad \text{(A-21)}$$

where $z_2^1(t)$ and $z_2^2(t)$ are the contributions to $z_2(t)$ from $y_1(t)$ and $e_1(t)$ respectively. That is, $z_2^1(t)$ and $z_2^2(t)$ satisfy

$$(\tau/m)\, \dot{z}_2^1(t) + z_2^1(t) = y_1(t) \qquad \text{(A-22)}$$

$$(\tau/m)\, \dot{z}_2^2(t) + z_2^2(t) = e_1(t)\ . \qquad \text{(A-23)}$$

Using (A-21) we have

$$| e_2(t) | = | z_2(t) - y_2(t) | \tag{A-24}$$

$$\leq | z_2^1(t) - y_2(t) | + | z_2^2(t) |. \tag{A-25}$$

Note that the input to (A-22) is $y_1(t)=u(t-\tau/m)$ by (A-19). Therefore, this equation is simply a time shifted version of the equation in (A-2) for z_1. Therefore, since $y_2(t)=u(t-2\tau/m)$, the first term in (A-25) is

$$| z_2^1(t) - y_2(t) | = | e_1(t-\tau/m) | \leq K(\tau/m) \tag{A-26}$$

The last inequality follows from our previous result, (A-17).

Applying Fact 1 to (A-23) we obtain

$$| z_2^2(t) | \leq | \tau/m | \, | me_1(t)/\tau | \tag{A-27}$$

$$= | e_1(t) | \, . \tag{A-28}$$

Note that (A-28) shows that the output of a link in (A-2) is bounded by the bound on its input. Applying (A-17), the result for one link, to (A-28) yields

$$| z_2^2(t) | \leq K(\tau/m)^2 \, . \tag{A-29}$$

Therefore, using (A-26) and (A-29), (A-25) becomes

$$| e_2(t) | \le K(\tau/m)^2 + K(\tau/m)^2 = 2K(\tau/m)^2 \qquad \text{(A-30)}$$

Continuing in this manner we get

$$| e_m(t) | \le mK(\tau/m)^2 = K\tau^2/m \qquad \text{(A-31)}$$

from which it follows that $z_m(t) \rightarrow y(t)$ uniformly as $m \rightarrow \infty$.

Differentiable u(t) with Bounded First Derivative

We now weaken the condition on u(t). Suppose that u(t) is continuous and that it has a first derivative with $|\dot{u}(t)| \le M$ for all t. Define,

$$u(t) = u^1(t) + u^2(t) \qquad \text{(A-32)}$$

where,

$$u^1(t) = 1/h \int_t^{t+h} u(\xi)\, d\xi \qquad \text{(A-33)}$$

We then have,

$$\dot{u}^1(t) = \frac{u(t+h) - u(t)}{h} \tag{A-34}$$

which means,

$$| \ddot{u}^1(t) | \le 2M/h \tag{A-35}$$

For $u^2(t)$ we have $|u^2(t)| = |u(t) - u^1(t)|$. By the mean value theorem we can write,

$$u(\xi) = u(t) + (\xi - t)\, \dot{u}(\, \theta(\xi)\,) \tag{A-36}$$

for some $\theta(\xi)$. Therefore we have,

$$| u^2(t) | = | u(t) - 1/h \int_t^{t+h} [\, u(t) + (\xi - t)\, \dot{u}(\, \theta(\xi)\,)\,]\, d\xi \,| \tag{A-37}$$

$$\le 1/h \int_t^{t+h} (\, \xi - t\,)\, M\, d\xi = Mh/2 \tag{A-38}$$

By the definition of e_m, (A-18), and linearity we have,

$$| e_m(t) | = | z_m(t) - y(t) | \tag{A-39}$$

$$= | z_m^1(t) + z_m^2(t) - y^1(t) - y^2(t) | \tag{A-40}$$

where $y^1(t)$ and $y^2(t)$ are the delayed outputs due to the inputs $u^1(t)$ and $u^2(t)$ respectively. From (A-40) we have

$$| e_m(t) | \le | z_m^1(t) - y^1(t) | + | z_m^2(t) | + | y^2(t) | \qquad \text{(A-41)}$$

Note that $|y^2(t)|=|u^2(t-\tau)|\le Mh/2$ by (A-38). Also, by Fact 1, the output of each link is bounded by the bound on its input (cf. (A-27),(A-28)) so $|z_m^2(t)|\le Mh/2$. Finally, because $u^1(t)$ is twice differentiable with $|\ddot{u}^1(t)|\le 2M/h$ we have by our previous result, (A-31),

$$| z_m^1(t) - y^1(t) | \le 2M\tau^2/hm \ . \qquad \text{(A-42)}$$

Combining all these bounds with (A-41) gives us,

$$| e_m(t) | \le Mh + 2M\tau^2/hm \qquad \text{(A-43)}$$

Letting $h=\tau/\sqrt{m}$ yields,

$$| e_m(t) | \le \frac{M\tau}{\sqrt{m}} + \frac{2M\tau}{\sqrt{m}} \qquad \text{(A-44)}$$

from which it follows that $z_m(t)\rightarrow y(t)$ uniformly as $m\rightarrow\infty$.

We summarize our results so far in the following lemma.

Lemma A.1

If the input, u(t), to systems (A-1) and (A-2) has a first derivative and that first derivative is bounded for all t, then $z_m(t) \rightarrow y(t)$ uniformly in t.

Proof of Lemma 4.1

Lemma A-1 is the result of Repin (1965). Unfortunately, in our problem u(t) does not necessarily have a well defined first derivative at all times and thus Lemma A.1 alone does not guarantee that the linear system approximation in Lemma 4.1 is valid. Note, however, that the state equation of a FMS is of the form,

$$\dot{x}(t) = u(t-s) - d. \qquad \text{(A-45)}$$

x(t) is an integrator, therefore if we replace u(t-s) by $z_m(t)$, the x-trajectory will be unchanged if,

$$\int_0^t z_m(\xi)\, d\xi \rightarrow \int_0^t u(\xi-\tau)d\xi = \int_0^t y(\xi)\, d\xi \qquad \text{(A-46)}$$

as m→∞. This is what Lemma 4.1 asserts. The integrals in (A-46) will, however, only be defined for all t if u(t) is absolutely integrable. We therefore require u(t) to satisfy this condition.

If using $z_m(t)$ in place of u(t-s) results in identical trajectories for a given input, u(t), it follows that an optimal control obtained by using the m'th order system with m→∞ will also be optimal for the original delay system.

To show that the integrals converge, we integrate at the input of the two systems rather than at the output. This change does not affect the input-output response of the system. If u(t) is integrated before being applied to the linear system approximation, the results of Lemma A.1 can be applied to prove Lemma 4.1.

Let $H_1(s)$ be the Laplace transform of the delay system and $H_2(s)$ be the Laplace transform of the m'th order approximation. Because we have assumed the initial conditions are zero at t=0, and we have defined our integrals over [0,t] we can write,

$$Y_1(s) = \frac{H_1(s)U(s)}{s} \qquad \text{(A-47)}$$

$$Z_m(s) = \frac{H_2(s)U(s)}{s} \qquad \text{(A-48)}$$

where $U(s)=\mathcal{L}\{u_2(t)\}$. If we define,

$$U_I(s)-U(s)/s \qquad \text{(A-49)}$$

then

$$u_I(t) = \int_0^t u_2(\xi)\ d\xi \qquad \text{(A-50)}$$

and

$$Y_1(s) = H_1(s)U_I(s) \qquad \text{(A-51)}$$

$$Z_m(s) - H_2(s)U_I(s) \qquad \text{(A-52)}$$

Note that $u_I(t)$ is continuous with

$$|\dot{u}_I(t)| = |\ u(t) - 0\ | \le K\ . \qquad \text{(A-53)}$$

If we apply the signal $u_I(t)$ to the two systems, the outputs will converge uniformly by Lemma A-1, but these outputs are given by (A-51) and (A-52). Therefore, we conclude that,

$$\int_0^t z_m(\xi)\ d\xi \rightarrow \int_0^t y(\xi)\ d\xi \qquad \text{(A-54)}$$

as $m \to \infty$ uniformly for all t. This concludes the proof of Lemma 4.1.

We will sometimes want the input to the m'th order system to be $u(t)\tau/m$ instead of $u(t)$ for physical reasons. Multiplying $u(t)$ by τ/m gives the variables z_j units of material rather than units of material per unit time. This make the results of the approximation easier to interpret. Therefore, we need the following corollary to Lemma 4.1.

Corollary

Suppose the first equation of (A-2) in Lemma 4.1 is,

$$(\tau/m)\dot{z}_1(t) + z_1(t) = (\tau/m)\, u(t) \qquad \text{(A-55)}$$

and

$$y(t) = (\tau/m)u(t - \tau) \qquad \text{(A-56)}$$

then

$$\int_0^t z_m(\xi)\, d\xi \to \int_0^t y(\xi)\, d\xi \qquad \text{(A-57)}$$

as $m \to \infty$ uniformly for all t.

The proof follows from the linearity of the system. Let $\tilde{u}=(\tau/m)u$ and $\tilde{M}=(\tau/m)M$ and repeat the proof of Lemma 4.1 up to (A-43). Substituting $(\tau/m)K$ for $\tilde{K}$, the convergence follows as before.

APPENDIX B: VALUE ITERATION SOURCE CODE

{ This program computes the optimal control for the propblem in chapter 4 by using value iteration and a finite state approximation to the state space.

The state space is confined to integer values for x. The alpha state is represented by an integer as follows: If alphai=1 machine i is up, if alphai=0 machine i is down. The combined alpha state is represented by a single integer as shown:

alpha1	alpha2	state
0	0	0
0	1	1
1	0	2
1	1	3

The state of the surplus, denoted y, and the internal buffer denoted x evolve as follows:

```
                  - - - -              - - - -
-----> u1 ---> | m1 | --> x --- u2 ---> | m2 | --> y ---> d
               |         |          |          |
```

The controls, u1 and u2, are also limited to their extreme values,

```
u1   {0,r1}
u2   {0,r2}
```

The value interation algorithm takes the form

STEP0:Set V0(i)=0 for all states i.

STEP1:Repeat for all states i

```
Vk+1(i) = min ( g(i) + disc * E  Vk(j) )
           u                  J ( u )
```

J(u) is the set of states that can be reached from state i.

```
        STEP2:Compute the bounds,

                   cku = max { 1/(1-disc)(Vk(i) - Vk-1(i)) }
                          i

                   ckl = min { 1/(1-disc)(Vk(i) - Vk-1(i) }
                          i

                  then if |cku-ckl|<EPS STOP else got step1


 }

program opt;

const
  disc=0.9; { The discount rate, epsilon convergence criteria,
  eps=1E-2; demaand rate, and maximum machine rates }
  d=1;
  r1=2;  r2=2;

type
 valuearray=array[-30..30,0..20,0..3]  of  real;
 array4I=array[0..3]  of  integer;
 array4R=array[0..3]  of  real;
 array4x4=array[0..3,0..3]  of  real;
 string12=string[12];

var
 outfile:text;
 filename:string12;
 value,valuek:valuearray;  {Value function arrays}
 alpha1,alpha2:array4I;  {Translates machine state}
 p:array4x4;   {Machine state transition matrix}

 i,j,k,l,m,n,iter,u1min,u2min,xk1,yk1:integer;

 cp1,cp2,cn2:real;    {Buffer  costs}
 pr1,pr2,pf1,pf2:real;  {repair rates, failure rates}
 g,min,cku,ckl,temp:real;

begin
{READ IN VALUES FOR MACHINE PARAMETERS AND BUFFER COSTS}

  filename:='input';
  assign(outfile,filename);
  reset(outfile);

  readln(outfile,pr1,pr2,pf1,pf2,cp1,cp2,cn2);
  close(outfile);
```

```
{THESE TRANSLATE GLOBAL MACHINE STATE TO INDIVIDUAL MACHINE STATE}

  alpha1[0]:=0;  alpha2[0]:=0;
  alpha1[1]:=0;  alpha2[1]:=1;
  alpha1[2]:=1;  alpha2[2]:=0;
  alpha1[3]:=1;  alpha2[3]:=1;

{DEFINE MACHINE STATE TRANSITION MATRIX}

  p[0,0]:=(1-pr1)*(1-pr2);  p[0,1]:=(1-pr1)*pr2;
  p[0,2]:=pr1*(1-pr2);   p[0,3]:=pr1*pr2;

  p[1,0]:=(1-pr1)*pf2;   p[1,1]:=(1-pr1)*(1-pf2);
  p[1,2]:=pr1*pf2;    p[1,3]:=pr1*(1-pf2);

  p[2,0]:=pf1*(1-pr2);   p[2,1]:=pf1*pr2;
  p[2,2]:=(1-pf1)*(1-pr2);  p[2,3]:=(1-pf1)*pr2;

  p[3,0]:=pf1*pf2;    p[3,1]:=pf1*(1-pf2);
  p[3,2]:=(1-pf1)*pf2;   p[3,3]:=(1-pf1)*(1-pf2);

{INITIALIZE VALUE FUNCTION AT ZERO}

  for i:=-30 to 30 do
    for j:=0 to 20 do
      for k:=0 to 3 do
        value[i,j,k]:=0.0;

  iter:=0;

{THIS IS THE MAIN LOOP}

  repeat
    cku:=0;
    ckl:=1E20;

    for i:=-30 to 30 do
    for j:=0 to 20 do
    for k:=0 to 3 do
        begin
          min:=1E20;
          if i<0 then g:=cp1*j-cn2*i else g:=cp1*j+cp2*i;
                                       {Cost function}

               {Min. over all possible controls}

          for m:=0 to alpha1[k] do
            for l:=0 to alpha2[k] do
```

```
              begin
                if  r2*l>j+r1*m  then  l:=0;
                             {If  buffer<0  do  not  allow  it}
                xk1:=j+r1*m-r2*l;
                yk1:=i+r2*l-d;

                    {Cost  approximation  at  -30}

                if  (yk1<-30)  and  (l=0)  then
                  begin
                   yk1:=-30;
                   g:=g+cn2*d*r2/(2*(pr2*pr2*(r2-d)));
                     {Add  cost  of  a  mean  path  to  recovery}
                  end;
                if  yk1>30  then  {Upper  boundry.}
                  begin
                   yk1:=30;
                   l:=0;
                  end;

                if  xk1>20  then  {Buffer  upper  boundry}
                 begin
                   xk1:=20;
                   m:=0;
                  end;

                temp:=0.0;  {Now  do  the  minimization}
                for  n:=0  to  3  do
                   temp:=temp+p[k,n]*value[yk1,xk1,n];
                if  temp<min  then
                      min:=temp;
              end;

              {Generate  new  value  function}

           valuek[i,j,k]:=g  +  disc*min;

{Calculate  the  error  bounds}

           if  (disc/(1-disc)*(valuek[i,j,k]-value[i,j,k]))>cku  then
             cku:=(disc/(1-disc)*(valuek[i,j,k]-value[i,j,k]));

           if  (disc/(1-disc)*(valuek[i,j,k]-value[i,j,k]))<ckl  then
             ckl:=(disc/(1-disc)*(valuek[i,j,k]-value[i,j,k]));

        end;
```

```
{Update Value Function}

   for i:=-30 to 30 do
    for j:= 0 to 20 do
     for k:=0 to 3 do
         value[i,j,k]:=valuek[i,j,k];


   iter:=iter+1;
   writeln(iter,'   ',cku,'   ',ckl);


   until abs(cku-ckl)<eps; {If bounds have converged, stop}

 { ITERATIONS HAVE CONVERGED, NOW COMPUTE CONTROLS AND SAVE THEM }

 filename:='output';
 assign(outfile,filename);
 rewrite(outfile);


{ COMPUTE THE CONTROLS BASED ON THE CURRENT VALUE FUNCTION }

 for i:= -30 to 30 do
   for j:=0 to 20 do
     for k:= 0 to 3 do
       begin
          min:=1E20;
          value[i,j,k]:=value[i,j,k]+(cku+ckl)/2;
             {Use bounds to improve value function estimate}

          for m:=0 to alpha1[k] do
            for l:=0 to alpha2[k] do
              begin
                if r2*l>j+r1*m then l:=0;
                xkl:=j+r1*m-r2*l;
                ykl:=i+r2*l-d;

                if ykl<-30 then
                   ykl:=-30;
                if ykl>30 then
                   ykl:=30;
                if xkl>20 then
                   xkl:=20;

                temp:=0.0;
                for n:=0 to 3 do
                   temp:=temp+p[k,n]*value[ykl,xkl,n];
```

```
                    if  temp<min  then
                        begin
                          min:=temp;
                          u1min:=m;
                          u2min:=1;
                        end;
                  end;

        write(outfile,2*u2min+u1min,'  '); {This our code for the control}
      end;
 close(outfile);
 writeln('File  ',  filename,'  created.');
end.
```

APPENDIX C: TWO-MACHINE TRANSFER LINE FORMULAE AND ONE-MACHINE, ONE-PART-TYPE FORMULAE

This appendix contains the results of Gershwin and Schick (1980) for a two-machine, continuous flow transfer line and the results of Akella and Kumar (1986) for a one-machine, one-part-type system. The transfer line formulae are need to evaluate $\bar{B}(B^H)$ and $\bar{u}(B^H)$ in the minimization (5.33). The one-machine, one-part-type formulae are needed to determine the hedging point, x_2^H, form Section 5.4.1, where we consider Machine 2 in isolation with repair rate, r, and failure rate p' given by (5.35).

Two-Machine Transfer Line

The model of the two-machine transfer line is shown in Figure 5.2. This is a special case of the problem addressed by Gershwin and Schick (1980), who considered machines with different machine speeds and reliability parameters.

Both machines produce at rate 1 when they are operational. The buffer level, B(t), is bounded by

$$0 \le B(t) \le B^H \tag{C.1}$$

because B^H is the buffer hedging point for Machine 1, and, therefore, Machine 1 will not produce if B(t) exceeds B^H. If the buffer is empty, Machine 2 produces at the maximum rate when Machine 1 is operational and at rate zero otherwise. Similarly, if $B(t)=B^H$, Machine 1 produces at the maximum rate when Machine

2 is operational, and at rate zero otherwise.

The machines have Markovian failures and repairs. Machines can only fail when they are producing. Let $\alpha_i(t)=1$ if Machine i is up at time t and $\alpha_i(t)=0$ if Machine i is down at time t. Then

$$P[\ \alpha_i(t+\delta t)=1 \mid \alpha_i(t)=0\ ] = r_i \quad i=1,2 \tag{C.2}$$

$$P[\ \alpha_1(t+\delta t)=0 \mid \alpha_1(t)=1,\ B(t)\neq B^H \text{ if } \alpha_2(t)=0] = p_1 \tag{C.3}$$

$$P[\ \alpha_2(t+\delta t)=0 \mid \alpha_2(t)=1,\ B(t)\neq 0 \text{ if } \alpha_1(t)=0] = p_2 \tag{C.4}$$

The state of the system is given by the values of B(t), $\alpha_1(t)$ and $\alpha_2(t)$. The steady state distribution of the system consists of a probability mass at $B=0$ and $B=B^H$ and a probability distribution for $0<B<B^H$. The probability mass is denoted by $P(B,\alpha_1,\alpha_2)$ while the probability distribution is denoted $f(B,\alpha_1,\alpha_2)$. The following values are derived by Gershwin and Schick for $P(B,\alpha_1,\alpha_2)$ and $f(B,\alpha_1,\alpha_2)$:

$$P(0,0,0) = 0 \tag{C.5}$$

$$P(0,0,1) = C\,\frac{r_1 + r_2}{r_1\ p_2} \tag{C.6}$$

$$P(0,1,0) = 0 \tag{C.7}$$

$$P(0,1,1) = \frac{C}{p_2} \frac{r_1 + r_2}{p_1 + p_2} \tag{C.8}$$

$$p(B^H,0,0) = 0 \tag{C.9}$$

$$P(B^H,0,1) = 0 \tag{C.10}$$

$$P(B^H,1,0) = C \; e^{\lambda B^H} \frac{r_1 + r_2}{p_1 \; r_2} \tag{C.11}$$

$$P(B^H,1,1) = \frac{C}{p_1} \; e^{\lambda B^H} \frac{r_1 + r_2}{p_1 + p_2} \tag{C.12}$$

$$f(B,\alpha_1,\alpha_2) = C \; e^{\lambda B} \; Y_0^{\alpha_1+\alpha_2} \tag{C.13}$$

where

$$\lambda = - (p_1 r_2 - r_1 p_2) (\frac{1}{p_1 + p_2} + \frac{1}{r_1 + r_2}) \tag{C.14}$$

$$Y_0 = \frac{r_1 + r_2}{p_1 + p_2} \tag{C.15}$$

and C is a normalization constant chosen to make

$$\sum_{\alpha_1=0,1} \sum_{\alpha_2=0,1} \left\{ P(0,\alpha_1,\alpha_2) + P(B^H,\alpha_1,\alpha_2) + \int_0^{B^H} f(x,\alpha_1,\alpha_2) \, dx \right\} = 1 \tag{C.16}$$

The following performance measures using the notation in Chapter 5 are based on the steady state distribution

Average Production Rate $\bar{u}(B^H)$

$$\bar{u}(B^H) = \sum_{\alpha_1=1} \left\{ P(B^H,\alpha_1,\alpha_2) + \int_0^{B^H} f(B,\alpha_1,1) \, dB \right\} \tag{C.17}$$

Average Buffer Level $B(B^H)$

$$B(B^H) = \sum_{\alpha_1=0,1} \sum_{\alpha_2=0,1} \left\{ B^H \; P(B^H,\alpha_1,\alpha_2) + \int_0^{B^H} B \; f(B,\alpha_1,\alpha_2) \; dB \right\} \qquad (C.18)$$

Starvation occurs when Machine 2 is up but the buffer is empty and Machine 1 is down. The probability of starvation, p_s, is therefore

$$p_s = P(0,0,1) = C \frac{r_1 + r_2}{r_1 \; p_2} \qquad (C.20)$$

One-Machine, One-Part-Type System

We can use the following hedging point values given by Akella and Kumar (1986) for a one-machine, one-part-type system as described in Section 5.4.1. A single machine with infinite supply has a production rate, u(t), a constant demand d and Markovian failure and repairs, with failure rate p' given by (5.35) and repair rate r. The machine produces at a maximum rate of 1 while operational. The cost function to be minimized is

$$E \int_0^{\infty} e^{-\beta t} (\, c^+ \; x^+(t) + c^- \; x^-(t) \,) \; dt \qquad (C.21)$$

The optimal policy is shown to be,

$$u(t) = 1 \quad \text{if } x(t)<x^h \tag{C.22}$$

$$u(t) = 0 \quad \text{if } x(t)>x^h \tag{C.23}$$

$$u(t) = d \quad \text{if } x(t)=x^h \tag{C.24}$$

where the optimal hedging point, x^h, is given by

$$x^h = \max \left\{ 0, \frac{1}{\lambda_-} \ln \left[\frac{c^+}{c^+ + c^-} \left(1 + \frac{\beta \ d}{p'd-(\beta+r+d\lambda^-)(1-d)} \right) \right] \right\} \tag{C.25}$$

where λ^- is the only negative eigenvalue of the matrix

$$A = \begin{bmatrix} \frac{\beta+p'}{1-d} & \frac{-p'}{1-d} \\ \frac{r}{d} & -\left(\frac{r+\beta}{d}\right) \end{bmatrix} \tag{C.26}$$

Evaluation x^h for Machine 2 as described in section 5.4.1 will given a approximation to the hedging point x_2^h.

REFERENCES

Akella R., Choong Y. and Gershwin, S. B., "Performance of Hierarchical Production Scheduling Policy", IEEE Transactions on Components, Hybrids and Manufacturing Technology, Vol. CHMT-7, No. 3 (September 1984).

Akella, R. and Kumar P. R., "Optimal Control of Production Rate in a Failure Prone Manufacturing System", IEEE Transactions on Automatic Control, Vol. AC-31, No. 2 (February 1986).

Bertsekas, D. P., Dynamic Programming and Stochastic Control, Academic Press, 1976.

Bielecki, T., and Kumar, P. R., "Optimality of Zero-Inventory Policies for Unreliable Manufacturing Systems", Coordinated Science Laboratory, University of Illinois Working Paper.

Buzacott, J. A., "'Optimal' Operating Rules for Automated Manufacturing Systems", IEEE Transactions on Automatic Control, Vol. 27, No. 2 (February 1982).

Buzacott, J. A. and Yao, D., "Flexible Manufacturing Systems; A Review of Models", TIMS/ORSA Detroit Meeting (April 1982).

Donoghue, J. F., "A Comparison of the Smith Predictor and Optimal Design Approaches for Systems with Delay in the Control", IEEE Transactions on Industrial Electronics and Control Instrumentation, Vol, IECI-24, No. 1 (February 1977).

Ephremides, A., Varaiya, P. and Warland, J., "A Simple Dynamic Routing Problem", IEEE Transactions on Automatic Control, Vol. AC-25, No. 4 (August 1980).

Gershwin, S. B., Akella, R. and Choong, Y. F., "Short-Term Production Scheduling of an Automated Manufacturing Facility", IBM Journal of Research and Development, Vol. 29, No. 4 (July 1985).

Gershwin, S. B., Hildebrant, R. R., Suri, R. and Mitter, S. K., "A Control Perspective on Recent Trends in Manufacturing Systems", *IEEE Control Systems Magazine*, Vol. 6, No. 2, (April 1986).

Gershwin, S. B., and Schick, I. C., "Continuous Flow Model of an Unreliable Two-Stage Material Flow System with a Finite Interstage Buffer," MIT LIDS Report #LIDS-R-1039 (1980).

Hahne, E. "Dynamic Routing in an Unreliable Manufacturing Network with Limited Resources", MIT LIDS Report No. LIDS-TH-1063 (1981).

Halanay, A. *Differential Equations, Stability, Oscillations, Time-Lags*, Academic Press, 1966.

Hess, R. A., "Optimal Control Approximations for Time Delay Systems", *AIAA Journal*, Vol. 10, No. 11 (November 1972).

Hess, R. A. and Hyde, J.G., "Suboptimal Control of Time-Delay Sytems", *IEEE Transactions of Automatic Control*, (December 1973).

Kimemia, J. G., "Hierarchical Control of Production in Flexible Manufacturing Systems", MIT LIDS Report No. LIDS-TH-1215 (1982).

Kimemia, J. and Gershwin S. B., "An Algorithm for the Computer Control of a Flexible Manufacturing System", *IIE* Transactions, Vol. 15, No. 4 (December 1983).

Koivo, A. J., "Certainty Equivalence in Stochastic Linear Time-Lag Systems", *Proceedings of the IEEE*, (November 1969), pp. 2061-2062.

Little, J.D.C., "A Proof of the Queuing Formula: $L=\lambda W$," *Operations Research*, Vol. 9, No. 3 (1961).

Mariani, L. and Nicolleti, B., "Optimal Discrete Systems with Pure Delay", *IEEE Transactions on Automatic Control*, Vol AC-18, No. 3 (June 1973).

Nag, Amal "Auto Makers Discover 'Factory of the Future' is Headache Just Now", The Wall Street Journal, Tuesday, (May 13, 1986), Front Page.

Oguztoreli, M. N., Time Lag Control Systems, Academic Press, (1966).

Olsder, G. J. and Suri, R., "Time Optimal Control of Parts-Routing in a Manufacturing System with Failure Prone Machines", Proceedings of the 19th IEEE Conference on Decision and Control, Alburquerque, New Mexico (1980).

Orlicky, J., Materials Requirements Planning, McGraw-Hill (1975).

Repin, I. M., "On the Approximate Replacement of Systems with Lag by Ordinary Dynamical Systems", Journal of Applied Mathematics and Mechanics, Vol. 29, No. 2 (1965).

Rishel, R. "Dynamic Programming and Minimum Principles for Systems with Jump Markov Disturbances", SIAM Journal on Control, Vol.13, No.2 (Feburary 1975).

Ross, D. W. and Flugge-Lotz, I., "An Optimal Control Problem for Systems with Differential-Difference Equations Dynamics", SIAM Journal on Control, Vol.7, No.4 (November 1969).

Seidman, A. and Schweitzer, P. J., "Real-Time Control of a FMS Cell", Paper QM8217, Graduate School of Management, University of Rochester, NY (1982).

Smith, O. J. M., "Closer Control of Loops with Dead Time", Chemical Engineering Progress, Vol. 53, No. 5 (May 1957).

Stidham, S., "Optimal Control of Admission to a Queueing System", IEEE Transactions on Automatic Control, Vol. AC-30, No.8 (August 1985).

Tsitsiklis, J., "Optimal Dynamic Routing in Unreliable, Continuous Queueing Networks", MIT LIDS Report No. LIDS-TH-1069 (1980).