April, 1980 ESL-FR-834-2

COMPLEX MATERIALS HANDLING AND ASSEMBLY SYSTEMS

Final Report

June 1, 1976 to July 31, 1978

Volume II

Multicommodity Network Flow Optimization in Flexible Manufacturing Systems

by

Joseph Githu Kimemia and Stanley B. Gershwin

This report is based on the thesis of Joseph Githu Kimemia, submitted in partial fulfillment of the requirements of Master of Science at the Massachusetts Institute of Technology in January, 1979. Thesis supervisors were Dr. S. B. Gershwin, Lecturer, and Professor M. Athans, Department of Electrical Engineering and Computer Science. The research was carried out in the Laboratory for Information and Decision Systems with partial support extended by National Science Foundation Grants NSF/RANN APR76-12036 and DAR78-17826.

Laboratory for Information and Decision Systems
(formerly Electronic Systems Laboratory)
Massachusetts Institute of Technology
Cambridge, MA 02139

ABSTRACT

The problem of choosing an optimal mix of operating strategies in a flexible manufacturing system is solved by a network flow optimization approach. Mathematical methods which exploit the structure of the problem to generate manufacturing strategies are outlined. Numerical results show that the method produces results which agree with intuition and simulation for two- and four-workstation systems.

ACKNOWLEDGMENTS

Acknowledgements are due to several people who made this research possible. Thanks go to Professor Michael Athans for his comments, suggestions, and discussion. The contribution of the members of the Manufacturing Group at L.I.D.S., in particular John Ward for the CAN-Q results, G. Secco-Suardo and Konrad Hitz for discussion on the network of queues models and the scheduling problem, respectively, and Yehiam Horev for the simulation, is much appreciated.

In the work on the augmented Lagrange Multiplier algorithm,thanks go to Dr. Earl Barnes of I.B.M. for his help both in the theory and the coding of the algorithm. Comments made by Professor W. Maxwell of Cornell University are much appreciated.

Thanks finally go to Arthur Giordani for the drafting and the L.I.D.S. typists for their work in the preparation of the document.

TABLE OF CONTENTS

TABLE OF CONTENTS (cont'd)

LIST OF FIGURES

LIST OF FIGURES (cont'd)

LIST OF FIGURES (cont'd)

LIST OF TABLES

1. INTRODUCTION

1.1 The Strategy Assignment Problem in Flexible Manufacturing Systems

A large proportion of manufacturing activity is at a level which does not justify dedicated automation in the form of single-product machines or lines. In order to increase productivity in this sector of industry, flexible manufacturing systems are being designed and built.

A flexible manufacturing system such as the one depicted in Fig. 1.1, consists of workstations capable of performing a number of different tasks, interconnected by a transportation system. Workpieces are loaded onto pallets at a loading station, undergo a specified sequence of operations at the workstations, and then go to an unloading station. The processes at the workstations are mostly automatic. At certain stations, like the loading station for example, some manual operations may be performed (Hughes, 1977).

Several different kinds of pieces are manufactured simultaneously in the system. Each piece has a given number of operations necessary for its completion, as shown for example, in the piece of Fig. 1.2. There is a choice in the system as to which workstation should perform each operation. Any entering workpiece therefore has the choice of several different routes or manufacturing strategies available. A strategy for each piece assigns each operation to a workstation with the capability of performing that operation. The strategy also specifies the sequence of workstation visits.

In order to gain maximum output and utilization at minimum cost, the overall behavior of the system should be studied. Furthermore, mathematical models and algorithms are needed which will enable controllers to make decisions affecting the system with minimum human intervention.

An important problem, which has a fundamental effect on the production rate and utilization of the system, is the assignment of strategies to the workpieces. Given a flexible manufacturing system with a specified production mix of pieces and given the locations at which all the operations can be performed in the system, one wishes to pick the optimal steady-state mix of strategies for all of the pieces being produced.

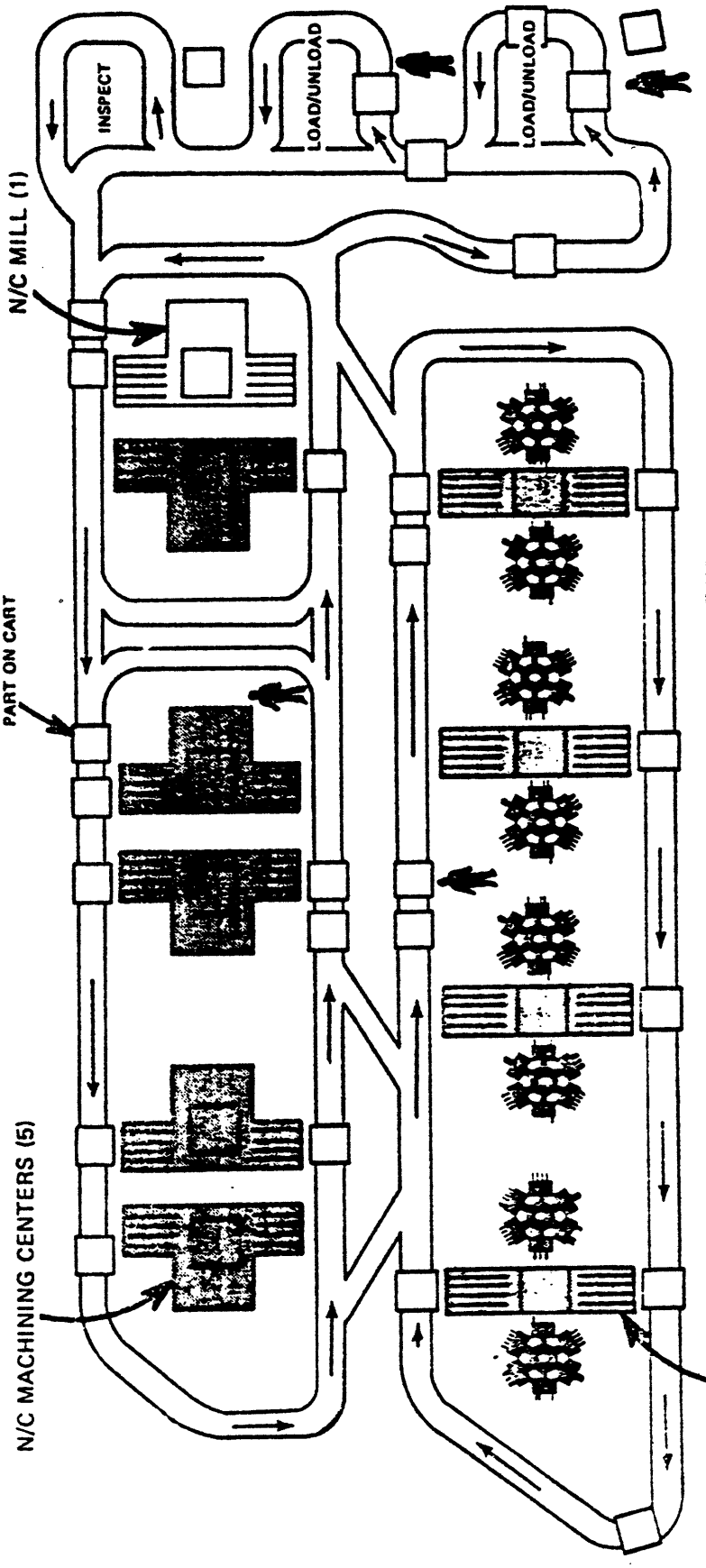


Fig. 1.1. An Example of a Flexible Manufacturing System

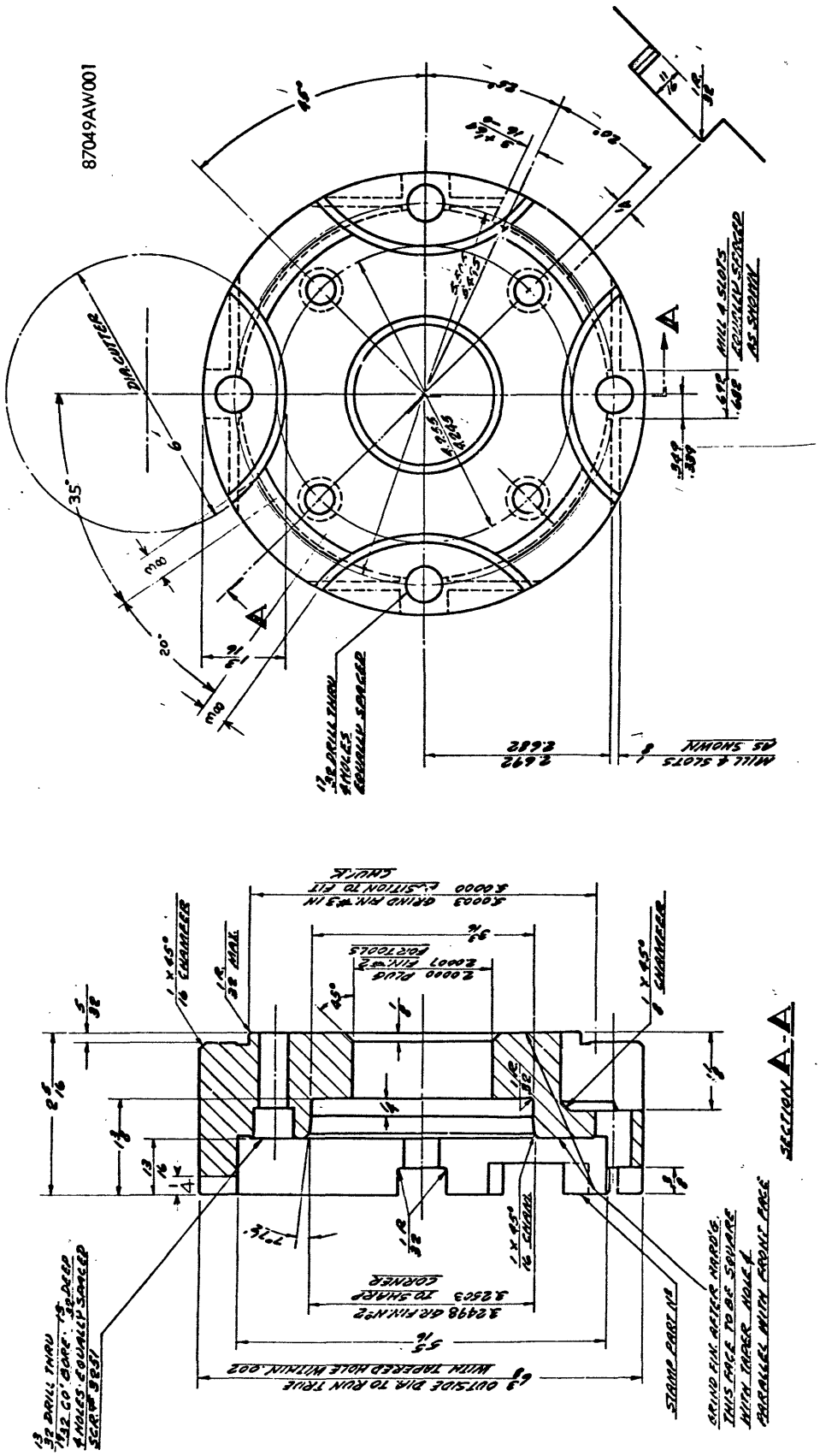


Fig. 1.2. Operational Requirements for a Machine Tool Chuck

Extensive simulation studies of flexible manufacturing systems have been made (Hutchinson, 1977) (Horev et al., 1978) (Lenz and Talavage, 1977). They allow detailed investigation of the effects of parameter variation and strategy assignment on system performance.

Solberg (1977) and Ward (1980) model the system as a closed network of queues. Steady state results which are in good agreement with simulation results and observed performance of an actual system are obtained. The use of the closed network of queues model as an analytic method of strategy assignment has been suggested by Secco-Suardo (1978).

The machine or job shop problem has had considerable attention in the past and is in the class of combinatorial problems. They can be formulated and solved as 0-1 integer programming problems (Stern et al., 1977) (Fisher, 1970). In the case of a flow shop where the jobs must undergo a sequence of operations, the solution is difficult even for a three-machine system (Kanellakis, 1978). A particular difficulty with this approach is that it makes an optimal schedule for a given number of jobs. What is required is a method of calculating optimal strategy assignments for a system that is operating continuously.

1.2 The Network Flow Optimization Approach

In this research, a network flow optimization approach is taken. Rather than analyze the movement of individual pieces through the system, the aggregated flow of pieces is analyzed. Network of queues models are used to account for congestion effects at the workstations.

Flow optimization techniques have been successfully applied to transportation and computer communication problems. In transportation systems, a frequently occuring problem is that of predicting traffic flows on a network of roads given travel demand between origin-destination pairs in the network. The solution is given by Wardrop's Principle; traffic distributes itself on the available routes in such a way that no single user can shorten his or her travel time or cost by using another route. For this reason it is often referred to as "user optimized flow" (Dafermos and Sparrow, 1969). A related problem but with a different solution is the system optimization

problem (Dafermos and Sparrow, 1969). In this case, given the travel demand between the various origin-destination pairs, one wishes to route the traffic in such a way that some system cost criterion is minimized.

Problems occur in computer systems where the computers are connected by data links as in the ARPA-network. Messages are routed from origins to destinations via intermediate computers. Each message experiences a random delay which is on the average a non-linear function of the flow rate (usually measured in bits per second) on a link. The objective is to route the messages in such a way that the total overall delay is minimized. This problem has been formulated and successfully solved as a non-linear network flow optimization problem (Frank and Chou, 1971).

Multi-commodity, minimum-cost, network flow optimization problems with resource constraints at network nodes have been examined by Wollmer (1972), Malek-Zavarei and Frisch (1971). Resource constrained problems occur, for example, in transportation problems with a limited number of vehicles or communication problems where there are capacity constraints at network nodes. Decomposition methods have been applied to solve such problems. The workstations in flexible manufacturing systems can be viewed as scarce resources to be shared amongst all the types of pieces in the system. Similar methods can then be used to decompose the problem into easily solved sub-problems.

1.3 An Outline of the Report

The model is presented and the optimization problems formulated in Chapter 2. Systems having nondeterministic arrivals and processing times give rise to non-linear optimization problems. The production rate of the system should be maximized but the build up of queues within the system should be avoided. A price can be put on the average number of pieces within the system (the in-process inventory). Alternatively the inventory can be constrained to be below a certain given value. Deterministic systems, or systems in which the processing and interarrival times have a small variance, give rise to linear programs. Asymptotic results for closed queueing network models (Gordon and Newell, 1967) (Secco-Suardo, 1978) and work rate theorems (Chang and Lavenberg, 1972) indicate that the linear programs

are valid for finding the asymptotic maximum production rate in systems with general service time distributions.

Mathematical methods which exploit the structure of the problem in order to solve the optimization problems of Chapter 2 are discussed in Chapter 3. Decomposition methods (Dantzig, 1963) are used to break linear programs into a set of strategy-generating minimum processing cost subproblems each involving only one type of workpiece. Only a subset of all the possible manufacturing strategies are considered and they do not have to be enumerated in advance. A master problem finds the optimal combination of strategies for all the pieces.

An extremal flow algorithm (Cantor and Gerla, 1974) (Defenderfer, 1977) minimizes non-linear objective functions subject to linear constraints by expressing the network flow rates as a convex combination of extremal flows. The extremal flows are generated by solving a linear program at each step. This method was originally developed for solving routing problems in packet switched computer networks (Cantor and Gerla, 1974) and has proven to be an effective method of obtaining the optimal routing in a network (Defenderfer, 1977). The Lagrange multiplier method of Hestenes (1969) and Powell (1968) converts a non-linearly constrained optimization problem into a series of problems where a non-linear Lagrangian function is minimized subject to the linear flow and resource conservation constraints. The extremal flow algorithm can then be used to minimize the Lagrangian function.

As an example of the application of the network flow approach to the strategy assignment problem, numerical results for a two- and four-workstation system are presented. The effect of changing some of the system parameters on the optimal strategy assignment, production rate and workstation utilization is investigated for the two-workstation system. The strategy assignments for the four-workstation system are implemented on a discrete simulation and the effects observed.

There are a number of outstanding problems for which analytic solution techniques would be extremely useful. Chapter 5 identifies problems for which network flow optimization appears to be promising as a component of a solution technique.

2. THE MODELLING OF FLEXIBLE MANUFACTURING SYSTEMS

2.1 Introduction

Accurate modelling of flexible manufacturing systems is important if an understanding of overall system behavior is to be gained. Of even greater importance is the building of models which will enable computers to make decisions either on- or off-line when running the system under automatic control.

On a system-wide level, the static optimization problem is concerned with the steady state behavior of the system. The average values of utilization of the workstations, queue lengths at workstations, flow rates on the transportation links and the in-process inventory are of interest and define the state of the system.

Preliminary investigation is being carried out on small two- to four-workstation simulated systems. Practical systems will be much larger. The Sundstrand system at the Caterpiller plant at Peoria, Illinois, for example has nine workstations, sixteen dual loading/unloading stations and produces two sizes of gear box casings, each consisting of two parts (Stecke, 1977). The size of the system gives rise to models with large numbers of variables. Care must be taken in keeping the dimension of the model to a minimum. In Section 2.2 flexible manufacturing systems are modelled as networks of queues. Exact solution methods which have been applied to models of actual systems are surveyed. These methods are restricted to system models which satisfy certain assumptions regarding service time distributions and arrival processes. Approximate methods are introduced for application to more general models. Optimization problems based on networks of queues are formulated in Section 2.3.1.

Section 2.3.2 formulates linear programming problems for systems whose service times are either deterministic or have small variances. In this case the non-linearities which account for the build up of queues are absent. The flow rates in the system are then the only variables of concern. Section 2.4 develops an approximation to the production rate of a balanced system with a finite number of pallets. Some aspects of the optimal solution of the programming problem are discussed in Section 2.5.

2.2 The Stochastic Model

A network of queues consists of M nodes at which there are one or more servers. In a flexible manufacturing system these would correspond to the workstations and the loading and unloading stations. The service time at station i is taken to be a random variable with a known probability density function and mean $1/\mu_i$. In a manufacturing system there are different types of workpieces each with its own service time distribution at each workstation. In most practical cases, the ratio of the numbers of different types of pieces being produced is specified.

It is assumed that once a workpiece leaves workstation i, it proceeds to workstation j with probability p_{ij}. Workpieces originating from outside the system arrive at workstation i at a rate a_i. The arrival process is stochastic with known statistical properties. The arrival rate λ_j at workstation j thus satisfies

$$\lambda_j = a_j + \sum_{i=1}^{M} p_{ij}\lambda_i \tag{2.1}$$

The probability that a workpiece leaves the system after the completion of service at workstation i is simply $1 - \sum_{j=1}^{M} p_{ij}$.

A network of queues is described as open if there are arrivals and departures to and from outside the network (Baskett et al., 1975). If, in equation (2.1), $a_i=0$ and $\sum_j p_{ij} = 1$ for all i, the system is closed. In this case there are N jobs circulating inside the network with none leaving and no fresh arrivals. The arrival rates λ_j then satisfy

$$\lambda_j = \sum_{i=1}^{M} p_{ij}\lambda_i \tag{2.2}$$

The matrix $p=(p_{ij})$ represents transitions in an underlying ergodic Markov chain (Baskett et al., 1975). With non-zero values of a_i, (2.1) can be solved to give unique values of λ_i. Equation (2.2) however, consists of self-consistent equations which can only be solved to within a multiplicative constant.

2.2.1 Exact Solution of Network of Queues Models

The open network was originally studied by Jackson (1963). The assumptions made were that the service time distribution at all nodes is exponential and that the arrival process from outside the network is Poisson. It is also assumed that there is only one class of customers. Under these assumptions and also given that there is unlimited queueing space at all the nodes, the system can be modelled as an infinite (but countable) state Markov process. Each state is defined by the vector $k=(k_1,k_2,\ldots k_M)$ where k_i is the number of customers either receiving or awaiting service at node i. Jackson's result is that the steady state limiting probability of being in any state k can be written in product form as

$$P(k) = p_1(k_1)p_2(k_2)\ldots p_M(k_M) \tag{2.3}$$

$p_i(k_i)$ is the marginal probability of having k_i customers at node i. The amazing thing is that $p_i(k_i)$ is identical to the steady state probability distribution of a single M/M/n queue. The implication of this result is that under the Poisson arrival, exponential service time assumptions, the variables k_i are mutually independent in the steady state and thus each queue may be analyzed in isolation. Gordon and Newell (1967) derived the steady state probability distribution for a closed network with N identical customers, and an exponential service time distribution at each of the M nodes. A finite state Markov model results. The number of states is equal to $\binom{N+M-1}{N-1}$ which is the number of ways that the N customers can be placed at the M nodes. A product form solution is again found with

$$P(k) = \frac{1}{G(M,N)} \prod_{i=1}^{M} f_i(k_i) \tag{2.4}$$

$$\text{and} \quad \sum_{i=1}^{M} k_i = N \tag{2.5}$$

in which G(M,N) is a normalizing constant. The functions $f_i(k_i)$ satisfy the flow balance equations of the Markov chain model of the system. In this case there is strong interaction among the system nodes through the relationship (2.5).

An important effect is the asymptotic behavior of a closed network of queues as the number of customers N inside the system grows without bound. Let x_i be an arbitrary solution to equation (2.2). If there are r_i servers at station i, each with service rate μ_i, the relative utilization $\bar{u}_i$ of each workstation is defined as

$$\bar{u}_i = \frac{x_i}{r_i \mu_i} \tag{2.6}$$

There exists one or more stations with $\bar{u}_s = \max_i \bar{u}_i$. These stations are termed bottleneck stations (Gordon and Newell, 1967) for the closed network. It is shown that at any state for which k_s, the number of customers at the bottleneck station, is finite,

$$\lim_{N \to \infty} P(k_1, k_2, \ldots k_M) = 0 \tag{2.7}$$

The marginal distribution $P_B(k_1, \ldots k_{s-1}, k_{s+1}, \ldots k)$ taken at all stations excluding the bottleneck stations is finite and well defined and takes the product form

$$P_B(k_1, k_2, \ldots k_M) = \prod_{\substack{i=1 \\ i \notin B}}^{M} p_i(k_i) \tag{2.8}$$

where B is the set of bottleneck stations. Thus as the number of customers inside the network becomes large, the bottleneck stations act as generators of Poisson arrivals. The rest of the network behaves like an open network (Secco-Suardo, 1978).

The analyses of Jackson, Gordon and Newell apply only to networks with exponential servers. Jackson also assumes external Poisson arrival processes.

Baskett et al., (1975) provide perhaps the most complete analysis of the equilibrium probability distribution for networks of queues. Any service time distribution with a rational Laplace transform is permitted subject to certain assumptions on the queueing discipline. Mixed classes of customers,for some of whom the network may be closed and others open,can exist. A product form solution is shown to exist for the balance equations of the Markov system. The state space is particularly large since at each workstation the class of customer at each position in each queue must be accounted for.

Let y_i be a vector with components n_{ir}, the number of class r customers at station i. The marginal probability distribution $P(y_1,y_2,\ldots y_M)$ has a product form given by

$$P(y_1,y_2,\ldots y_M) = C\, d(S) \prod_{i=1}^{M} g_i(y_i) \qquad (2.9)$$

where C is a normalizing constant and d(S) is a function of the state S of the system and is dependent on the nature of the external arrival process. In a network that is closed for all classes of customers, d(S)=1. The functions $g_i(y_i)$ depend only on the mean arrival and service rates at workstation i. For a single customer class they are identical to the $f_i(k_i)$ of equation (2.4).

In an open network with Poisson arrivals, the marginal probability distribution of the total number of customers at any node is independent of the number at the other nodes. It is identical to the M/M/1 probability distribution if there is a single server with general service time distribution and a queue discipline that starts service on a customer immediately upon arrival, and to the $M/G/\infty$ distribution when there are an infinite number of servers. A very surprising result.

The existence of the product form of solution is related to the nature of the flow processes inside the network. A sufficient condition for the product form to exist is that a network should satisfy local balance equations (Chandy et al., 1977), (Chandy, 1972) with respect to a state in the

Markov chain modelling the network and a particular node i. Local balance equations equate the state transition rate into a Markov model state due to an arrival of a customer at node i to the transition rate out of the state due to the departure of a customer from node i.

A closely related property is the "M ⇒ M" property (Chandy, 1972). A queue is said to have the "M ⇒ M" property if the departure process at a queue with a Poisson arrival process is also Poisson. This holds for queues with exponential servers.

Non-exponential servers satisfy local balance equations if they have a service discipline which begins service on a new customer immediately upon arrival. Thus the allowed service disciplines are last-come, first-served with pre-emption, and processor sharing. An infinite server station also satisfies this condition.

Network of queues models have been used to model time sharing computer systems (Kleinrock, 1976) and it is this field which has given rise to the interest in networks of queues. Flexible manufacturing systems have been successfully modelled as networks of queues (Solberg, 1977). Taking into account the number of assumptions which do not necessarily hold in actual systems, the accuracy of the network models is somewhat surprising. Denning and Buzen (1977) have suggested that the assumptions needed to define state transition probabilities as such in the Markov chain representing a network of queues may in fact be too strong. They derive similar expressions to those of Jackson, Gordon and Newell from an operational point of view. That is, rather than defining p(n) as a probability, they define it as the proportion of time the system spends in state n in an observation

period (0,T). This quantity is related to observed quantities like $A_i(n)$, the number of arrivals in (0,T) at station i when n customers are present, and $x_i(n)$, the number of service completions in the same period. They make no assumptions regarding service time distributions and arrival process characteristics. Their assumptions regarding the one-step behavior of the system--namely, that observable state changes are the result of the movement of single jobs either into or out of the system or between two nodes--is very similar to the local balance requirement of Chandy et al.,(1974).

2.2.2 Approximate Methods for the Analysis of Network of Queues Models

The exact methods discussed above are restricted to system models satisfying certain assumptions on service time distributions, arrival processes and queueing discipline. Exact solutions for more general systems are hard to obtain and in many cases they have not yet yielded to exact analysis (Kleinrock, 1976). What is needed are approximate methods which retain the qualitative behavior of actual systems and permit good estimates of the quantities of interest such as average queue lengths.

The accuracy of approximate methods is dependent on the methods used to model the flow processes within the network. The elements within the network are decomposition points where flows diverge, merges where there is convergence of flows and the actual servers themselves (Disney, 1975). A key simplifying assumption usually made is that arrivals and departures at network nodes constitute renewal processes. That is the time intervals between arrivals or departures are independent, identically distributed random variables.

For optimization purposes, a decomposition approach seems ideal. The results of Jackson (1963) show that an open network with exponential servers and Poisson arrivals can be exactly analyzed by looking at each node in isolation. Open networks with general service time distributions may likewise be analyzed so long as they satisfy the conditions of Baskett et al., (1975) and Chandy et al., (1977); namely, that the local balance equations must be satisfied. In general, however, open networks do not satisfy the conditions required to yield a product form solution and it is here that assumptions are made concerning flow processes in the network so as to apply

approximate methods.

Kuhn (1976) studies a network consisting of G/G/1 elements arbitrarily connected by considering the propagation of the mean and coefficient of variation C_i of the interarrival times in the network. The coefficient of variation is defined as

$$C_i = \sqrt{\left(\frac{E\{t_i^2\}}{\{E(t_i)\}^2}\right) - 1} = \frac{\sigma_{t_i}}{E(t_i)} \qquad (2.10)$$

where $E(t_i)$ = expected value (mean) of t_i

$E\{t_i^2\}$ = mean square value of t_i

σ_{t_i} = standard deviation of t_i

A heuristic expression which is exact for isolated M/G/1 systems is used to calculate the average waiting time and hence queue lengths at network nodes. The results of the decomposition approach are found to be close to observed simulation results for open networks.

Closed networks with exponential servers can be decomposed, depending on the relative magnitudes of the service rates at the nodes. This is useful in computer systems where, for example, the central processing unit might be much faster than the other devices (Courtois, 1975). The parametric method of Chandy et al., (1975) might prove to be useful in situations where the performance of a single workstation is of particular interest. They show that the behavior of a workstation in a closed system with exponential servers does not change if the rest of the network is replaced by a single composite queue with a service rate dependent on n, the numbers of customers in the composite queue. A necessary condition is that the network must satisfy local balance equations (Chandy et al., 1977) (Chandy, 1972). The method has been extended to give an iterative approximate method for general networks (Chandy et al., 1974).

The diffusion approximation uses the central limit theorem to approximate

N(t), the number of customers in a single queue by a continuous random variable x(t) whose propagation obeys the diffusion equation

$$\frac{\partial}{\partial t}\, p(x_o,x;t) = \frac{1}{2}\,\alpha\,\frac{\partial^2}{\partial x^2}\, p(x_o,x;t) - \beta\,\frac{\partial}{\partial x}\, p(x_o,x;t) \qquad (2.11)$$

where $p(x_o,x;t)$ is the probability density function of x(t) given an initial condition x_o, and α and β are the expected value and variance of the instantaneous change in x(t) which in this case are independent of x(t)

$$\alpha = \lim_{\Delta t \to 0} \text{var}\,(x(t+\Delta t) - x(t))/\Delta t \qquad (2.12)$$

$$\beta = \lim_{\Delta t \to 0} E\,(x(t+\Delta t) - x(t))/\Delta t \qquad (2.13)$$

The steady state solution of (2.11) taken in the limit as t becomes large is an explicit expression for $P_x(X) = \Pr(x \leq X)$ which is discretized by integrating over an appropriate interval to obtain $\hat{p}(n)$, the diffusion approximation of the probability of having n customers in the queue.

The boundary conditions used in solving the diffusion equation are very important. Kobayashi (1974), Kobayashi and Reiser (1974) impose reflecting barriers at the boundary x(t)=0 and thus their solutions are accurate during a busy period or for a queue whose utilization is close to unity. Gelenber (1975) assumes that once x(t) is at the boundary it remains there for an exponentially distributed time interval and then instantaneously jumps to some internal internal value with a given probability. This leads to a more accurate approximation, especially for a lightly loaded server. The constants α and β are chosen by assuming via the central limit theorem (Kleinrock, 1976), that N(t) may be approximated by the continuous random variable x(t) with mean $(\lambda-\mu)t$ and variance $(\mu^3 v_a - \mu^3 v_b)t$, where λ and μ are the mean arrival

and service rates, v_a and v_b are the variance of the interarrival and service times respectively, (Gelenbe, and Pujole, 1975).

The diffusion approximation has been applied to open networks of queues by considering a vector valued diffusion process (Kobayashi, 1974). A product form solution results. A simpler approach is to use the diffusion approximation to analyze each queue individually and to note that the arrival process at any queue is the superposition of departure processes from other queues and perhaps from outside the network (Gelenbe and Pujole, 1975)(Kobayashi, 1974).

Closed queueing systems have been analyzed using the diffusion approximation yielding product form solutions (Kobayashi, 1974). The decomposition of the closed network is made difficult by the fact that equation (2.2) does not have a unique solution and the distribution over a finite number of customers. This results in simultaneous equations to be solved, and normalizing constants which have to be evaluated. These difficulties are overcome by assuming that the number of customers inside the network is large and that a bottleneck station exists (Kobayashi, 1974) (Gelenbe and Muntz, 1976). The solution of the diffusion equation is then made to fit this asymptotic case.

The diffusion approximation is similar to the decomposition approach of Kuhn (1976) in that the behavior of the network of queues is taken to depend on the first and second moments of the stochastic processes.

2.3 Modelling and Optimization of Flexible Manufacturing Systems

2.3.1 Modelling of Systems With Stochastic Operation Times

A flexible manufacturing system consists of M workstations connected by a transportation system. There are P different types of pieces being produced simultaneously. Each piece of type i has S_i manufacturing strategies available to it. A strategy is simply a sequence of operations required to complete a workpiece. All together, there are S strategies enumerated in the system, with

$$S = \sum_{i=1}^{P} S_i \qquad (2.14)$$

The number S may be large if there are a large number of options available in the system so that it might not be worthwhile to identify all possible

strategies in advance.

For each piece of type i, the matrix T_i represents all possible manufacturing options. The elements of T_i are t_{ij}^k, the time to perform operation k at workstation j on a piece of type i. The number k represents a particular operation and does not imply that there are strict precedence constraints. As an example, consider the component in Fig. 2.1, which is an idealized representation of the housing for a two way hydraulic control valve. The part is made from a casting which has the correct external dimensions. The operations required are the drilling and tapping of holes to the required tolerances.

A flexible manufacturing system produces a family of such parts which are of different sizes, built to different tolerances and materials. It will be assumed for the sake of this example that the left hand edge is machined first. The part then goes to the loading station for re-fixturing before the right hand edge is machined. For modelling purposes, the left and right hand edges are identified as two distinct types of pieces each with its own T_i matrix.

For the left hand end of the part in Fig. 2.1, the following operations are identified. They are referred to by the superscript k in the variable t_{ij}^k.

k=1 : Drill and tap the four bolt holes

k=2 : Drill and bore valve chamber to required tolerance

k=3 : Drill axial passage

k=4 : Drill and tap outlet lines

k=5 : Drill and tap supply line

The definition of the operations is dependent on the capability of the machines and the distribution of tools amongst them. Operation 2, because of close tolerance requirements,may need a rough cut and then finishing which might not be done at the same machine. Drilling and tapping similarly may be done at two different machines. In this example, however, we will assume that each operation is completed during a single visit to a workstation.

87049AW030

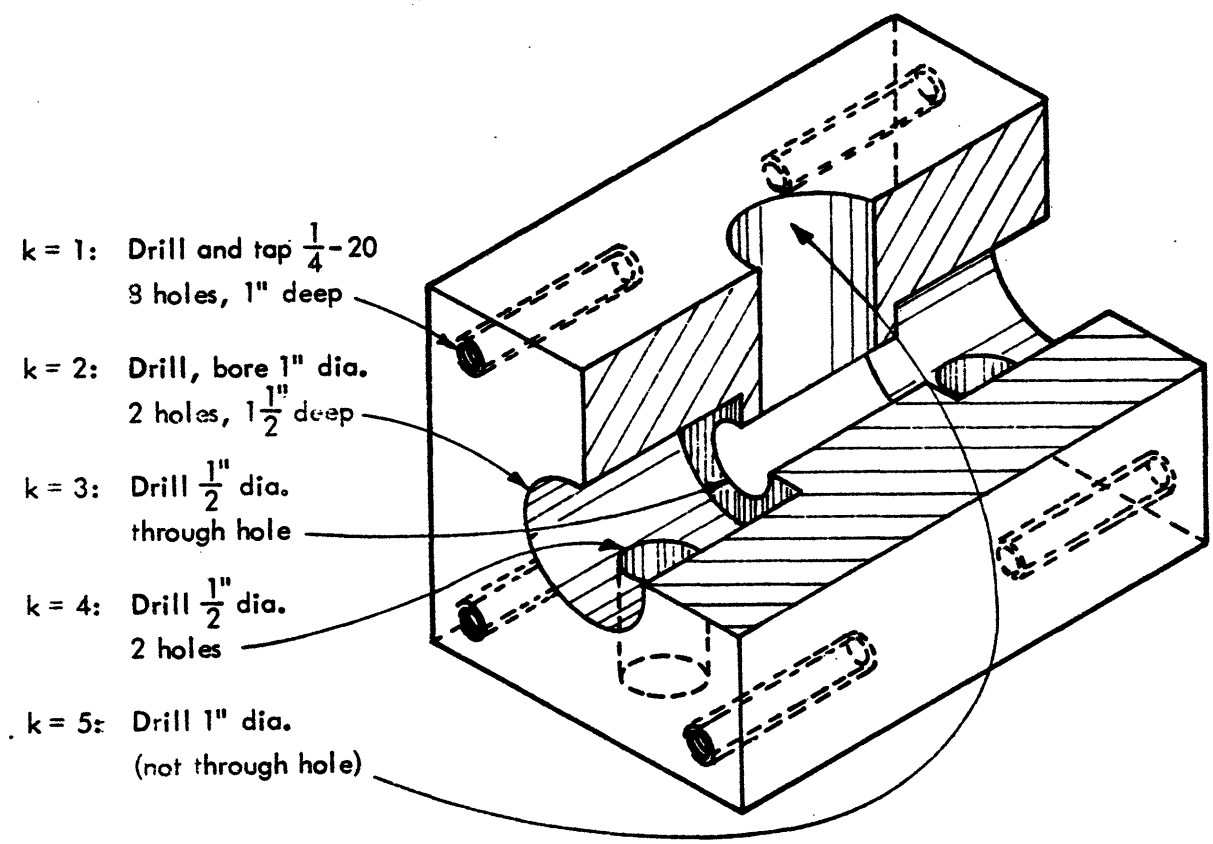


Fig. 2.1. An Example of a Workpiece

The only precedence constraints in this example are:

1. Operation 2 should precede both operations 3 and 5.

2. Operation 3 should precede operation 4.

Suppose there are four machines available to manufacture the valve housing. The figures in Table 2.1 show the locations at which the operations can be performed and the length of time in seconds that each operation takes. An entry of ∞ (infinity) indicates that the operation cannot be performed at that location. The top row gives the machine number and the column is the operation number. The element t_{ij}^{k} of the Matrix T_i will be the number in row k and column j of the table.

A strategy is a single sequence of workstation visits in which all necessary operations are performed. Two possible strategies are shown in Table 2.2. In strategy 1, operations are performed in the order 1-2-3-5-4, while the order is 1-2-4-5-3 for strategy 2.

Chapter 3 describes a method of generating strategies during the solution of the optimization problems formulated below.

If the strategies are enumerated in advance, the variable τ_{ij} represents the total time a piece following strategy 1 spends at workstation j. In the example above, the variables τ_{ij} for strategy 1 are $\tau_{i1} = t_{i1}^{1} = 15$, $\tau_{i2} = t_{i2}^{2} + t_{i2}^{3} = 55$ and $\tau_{i3}^{4} = t_{i3}^{5} + t_{i3} = 55$. Workstation 4 is not used, hence $\tau_{i4} = 0$. A graphical representation of strategies 1 and 2 of Table 2.2 is shown in Figure 2.2. The number in the circle is the workstation number and the one underneath is the duration of the visit. Above each circle are the particular operations being performed. The initial node L and the final node U are the loading and unloading stations, respectively.

Since the operation of the workstations is of primary concern, it will be assumed that the transportation system has a large enough capacity so that it does not reduce the performance of the system. After the important relationships affecting the performance of the workstations are introduced, it will be shown that the transportation system can be easily modelled using the same ideas.

Assuming that the matrices T_i are available for all workpieces, the flow rate of type i pieces to workstation j for operation k is defined as x_{ij}^{k}. The

k-Operation \ j-Workstation	1	2	3	4
1	15	∞	∞	70
2	∞	30	∞	40
3	∞	25	20	∞
4	45	∞	30	∞
5	40	∞	25	∞

Table 2.1 Machining Times t_{ij}^{k} for Operations at the Workstations on Part i, the Valve Housing

	Strategy 1		Strategy 2	
Visit	Operation	Machine	Operation	Machine
1	1	1	1	1
2	2	2	2	2
3	3	2	3	3
4	5	3	4	3
5	4	3	5	1

Table 2.2 Two Possible Strategies for the Manufacture of the Valve.

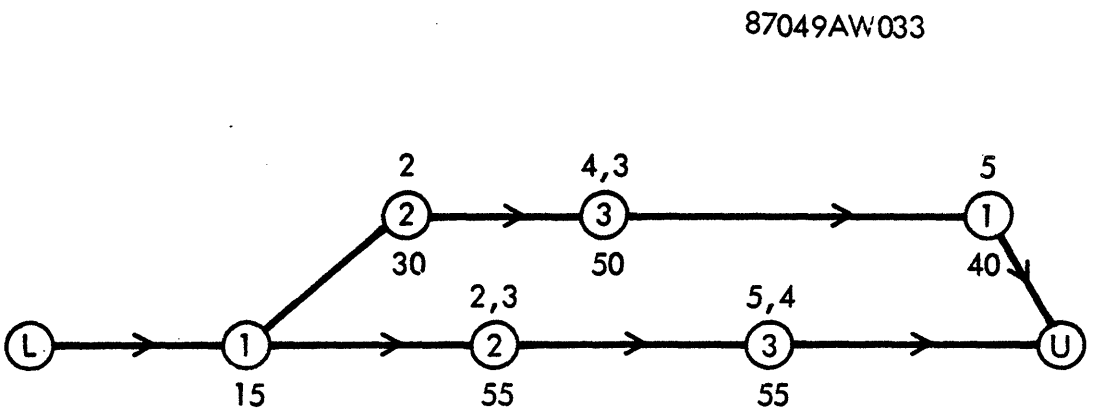


Fig. 2.2. Graphical Representation of Strategies

system controller monitors these variables and can affect them by varying the loading rate and allocating pieces entering the system to the strategies available.

The total arrival rate λ_j at workstation j is

$$\lambda_j = \sum_{i=1}^{P} \sum_{k} x_{ij}^{k} \tag{2.15}$$

The variables x_{ij}^{k} are related by conservation of flow equations and the production ratio requirement. Conservation of flow states that the flow rate of pieces undergoing any operation k is equal to the production rate of that type of piece. This is stated as

$$\sum_{j=1}^{M} x_{ij}^{k} = \sum_{j=1}^{M} x_{ij}^{k-1} \ldots = \sum_{j=1}^{M} x_{ij}^{1} = R_i \qquad i=1,\ldots,P \tag{2.16}$$

where R_i is the production rate of type i pieces. The total production rate is given by

$$R = \sum_{i=1}^{P} R_i = \sum_{i=1}^{P} \sum_{j=1}^{M} x_{ij}^{1} \tag{2.17}$$

The summation is carried out with k=1 for convenience. The production ratio requirement states that pieces of type i comprise a fraction α_i of the total production. This can be expressed as a relationship between R in equation (2.16) and the flow rate of pieces going for operation number 1

$$\sum_{j=1}^{M} x_{ij}^{1} = \alpha_i R = \alpha_i \sum_{i=1}^{P} \sum_{j=1}^{M} x_{ij}^{1} \tag{2.18}$$

where the α_i satisfy

$$0 \leq \alpha_i \leq 1 \qquad i=1,\ldots,P \tag{2.19}$$

$$\sum_{i=1}^{P} \alpha_i = 1 \tag{2.20}$$

An important performance measure of a workstation is the utilization u_j, defined as the probability that a workstation is occupied. Suppose that in an interval of time (0,T) the number of type i pieces passing through workstation j for the operation k is n_{ij}^k. The total time that the station is occupied is thus

$$\sum_{i=1}^{P} \sum_{k} n_{ij}^k \, t_{ij}^k \tag{2.21}$$

The utilization can then be written as

$$u_j = \frac{1}{T} \sum_{i=1}^{P} \sum_{k} n_{ij}^k \, t_{ij}^k \tag{2.22}$$

But it can be recognized that n_{ij}^k/T is the average flow rate x_{ij}^k so that

$$u_j(x) = \sum_{i} \sum_{k} x_{ij}^k \, t_{ij}^k \tag{2.23}$$

The methods of network-of-queues analysis can now be applied so as to express other system performance measures as functions of $x=x_{ij}^k$. Optimization problems can then be formulated so as to pick the assignments x_{ij}^k which maximize the production rate or perhaps some other index of performance.

The total number of customers inside the system either receiving service or waiting in queues is important. Let $q_j(x)$ be the average queue length at workstation j. The in-process inventory can then be defined as

$$I = \sum_{j=1}^{M} q_j(x) \tag{2.24}$$

The calculation of $q_j(x)$ depends on specific assumptions about the service processes at the workstations. If a manufacturing system has exponentially distributed service times and the arrival of pieces into the network constitutes a Poisson process, the result of Jackson (1963) discussed in Section 2.2.1. can be invoked. The workstations can be studied in isolation as M/M/1 queues. In this case, the average length is (Kleinrock, 1975)

$$q_j(x) = \frac{u_j(x)}{1-u_j(x)} \tag{2.25}$$

Similarly if the conditions of Baskett et al., (1975) hold, the relationships for M/M/s or M/G/∞ queues in equilibrium can be substituted in (2.25). For general networks approximate methods can be used to evaluate $q_j(x)$.

The presence of pallets in a system causes added complication. From the point of view of the pallets, the system is closed since there are a finite number of pallets circulating in the system. The methods of analyzing closed queueing systems can then be applied. Secco-Suardo (1978) expresses the probability distribution function (2.4) for a closed network with N customers as a function of the strategy assignments y_i. He suggests that it is possible to use a non linear programming method to maximize the throughput of the loading station and thereby attain the maximum production rate. The results of Denning and Buzen (1977) suggest that this method might be applicable to a wider class of systems than that with exponential servers.

The optimization problem is one of assigning operations to workstations so as to maximize some performance index. The assignments will be subject to constraints imposed by the problem structure.

In a stochastic system, the two important indicies are the production rate R, which should be maximized, and the in-process inventory I, which should be kept at a minimum. A natural objective function in this case

would be a weighted sum of the two. The weights would reflect the return in maintaining a certain production rate as compared to the cost incurred in keeping a certain level of in-process inventory. Thus the following non-linear programming problem results:

NLP 2.1

$$\text{Maximize} \quad \beta_1 \sum_{j=1}^{M} x_{ij}^{1} - \beta_2 \sum_{j=1}^{M} q_j(x) \tag{2.26}$$

$$\text{subject to} \quad \sum_j x_{ij}^{k+1} - \sum_j x_{ij}^{k} = 0 \qquad i=1,\ldots,P,\ k \geq 1 \tag{2.27}$$

$$\sum_{j=1}^{M} x_{ij}^{1} - \alpha_i \sum_{i=1}^{P} \sum_{j=1}^{M} x_{ij}^{1} = 0 \quad i=1,\ldots,P \tag{2.28}$$

$$u_j = \sum_{i=1}^{P} \sum_k x_{ij}^{k} t_{ij}^{k} \leq 1 \quad j=1,\ldots,M \tag{2.29}$$

$$x_{ij}^{k} \geq 0 \qquad \forall i,j,k \tag{2.30}$$

In the objective function (2.26) the production rate of pieces of type 1 is maximized. The ratio constraint (2.28) makes it unnecessary to include the production rate of the other types of pieces. The constraint (2.28) due to the production ratio requirement can be written in a form which is easier to evaluate since it does not involve summing over all the types of pieces.

$$\sum_{j=1}^{M} x_{ij}^{1} - \frac{\alpha_i}{\alpha_1} \sum_{j=1}^{M} x_{1j}^{1} = 0 \tag{2.31}$$

Equation (2.27) is the flow conservation constraint. The limited capacity of the workstations results in (2.29) which states that the utilization of any workstation can not exceed unity if a steady-state equilibrium is to be reached.

It should be noted that it is not necessary to identify strategies in order to formulate NLP2.1. This point will be discussed in Section 2.3 and further elaborated on in Chapter 3.

The problem NLP2.1 can be modified. There might be cases where the average in-process inventory is required to remain below a certain level Q. This can then be expressed as a constraint to give NLP2.2

$$\text{NLP2.2} \qquad \text{Maximize} \quad \sum_{j=1}^{M} x_{ij}^{1} \tag{2.32}$$

subject to (2.27), (2.29), (2.30), (2.31)

$$\text{and} \quad \sum_{j=1}^{M} q_j(x) \leq Q \tag{2.33}$$

Where queue lengths grow without bound as utilization approaches unity, constraint (2.33) may make (2.29) redundant.

Enumerating Strategies in Advance

There are instances where it is either necessary to enumerate strategies in advance or the number of possible strategies is not large and they can be readily identified. For example, if the four workstations in the example given above are arranged linearly, as in Fig. 2.3, there are then only two possible strategies. They are depicted in Fig. 2.4. The number of possible strategies S_i for a given piece normally depends on the nature and number of the operations and not just the geographic layout of the workstations.

Let y_ℓ be the flow rate into the network of pieces following strategy ℓ. The production rate is the total flow rate into the network

$$R = \sum_{\ell=1}^{S} y_\ell \tag{2.34}$$

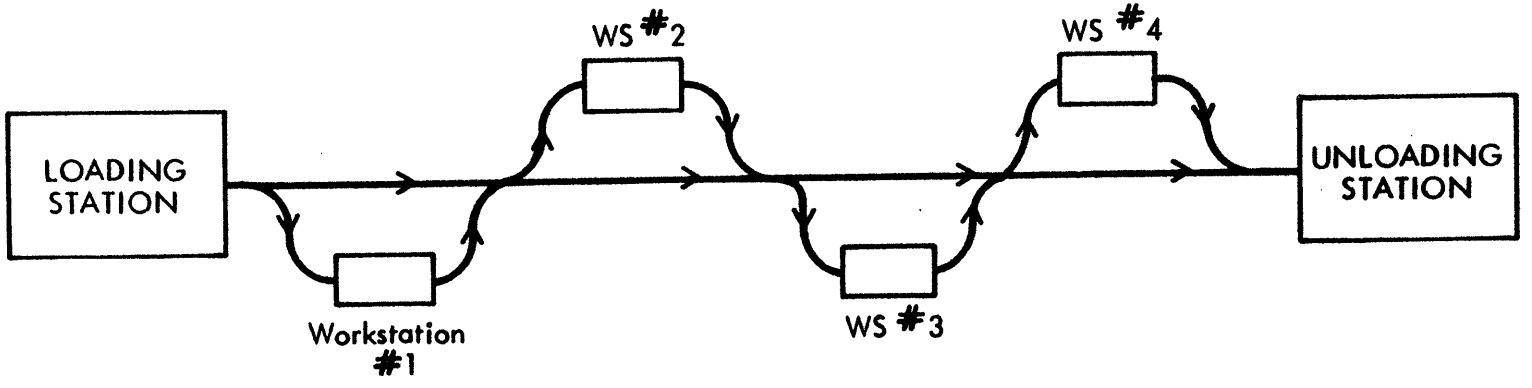


Fig. 2.3. Linear Arrangement of Workstation (Flexible Transfer Line)

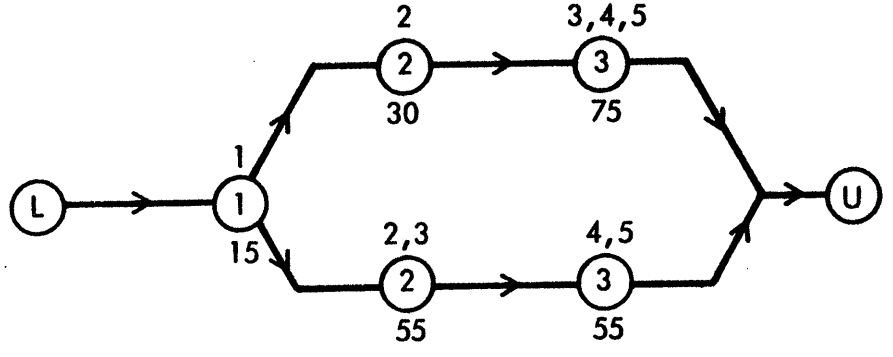


Fig. 2.4. The Two Possible Strategies

where

$S = \sum_{i=1}^{P} S_i$, and S_i is the number of strategies available for a piece of type i.

The arrival rate λ_j at workstation j is

$$\lambda_j = \sum_{\ell \in m(j)} y_\ell \tag{2.35}$$

where m(j) is the set of strategies that use workstation j. The utilization is given by

$$u_j = \sum_{\ell \in m(j)} y_\ell \tau_{\ell j} \tag{2.36}$$

These quantities can be used in NLP 2.3 and NLP 2.4 to find the optimal mixture of strategies in the system. The two programs NLP 2.3 and NLP 2.4 are analogous to NLP 2.1 and NLP 2.2 respectively. The relationship between y_ℓ and x_{ij}^k is given by equation (2.43) and (2.90).

NLP 2.3

$$\text{Maximize} \quad \beta_1 \sum_{\ell=1}^{S} y_\ell - \beta_2 \sum_{j=1}^{M} q_y^j(y) \tag{2.37}$$

subject to

$$\sum_{\ell \in m(j)} y_\ell \tau_{\ell j} \leq 1 \tag{2.38}$$

$$\sum_{\ell \in S(i)} y_\ell - \alpha_i \sum_{n=1}^{S} y_n = 0 \tag{2.39}$$

$$y_\ell \geq 0 \tag{2.40}$$

NLP 2.4

$$\text{Maximize} \quad \sum_{\ell=1}^{S} y_\ell \tag{2.41}$$

subject to (2.38), (2.39), (2.40)

and

$$\sum_{j=1}^{M} q_y^j(y) \leq 0 \tag{2.42}$$

Constraint (2.39) expresses the production ratio requirement. In calculating the average queue length $q_y^j(y)$ at the workstations, use is made of (2.35) which expresses the arrival rate at a workstation as a function of strategy assignments y .

Modelling of the Transportation System

The transportation system can be modelled as a network of arcs and nodes. The nodes are either merges or diverges of arcs, or the actual workstations themselves. It is natural to view most transportation systems as transportation networks (Magnanti, 1977). Hence network models are applied to a wide class of transportation systems. In flexible manufacturing systems, network models can be used to model conveyer belts or systems where pieces are carried on a vehicle moving along a guideway.

For convenience, it is assumed that the nodes are numbered so that the first M are workstations and the remainder merges or diverges. Furthermore it is assumed that the arcs are numbered so that arc i leads to workstation i, with the rest of the arcs being numbered M+1, M+2, and so on. The arc leading into the loading station is labelled 0. This allows congestion effects at the loading station to be modelled. The network of Fig. 2.3, for example, has the labels shown in Fig. 2.4. The circled numbers represent nodes while the rest are arc numbers. Define r_{ij} as the flow rate of type i pieces on arc j of the network. From the definitions,

$$r_{ij} = \sum_k x_{ij}^k \qquad j = 1,\ldots,M \tag{2.43}$$

The problem NLP 2.1 can be modified to become

NLP 2.1a

$$\text{Maximize } \beta_1 \sum_{i=1}^{P} r_{i0} - \beta_2 \sum_{j=1}^{M} q_j(x) + g(r) \tag{2.44}$$

subject to (2.27), (2.29), (2.30), (2.31), (2.43)

and $$\sum_{j \in A(n)} r_{ij} - \sum_{j \in D(n)} r_{ij} = 0 \qquad \forall\, n \tag{2.45}$$

$$\sum_{i} r_{ij} \leq d_j \tag{2.46}$$

$$r_{ij} \geq 0 \tag{2.47}$$

where the r_{i0} is the flowrate of type i pieces into the network. The constraint (2.45) expresses flow conservation at network nodes in which $A(n)$ is the set of arcs leading to node n and D(n) is the set of arcs carrying pieces away from the node. Arc capacity constraints if present are expressed by (2.46)

The in-process inventory consists of pieces queueing at the workstations $q_i(x)$ and those in transit in the network g(r). The derivation of g(r) is dealt with below. The total in-process inventory is thus on average

$$I = \sum_{j=1}^{M} q_j(x) + g(r) \tag{2.48}$$

This is incorporated in the cost function (2.44). Similarly NLP 2.2 can be written as

NLP 2.2a

$$\text{Maximize } \sum_{i} r_{i0} \tag{2.49}$$

subject to (2.27), (2.29), (2.30), (2.31), (2.43), (2.45), (2.46)

and

$$\sum_{i=1}^{M} q_j(x) + g(r) \leq Q$$

$$x_{ij}^{k}, r_{ij} \geq 0 \tag{2.50}$$

The number of pieces on any arc ℓ in the network is on the average given by

$$I_\ell = f_\ell \; t_\ell \tag{2.51}$$

where f_ℓ is the total flow rate on the arc and t_ℓ is the average travel time on the arc. This is an example of Little's formula (Kleinrock, 1975). If the arcs are subject to congestion effects, the travel time is then an increasing function of the total flowrate f_ℓ. The total flow rate is given by

$$f_\ell = \sum_{i=1}^{P} r_{i\ell} \tag{2.52}$$

$$\text{Then} \quad g(r) = \sum_{\ell > M} I_\ell \tag{2.53}$$

The transportation system can be handled in a similar fashion where the strategies are enumerated in advance.

The transportation network has path constraints characterized by networks of possible strategies such as Fig. 2.2. Each arc on the strategy network corresponds to a flow between two workstations. In a densely connected transportation system, there is a choice of paths between the two workstations while a simple system as in Fig. 2.5 provides no choice. Chapter 3 has a further discussion of these constraints and how the network structure may be exploited in order to solve the routing problem.

2.3.2 Modelling of Deterministic Systems

A deterministic flexible manufacturing system is one in which the processing times are entirely deterministic. The arrival process into the

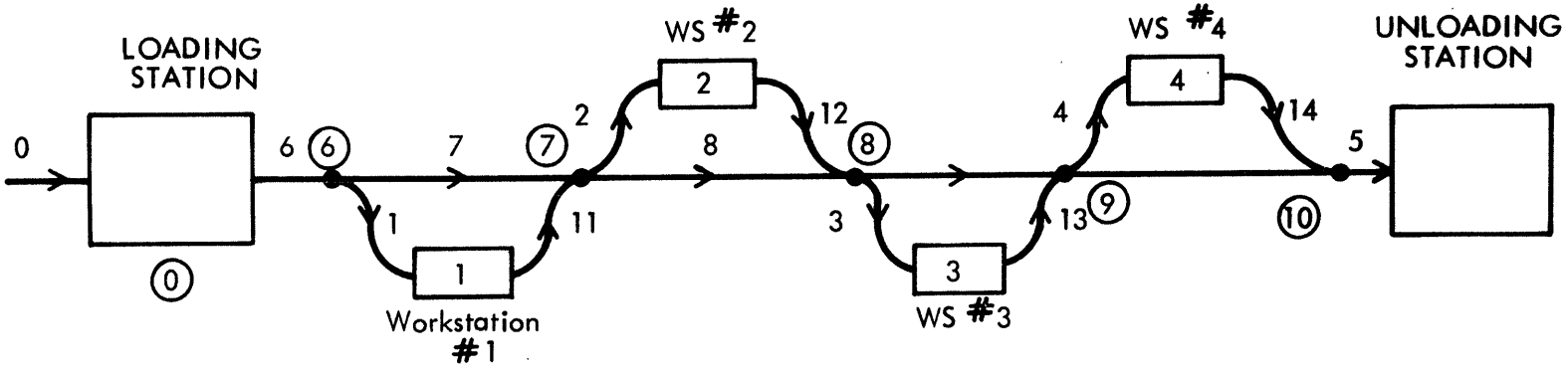


Fig. 2.5. A Flexible Manufacturing System

system is deterministic in the sense that workpieces can be introduced into the system at pre-determined time instants. The assignment problem can be formulated and solved as a job-shop scheduling problem (Fisher, 1970). There are added complications however. An optimal steady state assignment is being sought. This means that the number of jobs to be assigned is not only undetermined, but it is also likely to be large. For a similar reason, the time interval over which the assignments have to be made is undetermined. Each of the jobs to be scheduled has options as to which workstation it can go to for a particular operation. All of these factors increase the size and complexity of the scheduling problem. From a control point of view, precise schedules worked out in advance are difficult to implement, especially over long time intervals.

One way of overcoming these difficulties is to use a periodic schedule (Hitz, 1979). A periodic schedule is one in which a certain sequence of operations at the workstations is repeated at regular time intervals. There is a set of integer numbers n_i, i=1,...,P such that

$$n_i = \alpha_i \sum_{\ell=1}^{P} n_\ell \qquad (2.54)$$

if n_i is the number of type i pieces to be manufactured in a period. A schedule is sought in which there is no idle time on the bottleneck workstation. The bottleneck is the station j that maximizes $\sum_i \xi_{ij} n_i$, in which ξ_{ij} is the total time that pieces of type i spend at workstation j during their manufacturing process. The schedule should be such that it can be repeated without leaving any idle time on the bottleneck workstations. In order to derive ξ_{ij}, the strategies used to manufacture each of the pieces should be determined; then $\xi_{ij} = \sum_{\ell \in S(i)} \tau_{\ell j}$.

The aggregated flow approach used in stochastic models affords a way of simplifying the assignment problem. It will now be extended to deterministic systems.

Consider a time interval (0,T). Let n_{ij}^k be the number of type i pieces

that are sent to workstation j for operation k in the interval (0,T). The assignments n_{ij}^{k} are to be made in a manner which maximizes the total production while maintaining the ratio requirement (2.54). The following integer program can thus be solved in order to achieve this objective:

IP 2.1

$$\text{Maximize} \quad \sum_{j=1}^{M} \sum_{i=1}^{P} n_{ij}^{1} \tag{2.55}$$

subject to (2.54)

$$\text{and} \quad \sum_{j=1}^{M} n_{ij}^{k} = \sum_{j=1}^{M} n_{ij}^{k-1} \qquad k=2,... \tag{2.56}$$

$$\sum_{i=1}^{P} \sum_{k} n_{ij}^{k} \; t_{ij}^{k} \leq T \tag{2.57}$$

$$n_{ij}^{k} \geq 0 \tag{2.58}$$

$$n_{ij}^{k} \quad \text{integer}$$

The objective function (2.55) is the total production. Constraint (2.56) requires that all operations are carried out on all the pieces. Expression (2.57) reflects the fact that all manufacturing processes must be completed in the inverval (0,T). The solution of IP 2.1 could serve as a basis for the periodic scheduling algorithm. The problem would be in determining T, which would then be the period of the schedule.

The flow rates in the interval (0,T) can be defined as

$$x_{ij}^{k} = \frac{n_{ij}^{k}}{T} \tag{2.59}$$

With this transformation, consider the following linear program derived from IP 2.1

LP 2.1

$$\text{Maximize} \quad \sum_i \sum_j x_{ij}^1 \tag{2.60}$$

$$\text{s.t.} \quad \sum_{j=1}^{M} x_{ij}^1 = \alpha_i \sum_{i=1}^{P} \sum_{j=1}^{M} x_{ij}^1 \tag{2.61}$$

$$\sum_{j=1}^{M} x_{ij}^{k+1} = \sum x_{ij}^k \tag{2.62}$$

$$\sum_i \sum_k x_{ij}^k \; t_{ij}^k \leq 1 \tag{2.63}$$

$$x_{ij}^k \geq 0 \tag{2.64}$$

The relationship between LP 2.1 and NLP 2.1 or NLP 2.2 is obvious. The constraints (2.61), (2.62), and (2.63) are identical to (2.28), (2.27), and (2.29) of NLP 2.1 or NLP 2.2. The deterministic problem does not take into account the buildup of queues within the system. This accounts for the difference between LP 2.1 and the non-linear programs in the stochastic case.

If $\hat{x}_{ij}^k$ is the optimal solution of LP 2.1 then $\hat{n}_{ij}^k = T\hat{x}_{ij}^k$ is optimal in IP 2.1 if $T\hat{x}_{ij}^k$ is integer. Otherwise it provides an upper bound on the optimal value of the production rate. The time horizon over which the optimal assignment is carried out is long compared to the operation times t_{ij}^k. Thus the numbers $\hat{n}_{ij}^k$ are large. The difference between the optimal solution of IP 2.1 and $\hat{n}_{ij}^k$ are thus negligible (Salkin, 1975).

Secco-Suardo (1978) derives a similar linear program for maximizing the throughput of a network modelled as a closed network of queues. In the limit as the number of customers inside the network grows large, it is found that the throughput is proportional to the ratio of the relative utilization of the bottleneck workstation to that of the loading station. The problem is then one of finding the $\max \min_j \bar{u}_j(x)$. This is formulated

as a linear programming problem similar to LP 2.1.

Baskett et al., (1975) show that the marginal probability distribution of having n customers at a queue depends only on the mean service and arrival rate. This indicates that the asymptotic result holds for general networks satisfying Baskett et al.'s (1975) assumptions. Furthermore, as a variance of the service time distribution goes to zero, the linear program described is unchanged as long as the assumptions - including that service time distributions have rational Laplace transforms - remain satisfied. The deterministic case can thus be viewed as a limit of the class of systems to which the stochastic network of queues theory applies.

The work rate theorems of Chang and Lavenberg (1972) show that the throughput of a closed network is proportional to the ratio of the relative utilization of the bottleneck station and the loading station. They make no assumptions regarding the queue discipline. The only restriction on service time distributions is that they should have finite non-zero expectations.

The linear programs of this chapter yield the maximum asymptotic throughput solution for networks with general service time distributions. It should be noted however that they do not take into account the build up of queues within the network.

2.4 An Approximate Method for Finding the Production Rate of a Balanced Closed System

The linear programming formulation of Section 2.3.2 finds the limiting maximum production rate in a closed system as the number N of pieces in the system becomes large. The effect of a limited number of pallets is important. Simulation results show that the production rate of the system increases asymptotically to a maximum value as the number of pallets inside the system increases (Horev et al., 1978). Closed queueing network models also exhibit this rise in throughput as the number of pieces inside the network grows (Ward, 1980) (Secco-Suardo, 1978).

An estimate of the production rate as a function of the number of pallets can be derived. In a system where there are only single server stations with exponentially distributed service times, the probability

that there are n_i pieces at station i is given by (Gordon and Newell, 1967)

$$P(n_1, n_2 \ldots n_M) = \frac{1}{G(M,N)} \prod_{i=1}^{M} X_i^{n_i} \tag{2.65}$$

with

M = number of stations

N = number of pallets

X_i = relative utilization of station i

and

$$\sum_{i=1}^{M} n_i = N \tag{2.66}$$

The relative utilization is defined as

$$X_i = \frac{e_i}{\mu_i} \tag{2.67}$$

where μ_i is the service rate of station i, and e_i are constants satisfying $e_i = \sum_{j=1}^{M} p_{ji} e_j$ with p_{ji} being the probability that a workpiece goes to station i immediately upon completion of service at station j.

The factor G(M,N) is a normalizing constant,

$$G(M,N) = \sum_{S} \prod_{i=1}^{M} X_i^{n_i} \tag{2.68}$$

where S is the set of all partitions of N pieces at M stations.

Assume that the system is balanced in the sense that all stations have the same relative utilization. Then,

$$X_i = \frac{e_i}{\mu_i} = X \quad \forall i \tag{2.69}$$

Substituting (2.69) into (2.68) gives

$$G(M,N) = \sum_{S} \prod^{M} x^{n_i} = \sum_{S} x^{N} \tag{2.70}$$

The number of partitions in S is $\binom{N+M-1}{M-1}$

The normalizing constant is then

$$G(M,N) = \binom{N+M-1}{M-1} x^{N} = \frac{(N+M-1)!}{(M-1)!N!} x^{N} \tag{2.71}$$

The throughput T_i of station i is given by (Secco-Suardo, 1978)

$$T_i = e_i \frac{G(M,N-1)}{G(M,N)} \tag{2.72}$$

From (2.71) and (2.69) this can be written as

$$T_i = \frac{N}{N+M-1} \mu_i \tag{2.73}$$

The assumption behind (2.73) is that the utilizations of all stations in a balanced system are equal for all values of N. If R is the limiting maximum production rate of the balanced system, the actual production rate if the number of pallets is limited is

$$P = \frac{N}{N+M-1} R \tag{2.74}$$

This is established by noting that P is the throughput of the loading station. The relationship (2.74) then follows naturally from (2.73).

The approximation can be extended. If in (2.65) there are (M-1) balanced stations and station M (for convenience) has a relative utilization $q(0<q<1)$ times that of balanced stations, (2.70) can be written as

$$G(M,N) = \sum_{S} \prod_{i=1}^{M-1} (X^{n_i}) \, (qX)^{n_M} \tag{2.75}$$

This becomes

$$G(M,N) = X^N \sum_{S} q^{n_M} \tag{2.76}$$

This model corresponds to the practice of modelling the loading station and transportation system as a server in a closed network of queues model. Its utilization is usually lower than that of the servers corresponding to the workstations.

Equation (2.76) is a polynomial in q with $n_M = 0,\ldots N$. The coefficient of q^{n_M} is X^N times the number of partitions of $(N-n_M)$ pieces in (M-1) machines and is given by $\binom{N+M-2-n_M}{M-2}$. Thus

$$G(M,N) = X^N \sum_{i=0}^{M} \frac{(N+(M-2)-i)!}{(N-i)!\,(M-2)!} \tag{2.77}$$

The expression G(M,N-1)/G(M,N) is thus a ratio of two polynomials

$$\frac{G(M,N-1)}{G(M,N)} = \frac{\sum_{i=0}^{N-1} b_i q^i}{X \sum_{j}^{N} a_j q^j} \tag{2.78}$$

where

$$b_i = \frac{(N-1+(M-2)-i)!}{(N-1-i)!\,(M-2)!} \tag{2.79}$$

$$a_i = \frac{(N+(M-2)-i)!}{(N-i)!\,(M-2)!} \tag{2.80}$$

From (2.79) and (2.80) it can be seen that $b_{i+1} = a_i$, and

$$\frac{a_{i+1}}{a_i} = \frac{N-i}{N+M-2-i} \tag{2.81}$$

Thus if $M > 2$, then $a_{i+1} < a_i$.

The probability that there are $n_M = j$ pieces at the nonbottleneck station is,from (2.65) and (2.75)

$$P_M(j) = \frac{X^N}{G(M,N)} a_j q^j \tag{2.82}$$

Since q is less than 1 and $a_{i+1} < a_i$ for $M > 2$, then $P_M(j+1) < P_M(j)$. If q is sufficiently small, or in other words if the nonbottleneck station is much faster than the other balanced stations, it is reasonable to approximate $G(M,N-1)/G(M,N)$ by considering only the coefficients a_o and a_1 of the polynomials. This gives an approximation with the form

$$\frac{G(M,N-1)}{G(M,N)} \simeq \frac{1}{X} \frac{1}{A+Bq} \tag{2.83}$$

where A and B are constants. By equating coefficients for the first two terms in (2.78) and applying (2.80),

$$A = \frac{N+M-2}{N} \tag{2.84}$$

$$B = \frac{M-3}{N(N+M-3)} \tag{2.85}$$

Substituting into (2.83) gives the expression

$$\frac{G(M,N-1)}{G(M,N)} \simeq \frac{1}{X} \frac{N(N+M-3)}{(N+M-2)(N+M-3) + (M-3)q} \tag{2.86}$$

Note that for q = 0,

$$\frac{G(M,N-1)}{G(M,N)} = \frac{1}{X} \frac{N}{N+M-2} \tag{2.87}$$

This is equivalent to (2.74) but with one station less. In practice, the expression (2.86) is found to vary only slightly with q (Fig. 2.6) and it is better to use the simple expression (2.87) since it involves less computation.

To generalize, if the number of bottleneck stations, M_B, with equally high relative utilization is larger than two, an approximation to the production rate P as a function of the number of pallets and the limiting maximum production rate, R, is

$$P = \frac{N}{N+M_B-1} R \tag{2.88}$$

The equation (2.88) is useful because for a balanced system it is possible to obtain a good assessment of the production rate and the number of pallets required by solving a linear program and a simple equation. It is also possible to estimate the return from an investment on pallets. In a system with four balanced workstations, it takes approximately 27 pallets to have a production rate 90% (i.e., $\frac{XG(M,N-1)}{G(M,N)} = 0.9$) of the limiting rate. To raise that to 95% requires an additional 30 pallets. This assumes stochastic effects which are implicit in the closed queueing network model.

The rate at which the ratio $XG(M,N-1)/G(M,N)$ approaches unity as $N \to \infty$ is important. It determines the number of pallets that are needed in a system in order to have a production rate close to the asymptotic maximum. The rate depends on the number of bottleneck stations in the system. The more balanced the system is, the slower the convergence. Intuitively this may be explained by the fact that in a balanced system, the pallets distribute themselves evenly at the machines. The asymptotic production rate is achieved when there is no idle time at the bottleneck stations.

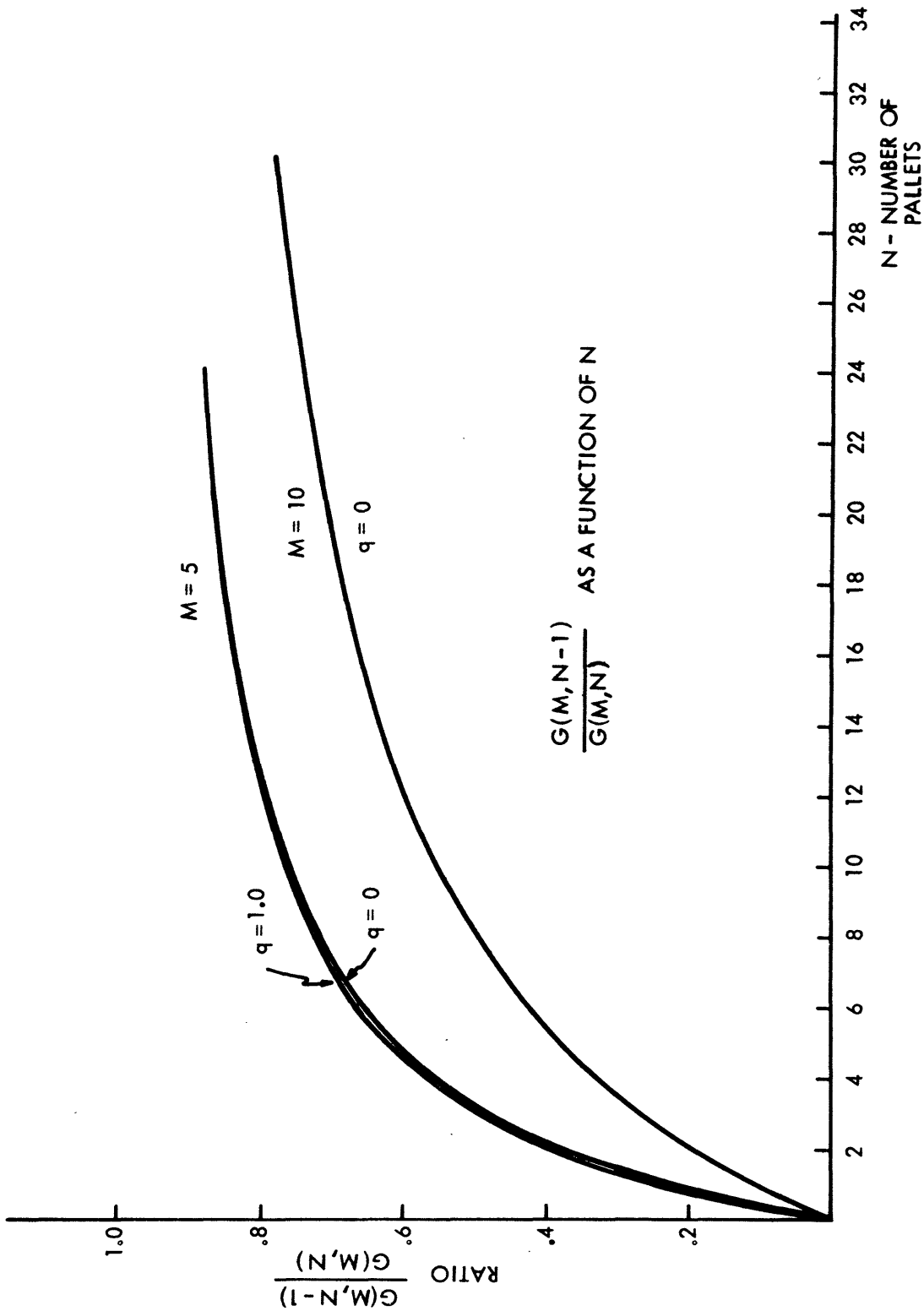


Fig. 2.6. Approximate Expression for $G(M,N-1)/G(M,N)$ As a Function of N.

Thus the more balanced a system is, the larger the number of pallets needed to do this. In Fig. 2.6, the ratio (2.87) is plotted for two different values of M and it can be seen that for the higher value of M, it approaches the asymptote much more slowly than for the lower value.

It may happen that the highest asymptotic production rate is achieved by a strategy assignment which produces a balanced system. It is also possible that for finite N, a different assignment leading to an unblanced system has a higher production rate. This is because the unbalanced system approaches its limiting production rate faster than the balanced system as N increases.

In Fig. 4.19, the throughput as a function of N for a four-workstation system modelled as a closed network of queues is shown. For N≃13, the production rate for the unbalanced system is the same as that of the balanced system. The balanced system needs about 30 pallets to reach 90% of its asymptotic throughput whereas the unbalanced system needs only 15. As N increases, the balanced system has a higher throughput.

In choosing the optimal mix of strategies, the number of pallets, if small, should be considered. The optimization model suggested by Secco-Suardo (1978) is a method of tackling such problems.

2.5 Some Characteristics of the Solutions of the Optimization Problems

Let $\hat{x}_{ij}^{k}$ denote an optimal assignment of operations in a flexible manufacturing system. The constraints due to the workstation capacity limitation are, at the optimal point

$$\hat{u}_j = \sum_i \sum_k \hat{x}_{ij}^{k} t_{ij}^{k} \leq 1 \qquad (2.89)$$

In LP 2.1 some will be satisfied as equality constraints and the others as inequalities. The workstations corresponding to constraints satisfied as equalities are the bottleneck stations. The production rate of the system can only be increased for a particular parts requirement by increasing the speed of the bottleneck workstation (i.e., decreasing t_{ij}^{k} for bottleneck stations).

In NLP 2.1 and NLP 2.2, it is likely than none of the constraints are satisfied at the boundary. The values of $\hat{u}_j$ then give a measure of the relative workloads at the stations.

A common industrial practice when assigning manufacturing strategies to workpieces is to do it in such a way that the workloads at the workstations are balanced (Hutchinson, 1977). That is,the values of $\hat{u}_j$ are as uniform as possible. While this may be optimal for a system designed for a certain specification of parts, in general it will not be optimal for an arbitrary mix of workpieces. The lifetime of a flexible manufacturing system is almost certainly longer than the production run of any specification of parts being produced. It seems unlikely therefore that a given production requirement utilizes all workstations evenly.

Given the parts specifications, machining and production ratio requirements for a flexible manufacturing system, the optimization procedure produces an optimal assignment of strategies and the utilizations of all the workstations. A shrewd production manager may then be able to undertake the manufacture of additional parts which are not in the original order. From the solution of the optimization, he can see how much idle time there is at each of the workstations. He then matches this idle time to the production requirements of any additional items which might be in short supply or are needed to maintain inventory levels. This is clearly an improvement. The productivity of the system is thereby increased and the workstations have a more balanced workload.

The variables $\hat{x}_{ij}^k$ and $\hat{r}_{ij}$ assign flow rates through the workstations and on the arcs of the network. They do not,however,define a unique routing for the pieces through the network. The strategy flow rates y_ℓ do define a unique routing because each strategy defines a path through the network from the loading to the unloading station. The relationship between y_ℓ, x_{ij}^k and r_{ij} is given by (2.43) and

$$\sum_i r_{ij} = \sum_{\ell \epsilon p(j)} y_\ell \qquad (2.90)$$

with p(j) being the set of strategies which use arc j. Thus given a solution $\hat{r}_{ij}$ and $\hat{x}_{ij}^k$ to the optimization problem, it is not possible to choose a unique set of strategies y .

The real-time system controller has to maintain the optimal flow rates $\hat{x}_{ij}^k$ and $\hat{r}_{ij}$. Equations (2.90) and (2.43) mean that the controller has some freedom in choosing the actual path followed by an individual piece through the system. It may be possible therefore for a local controller at a workstation to make a decision as to where to send a workpiece next, acting on information received from a central controller or from the other workstations themselves. Alternatively by deciding on the strategy a workpiece is to pursue at the loading station, the only task remaining for the local controller is that of keeping a workpiece on its proper path. Switching a workpiece from one strategy to another during its passage through the system is another possible control action.

Chapter 3 discusses optimization methods which generate the strategy flows y_ℓ. This involves, in addition to $\hat{r}_{ij}$ and $\hat{x}_{ij}^k$, storing additional information about the optimal solution which describes the sequence of visits to the workstations and the path followed on the transportation system.

3. OPTIMIZATION TECHNIQUES FOR FLEXIBLE MANUFACTURING SYSTEMS

3.1 Introduction

In order to implement efficient algorithms to solve the mathematical programming problems of Chapter 2, the structure of the problems must be exploited. In addition to finding the optimal value of the objective function, the routing of the workpieces and ordering of operations must be resolved. This may in principle be done by enumerating all possible strategies in advance but the problem structure is such that enough information is generated during the solution so that only a subset of the strategies need be considered in finding the optimal solution.

Section 3.2 covers linear programming problems. These are important not only in their own right but also because they form the strategy generating step in the solution of all the optimization problems. The decomposition principle of Dantzig and Wolfe (1963) is applied and the column-generating sub-problems are shown to produce a strategy for each type of workpiece.

The non-linear programming problems NLP 2.1 and NLP 2.2 are treated in Section 3.3. A modified form of the Cantor-Gerla extremal flow algorithm is used to solve NLP 2.1. The augmented Lagrangian method is used to attach the non-linear constraint in NLP 2.2 to the objective function. The extremal flow algorithm is then used to iteratively solve a sequence of linearly constrained problems. In each case, the decomposition method is used to solve the flow generating sub-problems.

3.2 Linear Programming and Flow Optimization in Flexible Manufacturing Systems

Linear programming is the most widely used form of optimization. There have been many recent advances in algorithms for specialized application in areas such as multi-commodity network problems. The flexible manufacturing system has a structure which is suited to some of these algorithms. The importance of linear programming is further enhanced by the fact that it is used in non-linear optimization problems as part of the solution procedure.

Assume that the elements x_{ij}^{k} of the flow vector defined in Section 2

are ordered in such a way that x, the flow vector, can be partitioned into a set of vectors x_i which describe the flow due only to pieces of type i. The linear program LP2.1 can be written as

LP 3.1 Maximize R (3.1)

subject to

$$\sum_{i=1}^{P} T_i x_i \leq 1 \tag{3.2}$$

$$c_i x_i = \alpha_i R \qquad i=1,\ldots P \tag{3.3}$$

$$A_i x_i = 0 \qquad i=1,\ldots P \tag{3.4}$$

$$x_i \geq 0 \tag{3.5}$$

in which R is the scalar production rate. The vector c_i has elements 0 and 1 such that

$$c_i x_i \equiv \sum_{j=1}^{M} x_{ij}^{1} \tag{3.6}$$

Thus $c_i x_i$ is the production rate R_i of type i_p pieces. The production ratio constraints are represented by (3.3). Since $\sum_{i=1} \alpha_i = 1$, (3.3) and (3.6) imply that the production rate R can be expressed as

$$R = \sum_{i=1}^{P} c_i x_i = \sum_{i=1}^{P} \sum_{j=1}^{M} x_{ij}^{1} \tag{3.7}$$

The matrix T_i comprises the elements t_{ij}^{k}, which are the operation times for type i pieces. The flow conservation constraints are defined by (3.4), the matrix A_i being composed of elements which are -1, 0, or 1.

The constraints in LP 3.1 consist of a set of decoupled constraints (3.4) and coupling constraints (3.2) and (3.3). Decomposition methods can therefore be applied to take advantage of the special structure of the flow conservation constraints.

The flow conservation constraints (3.4), define a convex cone Ω_i (Bazaraa, 1976). Let x_i^s (s=1,...) be a set of solutions to $A_i x_i = 0$. The vectors x_i^s (i=1,...P) are in the cone and define x^s to be

$$x^s = \sum_{i=1}^{P} x_i^s \tag{3.8}$$

The vector x^s consists of flow rates x_{ij}^k which satisfy the flow conservation constraints. By scaling x_i^s so that

$$c_i x_i^s = \alpha_i \tag{3.9}$$

the ratio requirement constraints are also satisfied. The linear program LP 3.1 can now be stated as

LP 3.2

$$\text{Maximize} \quad \sum_s q_s \tag{3.10}$$

Such that

$$\sum_s q_s \sum_{i=1}^{P} T_i x_i^s \leq 1 \tag{3.11}$$

$$q_s \geq 0 \qquad \forall\ s \tag{3.12}$$

The problem may be interpreted as one of choosing the optimum weighting q_s on the flows x^s. The objective function (3.10) is of this form because each of the x^s is normalized to represent a unit production rate. Thus the production rate R due to the weighted combination of flows x^s is

$$R = \sum_s \sum_{i=1}^{P} q_s c_i x_i^s \tag{3.13}$$

Using (3.9) and (2.18) this can be seen to be

$$R = \sum_s \sum_{i=1}^{P} q_s \alpha_i = \sum_s q_s \tag{3.14}$$

This method is a direct application of the price-directive decomposition method of Dantzig and Wolfe (1963). The constraint set in the decoupled subsystems are convex cones. The vectors x^s are columns in LP 3.2 and they are generated as needed by using a column generating method (Dantzig, 1963). If $\pi \varepsilon R^M$ are the dual variables associated with the constraint set (3.11), the x_i^s can be obtained by solving P sub-problems (Lasdon, 1970):

LP 3.3

$$\text{Minimize } (\pi T_i)\, x_i \tag{3.15}$$

$$\text{such that } \quad A_i x_i = 0 \tag{3.16}$$

$$c_i x_i = \alpha_i \tag{3.17}$$

$$x_i \geq 0$$

The solutions are the vectors x_i^s which are used to define the columns x^s. The optimal solution is reached in the master problem LP 3.2 when the dual variables π are all non-negative, indicating that the production rate can not be increased further by introducing a new column into the basis.

Solving the sub-problems corresponds to the operation of "pricing out" the columns of a linear program in the simplex method (Dantzig, 1963). In this case, however, the number of columns in LP 3.2 is not only large, but the columns are not known in advance. The sub-problems, in effect, find the column with minimum reduced cost which is to enter the basis.

The sub-problems LP 3.3 are easy to solve. Using the vector $\{\pi T_i\}$, the master problem LP 3.2 allocates a cost equal to $\pi_j t_{ij}^k$ for each piece of type i at workstation j for operation k. The sub-problems LP 3.3 then find the sequence of workstation visits with the lowest overall cost and allocates to it a flow equal to α_i for pieces of type i. This is easily accomplished by finding the workstation with the lowest cost for each operation and setting the corresponding variable equal to α_i. That is, for each k find

$$\pi_s t_{is}^k = \min_j \pi_j t_{ij}^k \tag{3.18}$$

and then set $x_{is}^k = \alpha_i$ and $x_{ij}^i = 0$ for $j \neq s$. The solution x_i^s to LP 3.3 is a strategy since it defines a sequence of operations for each type of workpiece. At this stage, if the subscripts k do not denote strict precedence constraints, the ordering of the operations can be resolved and stored. To resolve the ordering one need only send the workpieces along the shortest physical path in the transportation system which visits the required workstations. This is a traveling salesman problem in which only a subset of the nodes have to be visited.

With the sub-problems seen to be generating strategies, the master problem can be interpreted as choosing the optimal combination of strategies so as to maximize the production rate. The overall procedure is summarized in the flow chart of Fig. 3.1. It is assumed that an initial flow x^1 is available. This can be generated by assigning a flow α_i on an arbitrary strategy for each type of piece.

The decomposition approach results in a significant saving in computational effort. The initial linear program LP 3.1 with many constraints is replaced by the master problem LP 3.2 with fewer constraints. The computational effort required to solve a linear program is proportional to m^3, where m is the number of constraints (Bradley, 1977). The sub-problems LP 3.3 resulting from decomposition are easily solved leading to further savings in computational effort.

3.3 Non-Linear Optimization in Flexible Manufacturing Systems

The special structure of the flexible manufacturing system can be exploited in order to implement efficient non-linear programming techniques. The method of attack involves breaking the problem into flow-generating linear programs (which can be solved using the decomposition method of Section 3.2) and non-linear optimization problems with a reduced number of variables and simpler constraints.

The problem NLP 2.1 consists of a nonlinear objective function to be maximized subject to a set of linear constraints. A number of methods exist which exploit the convex structure of the linear constraint set to generate feasible ascent directions (Nguyen, 1974). Algorithms in this category include the Frank-Wolfe (Nguyen, 1974), gradient projection (Luenberger, 1973), and the reduced gradient (Himmelblau, 1972) methods.

In addition to obtaining the optimal solution, the routing of workpieces in the network must be resolved. The Dantzig-Wolfe (1963) decomposition principle applied to the linear program not only gives rise to sub-problems which are easy to solve, but also resolves the routing problem by generating strategies. It would be advantageous to incorporate this property into a method for solving the non-linear optimization problem.

The problem NLP 2.1 is in the form

NLP 3.1

$$\text{maximize } f(x) \tag{3.19}$$

subject to

$$\sum_{i=1}^{P} T_i x_i \leq 1 \tag{3.20}$$

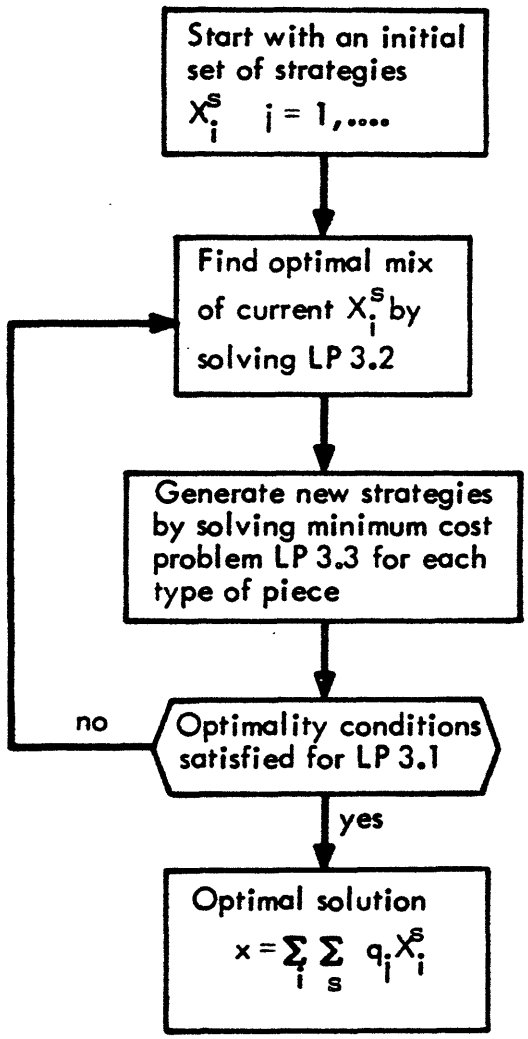


Fig. 3.1. Flow-Generating Decomposition Algorithm

$$c_i x_i = \frac{\alpha_i}{\alpha_1} c_1 x_1 \quad i=1,\ldots P \tag{3.21}$$

$$A_i x_i = 0 \tag{3.22}$$

$$x_i \geq 0 \tag{3.23}$$

The constraints (3.20)-(3.23) are the same as those of LP 3.1 in Section 3.2. The flow conservation constraint (3.21) here is in a slightly different but equivalent form.

The set Ω defined by the linear constraints (3.20)-(3.23) is convex, closed, and bounded. Because of (3.22), every member of Ω can be expressed as a vector sum of the elements of the lower dimension convex cones Ω_i defined in Section 3.1. Any point in the set Ω can be expressed as a convex combination of the extreme points ψ^l of the set(Bazaraa, 1976). That is, any point $x \varepsilon \Omega$ can be expressed in the form

$$x = \sum_l w_l \psi^l \tag{3.24}$$

$$\text{with } \sum_l w_l = 1 \tag{3.25}$$

$$w_l \geq 0 \tag{3.26}$$

Cantor and Gerla (1974) optimize non-linear functions subject to linear network flow conservation constraints by considering such convex combinations. The extreme points are generated by solving linear programs over the linear constraint set. The method can be adapted to solve NLP 2.1. The essential difference lies in the interpretation of the extreme points. In the Cantor-Gerla extremal flow algorithm, the extreme points are identified as characterizing flows following the shortest origin-destination path when a certain metric is defined on the arcs of the network. In the flexible manufacturing system, the extreme points correspond to certain weighted combinations of strategies. For this reason the adaptation is much closer to Defenderfer's tree flow (TR) formulation of the Cantor-Gerla algorithm (Defenderfer, 1977).

Substituting equations (3.24)-(3.26) into NLP 2.1 gives rise to the following non-linear programming problem with the variables w_ℓ

$$\text{NLP 3.2} \quad \max_{\underline{w}=(w_1,\ldots)} \; f\{\Sigma\, w_\ell \psi^\ell\} \tag{3.27}$$

subject to (3.25) and (3.25)

This problem can be solved by the gradient projection algorithm (Cantor and Gerla, 1974) (Luenberger, 1973). It is called the restricted master problem (Dantzig, 1963) if the set of all convex combinations of ψ^ℓ, referred to as the convex hull of ψ^ℓ , is a subset of Ω. The dimension of the set Ω is NA. Any element $x\varepsilon\Omega$ may be expressed as the convex combination of at most NA+1 linearly independent extreme points ψ^ℓ (Bazaraa, 1976). Thus if n <NA linearly independent vectors ψ^ℓ are available, only members of a certain subset (the convex hull of ψ^ℓ) of Ω can be expressed as the convex combination of the available extreme points ψ^ℓ.

Let x* be a solution to the restricted master problem. By applying the Karusch-Kuhn-Tucker optimality conditions for a mathematical program (Bazaraa, 1976) and noting that Ω is a convex set, x* is at least a local maximum if

$$e'(x-x^*) \leq 0 \tag{3.28}$$
$$\forall x \varepsilon \Omega$$

where $e = \left.\frac{\partial f}{\partial x}\right|_{x=x^*}$

If f(x) is a concave function, then f(x*) is the optimal value over the whole set. Thus to find the next extreme point to be incorporated in the solution procedure, the following linear program is solved (Cantor and Gerla, 1974).

LP 3.4

$$\text{maximize } ex \tag{3.29}$$

subject to (3.20) - (3.23)

This is the flow generating sub-problem and is the same as LP 3.1 save for the objective function, which does not include costs on x_i. This is reflected in the objective function in the de-coupled sub-systems which for LP 3.4 become $(e_i - \pi_i T_i)$ (Lasdon, 1970), where $e_i = \partial f/\partial x_i$.

Each of the extreme points ψ^ℓ can be expressed as

$$\psi^\ell = \sum_s q_s^* x^s = \sum_s \sum_i q_s^* x_i^s \tag{3.30}$$

where q_s^* is the solution of LP 3.2. The optimal solution of NLP 3.2 is thus $x^* = \sum_{\ell} w_{\ell}^* \psi^{\ell}$. The flow vectors x^s (as described in Section (3.2)) consist of single strategies for each type of workpiece. Thus the solution of NLP 3.2 contains all the information necessary to route all of the workpieces through the system.

The optimal point in the program is reached when for some solution ψ^{ℓ} to LP 3.4,

$$Q_{\ell} = e'(\psi^{\ell} - x^*) \leq 0 \tag{3.31}$$

The procedure may be terminated when Q_{ℓ} is below a given tolerance level with the assurance that x^* is always feasible.

The program NLP 2.1a, resulting from the incorporation of the transportation system into the optimization problem, can also be dealt with in the same way. Applying the method of Cantor and Gerla to this problem results in the following flow generating linear program

LP 3.5 maximize $e_r\, r$ (3.32)

subject to (2.27), (2.29), (2.30), (2.31), (2.43)
(2.46), and (2.47)

The vector r contains the elements x_{ij}^k and r_{ij}, and e_r is defined as

$$e_r = \frac{\partial f_r}{\partial r}\Big|_{r=r^*} \tag{3.33}$$

where $f_r(r)$ is the objective function (2.44).
This is a multi-commodity flow problem with shared resources at certain network nodes (the workstations). There is a further constraint in that the workpieces must pass through some specified nodes in going from the origin to the destination.

Applying the decomposition method of Section 3.2 to LP 3.5 results in the following sub-problems which have to be solved for each type of workpiece

LP 3.6 minimize $(\pi_i T_i - e_{ir})\, r_i$ (3.34)

subject to (2.27), (2.29), (2.30), (2.31), (2.43) and (2.46)

and $r_{oi} = \alpha_i$ (3.35)

with $e_r = (e_{ir},\ldots e_{pr})$ and $r = (r_1,\ldots r_p)$

The constraint (3.35) normalizes the solution vector so that the ratio-requirement constraint is satisfied when all of the sub-problems have been solved.

If $(\pi_i T_i - e_{ir})$ is interpreted as a vector of costs incurred in traversing each arc of the network, LP 3.6 can be interpreted as a constrained minimum cost routing problem for each type of workpiece. The constraint is that it must pass through one of the permissible work-stations for each operation. Provided there are no closed paths with a total negative cost, LP 3.6 has a bounded solution which involves sending all of the flow along the shortest path which passes through the required set of nodes.

The objective function from section 2.1 is

$$f_r(r) = \beta_1 \sum_i r_{io} - \beta_2\{\sum_{j=1}^{M} q_j(x) + g(r)\} \tag{3.36}$$

Thus the gradient $\frac{\partial f_r}{\partial r} = e_r$ is given by

$$\frac{\partial f}{\partial r_{ij}} = -\beta_2 \sum_j \frac{\partial q_j(x)}{\partial x_{ij}^k} \quad \text{at the workstations } j=1,\ldots M \tag{3.37}$$

$$\frac{\partial f}{\partial r_{ij}} = \begin{cases} -\beta_2 \dfrac{\partial g(r)}{\partial r_{ij}} & \text{on network arcs } j>m \quad (3.38) \\ \beta_1 - \beta_2 \dfrac{\partial g}{\partial r_{io}} & \text{on the input arc } j = 0 \quad (3.39) \end{cases}$$

In actual systems, the travel time where congestion effects are present is an increasing function of the flow rate. Thus $\partial g(r)/\partial r_{ij}$ is a positive quantity. Similarly the average queue length $q_i(x)$ is an increasing function of x. Thus the metrics on all the arcs of the network except the one from the loading station (j=0) are positive. The arc j=0 is an input arc and does not form a part of any closed path. With this formulation, therefore, there are no closed paths with a total negative cost. Shortest-path algorithms may then be used to solve the shortest-path problem. Dreyfus (1969) reviews constrained shortest-path algorithms

which could be used to solve the sub-problems. Kershenbaum and Golden (1976) solve constrained shortest-path problems where the order of visiting the nodes is not specified by a labelling algorithm. This could be useful where the ordering of the operations is not resolved. The use of shortest-path algorithms in flexible manufacturing systems should prove to be fruitful because they are unlikely to have as a dense a network as occurs in transportation and computer communication systems. For typical shortest-path algorithms, the number of computer operations required to solve the problem is of the order of N^3 where N is the number of nodes in the network. The network model of a manufacturing system such as that of Fig. 1.1 has far fewer nodes than, for example, urban traffic networks in which shortest-path algorithms have been applied (Nguyen, 1974). Furthermore, the ratio of the number of arcs to the number of nodes in a manufacturing system is relatively small. This can lead to savings in the computation time of shortest-path algorithms (Steenbrink, 1974).

A flow graph summarizing the TR formulation of the Cantor-Gerla algorithm is given in Fig. 3.2. The method has the advantage that only convex combinations of the extreme points are considered in the non-linear optimization. The number of variables is consequently much less than in the original problem. If at some stage the weight w_k on some extreme point is zero, it can be dropped from the set thereby keeping the number of variables small. Finally the application of the algorithm resolves the routing problem in the flexible manufacturing system.

The problem NLP 2.2 with non-linear constraints can be expressed as

NLP 3.3

$$\underset{x\in\Omega}{\text{minimize}}\ f(x) \tag{3.40}$$

$$\text{subject to } h(x) \leq 0 \tag{3.41}$$

where Ω is the set of feasible flows and

$$h(x) = \left\{ \sum_{i=1}^{M} q_i(x) - Q \right\} \tag{3.42}$$

The solution to NLP 3.3 is the pair $(\bar{x},\bar{\eta})$ which satisfies the Karusch-Kuhn-Tucker conditions (Bazaraa, 1976). Namely

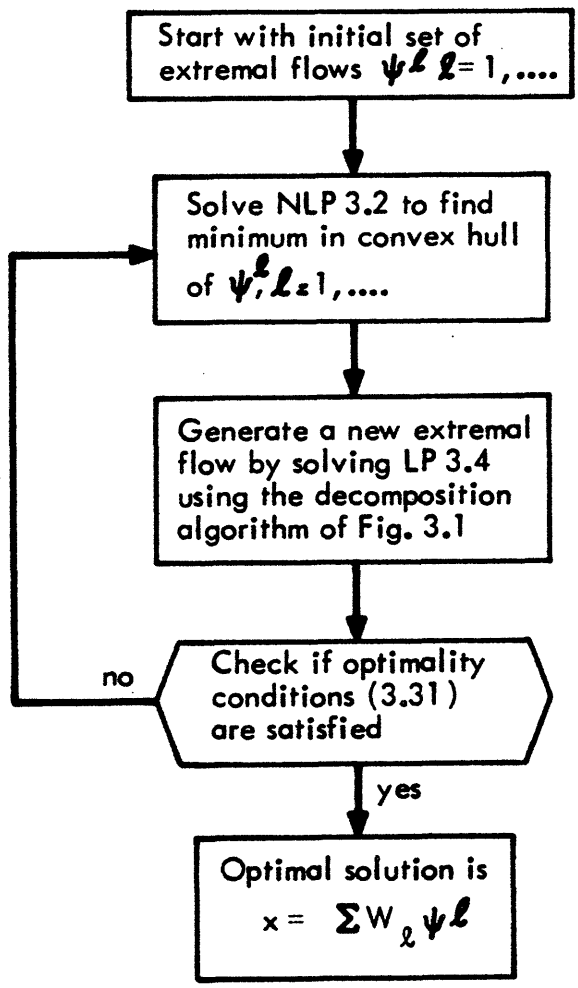


Fig. 3.2. Tree Flow Formulation of the Cantor-Gerla Extremal Flow Algorithm

$$\frac{\partial f(\bar{x})}{\partial x} + \bar{\eta} \frac{\partial h(\bar{x})}{\partial x} = 0 \tag{3.43}$$

$$\bar{\eta} h(\bar{x}) = 0 \tag{3.44}$$

$$\bar{\eta} \geq 0 \tag{3.45}$$

If $f(x)$ and $h(x)$ are convex functions then $f(\bar{x})$ is a global minimum within Ω. otherwise it is a local minimum.

The classical penalty function method attaches the constraint (3.41) to the objective function by means of a quadratic penalty function. The problem NLP 3.4 is then solved for an increasing penalty weight W_s,

NLP 3.4

$$\underset{x\varepsilon\Omega}{\text{minimize}} \quad \phi(x,W_s)=f(x)+W_s\{\theta[h(x)]\}^2 \tag{3.46}$$

$$\text{where} \quad \theta(t) = \begin{cases} t & t \geq 0 \\ 0 & t < 0 \end{cases} \tag{3.47}$$

If x_s solves NLP 3.4, it is known (Luenberger, 1973) that

$$\lim_{W_s \to \infty} x_s = \bar{x} \tag{3.48}$$

$$\lim_{W_s \to \infty} W_s \, h(x_s) = \bar{\eta} \tag{3.49}$$

The penalty function approach suffers from numerical difficulties because the Hessian of $\phi(x,W_s)$ becomes ill-conditioned as W_s becomes large. Furthermore, succeeding steps of the optimization process do not make use of previous information on $f(x)$ and $\phi(x)$ (Himmelblau, 1972).

The Langrangian function for NLP 3.3 is defined as

$$L(s,\eta_s) = f(x) + \eta_s \, h(x) \tag{3.50}$$

The dual function is defined as

$$D(\eta_s) = \min_{x\varepsilon\Omega} L(x,\eta_s) \tag{3.51}$$

It can be shown that (Bazaraa, 1976)

$$f(\bar{x}) \leq \sup_{\eta} D(\eta) \tag{3.52}$$

A necessary condition for the inequality to be satisfied as equality is that Ω, $f(x)$ and $h(x)$ should be convex. A duality gap is said to exist otherwise (Bazaraa, 1976) (Lasdon, 1969). Where there is no duality gap, NLP 3.3

could be solved by way of the dual function (3.51), with the knowledge that $\bar{\eta}$ is the solution to

$$\sup_{\eta \geq 0} D(\eta) \tag{3.53}$$

and $\bar{x}$ solves

$$\min \; L(x,\bar{\eta}) \tag{3.54}$$

This method has been used to decompose large non-linear programs (Lasdon, 1968).

Hestenes (1969) and Powell (1968) introduced the penalty Lagrangian method in order to overcome the disadvantages of classical penalty methods. Rockafellar (1973) (1974) extended the method to inequality constrained problems and gave convergence proofs using duality theory.

The algorithm is as follows

NLP 3.5

$$\underset{x\in\Omega}{\text{minimize}}\; L(x,\eta_s,W)=\begin{cases} f(x)+\eta_s h(x)+\frac{W}{2}[h(x)]^2 & h(x)\geq \frac{-\eta_s}{W} \\ f(x)-\frac{\eta_s}{4W} & h(x)< \frac{-\eta_s}{W} \end{cases} \tag{3.55}$$

If x_s solves NLP 3.5, apply the update (3.56) and repeat

$$\eta_{s+1} = \begin{cases} \eta_s + W\, h(s_s) & h(x_s)\geq \frac{-\eta_s}{W} \\ 0 & h(x_s) < \frac{-\eta_s}{W} \end{cases} \tag{3.56}$$

The solution is reached when conditions (3.43)-(3.45) are satisfied. This particular form of the Lagrangian function has the advantage that it has a continuous first derivative. The dual associated with $L(x,\eta,W)$ is

$$D(\eta,W) = \min_{x\in\Omega} \; L(x,\eta,W) \tag{3.57}$$

It can be shown (Rockafellar, 1973, 1974) that for W sufficiently large but finite

$$D(\bar{\eta},W) = \sup D(\eta,W) = \inf_{x\in\Omega} L(x,\bar{\eta}) = f(\bar{x}) \tag{3.58}$$

and is independent of W.

Furthermore,

$$\frac{\partial D(\eta_s,W)}{\partial \eta} = \max\ (h(x_s),-\frac{\eta_s}{W}) \tag{3.59}$$

The update rule (3.56) is thus a fixed-step-length, steepest ascent maximization of $D(\eta_s,W)$.

Bertsekas (1975) has explored the convergence properties of penalty Lagrangian methods. They have arbitrarily fast linear convergence rates. That is, the rate at which η_s converges to $\bar{\eta}$ and hence x_s to $\bar{x}$ is linear in the number of steps. The rate can be arbitrarily varied by choosing the weight W on the quadratic penalty term. Several methods (Betts, 1977) (Miele, 1971, 1972) have been suggested for increasing the rate of convergence by altering the updating rule (3.56).

In practice, the overall performance of the algorithm is found to depend also on the penalty weight W. If W is large, η_s converges rapidly to $\bar{\eta}$. However, the number of gradient steps required to minimize $L(x,\eta_s,W)$ grows with W. This is because the Hessian of $L(x,\eta_s,W)$ becomes increasingly ill-conditioned as W becomes larger. An improved update rule for the η_s allows a smaller penalty weight W to be used while maintaining a favorable convergence rate for η_s. An extrapolation method has been found to be quite effective in speeding the convergence of the algorithm. The method is given here for one constraint but it can be generalized to several constraints. For more than one constraint, the advantage would have to be weighed against the additional effort required for matrix inversion.

Define $h_\eta(\eta_s)$ as

$$h_\eta(\eta_s) = h(x) \Big|_{x=x_s \varepsilon \Omega} \tag{3.60}$$

where as before x_s minimizes $L(x,\eta_s,W)$.

The solution of NLP 3.3 occurs at $\bar{\eta}$ where $h_\eta(\bar{\eta})=0$. Figure 3.3 is a graph of $h_\eta(\eta)$. Minimizing $L(x,\eta_s,W)$ obtains the point P1 on the graph. The update (3.56) produces η_{s+1}, and the subsequent minimization of the penalty Lagrangian produces P2. At this stage rather than applying (3.56), a linear approximation is made to $h_\eta(\eta)$ and the next estimate of $\bar{\eta}$ is made by extrapolation to give

$$\eta_{s+2} = \eta_{s+1} + \frac{h_\eta(\eta_{s+1})\ \{\eta_{s+1}-\eta_s\}}{h_\eta(\eta_s) - h_\eta(\eta_{s+1})} \tag{3.61}$$

Subsequent values of η_s are obtained by extrapolating from the two latest

87049AW034

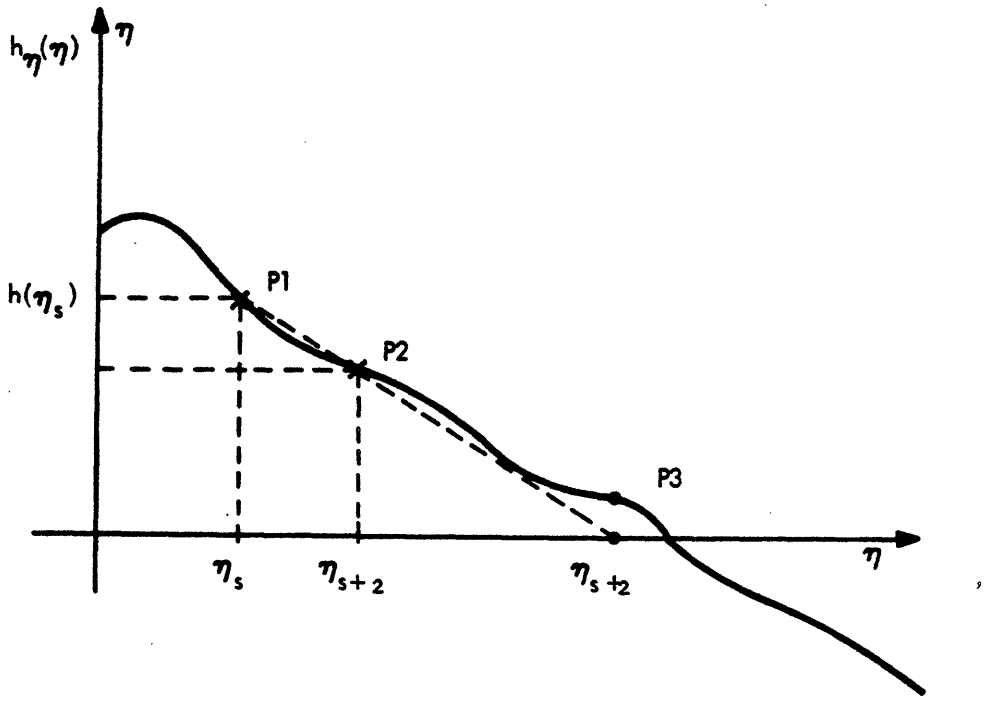


Fig. 3.3. Graph of $h_\eta(\eta)$ Showing the Extrapolation Step

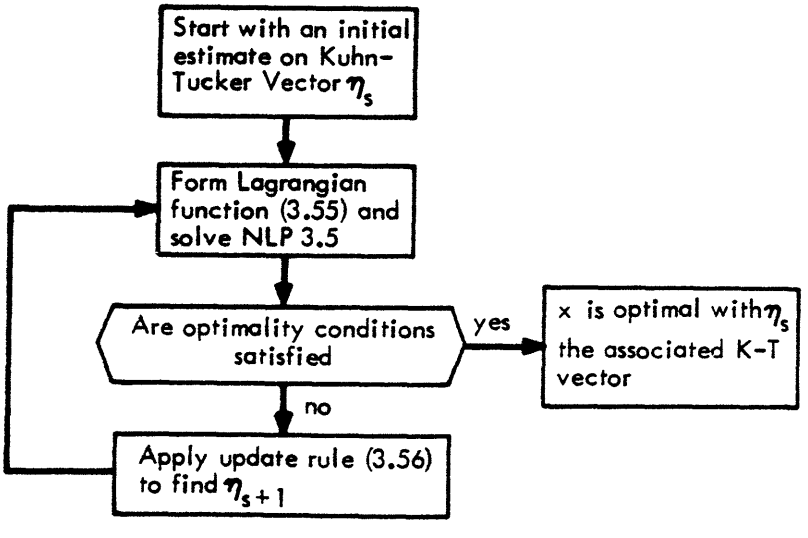


Fig. 3.4. The Augmented Lagrangian Algorithm

estimates. This is equivalent to a Newton method (Luenberger, 1973) because in extrapolating, an approximation is being made to the second derivative of $D(\eta)$.

The minimization of the Lagrangian function NLP 3.5 subject to a linear constraint set can be done using the extremal flow algorithm, Fig. 3.2. The x-variables in NLP 3.5 are then the weights w_k on the extreme points Ψ^k. Fig. 3.4 is a flow chart summarizing the algorithm.

3.4 Conclusion

Optimization methods which exploit the structure of the flexible manufacturing system models have been discussed. The decomposition method for linear programming problems has the strategy generating sub-problems LP 3.3 which are co-ordinated by a linear program with fewer constraints than the original problem. The method is iterative. Strategies are generated successively until the co-ordinating program finds an optimal combination. Thus only a subset of all the possible paths through the system is considered.

The extremal flow algorithm for optimizing non-linear objective functions, subject to linear constraints, expresses the flow vector x as a convex combination of the extreme points of the feasible flow set Ω. The extreme points are generated as required by linearizing the objective function at a current point and solving the resulting linear program by the decomposition method of section 3.2. In this way, not all extremal flows need to be considered in order to arrive at an optimal solution.

The augmented Lagrangian algorithm converts the nonlinearly constrained problem into problems with only linear constraints. The extremal flow algorithm can then be used to minimize the Lagrangian functions $L(x,\eta_s,W)$ subject to the linear constraint set. An attractive feature of this scheme is that when η_s is close to the Karusch-Kuhn-Tucker vector $\bar{\eta}$, the optimal points of the Lagrangian function $x(\eta_s)$ are close to the optimum $\bar{x}$. The number of additional strategies that have to be generated in order to optimize successive Lagrangians is then small.

4. NUMERICAL RESULTS FOR TWO- AND FOUR-WORKSTATION SYSTEMS

4.1 Introduction

In order to test the applicability of the ideas of Chapter 2 and 3, some hypothetical systems were postulated and optimized. In the stochastic case, a two-workstation system with two different kinds of pieces was used. The workstations were assumed to have an exponential service time distribution with Poisson arrival processes. This model serves a useful role as a test bed for the proposed algorithms and for gaining insight into how the optimal strategy choice depends on system parameters.

One might ask how realistic these assumptions are. At the present stage of the investigation the types of variation in the duration of time a workpiece spends at a workstation are not known. However, in the Baskett et al. (1975) model, the behavior of a network of queues is strongly dependent on the mean of the service and interarrival time distributions and not on higher moments. Particularly relevant is the operational result of Denning and Buzen (1977) which is derived without making any assumptions as to the distribution of random processes involved. It can be expected that systems with non-exponential servers and non-Poisson arrival processes will behave in the same qualitative manner.

The assumptions made will not fit actual systems. The exponential distribution has the memoryless property. Thus no matter how long a piece has been at a workstation, the time remaining until service is complete is still exponentially distributed with the same mean. This is not a realistic assumption for many manufacturing processes. The same observation holds for a Poisson arrival process for which the time between arrivals is exponentially distributed. The effect of these factors is to make calculations of average queue lengths less accurate. The effect on the optimal strategies is difficult to judge, and will require some study.

Section 4.2 presents the model and optimization results for different values of system parameters. In Section 4.3 the linear model for systems which are nearly deterministic is applied to a four-workstation simulation where the size of the in-process inventory is not of concern. The flow optimization results are used to run a simulation of the system and the results are discussed.

4.2 Optimization Results for a Two-Workstation System

Consider the system depicted in Figure 4.1. The work-stations and the loading station have exponentially distributed service times with mean $1/\mu_i$ at station i. The service time distribution is independent of the type of piece being worked on.

There are two types of pieces being manufactured. The first needs one operation which may be performed at either workstation. The second needs two operations, one at workstation 1 and the second at workstation 2. The operations can be done in either order. The four possible strategies, two for each piece, are summarized in Figures 4.2 and 4.3. Also shown are the t_{ij}^k matrices from which the strategies are derived by the method of Section 2.3.1. In this case the four strategies are easily identified. The variables y_ℓ $\ell = 1,\ldots,4$ represent the flow rate of strategy ℓ pieces into the system.

The ratio requirement is that two type 2 pieces should be produced for every type 1 piece. This can be expressed mathematically as

$$2(y_1 + y_2) - (y_3 + y_4) = 0 \tag{4.1}$$

The total production rate is

$$R = \sum_{\ell=1}^{4} y_\ell \tag{4.2}$$

The in-process inventory I consists of pieces on the transportation system and pieces awaiting service at the workstations. The pieces travel at constant speed on the transportation system. Thus the travel time is proportional to how far a piece travels while following a certain strategy. The travel time on each arc of the network (see Fig. 4.1) is taken to be τ (independent of the arc). The travel time for each strategy is given in Table 4.1. This assumes that no piece is ever rejected from a workstation, which is consistent with the assumption that input queues have infinite capacity. The number of pieces in the transportation system is thus on average

$$\tau[4(y_1 + y_2) + 3y_3 + 9y_4] \tag{4.3}$$

The system is modeled as an open network of queues. Thus at workstation j, the average queue length is given by (Kleinrock, 1975)

$$q_j = \frac{\sum_{\ell \in M(j)} y_i}{\mu_j - \sum_{\ell \in M(j)} y_\ell} \quad j = 0,1,2 \tag{4.4}$$

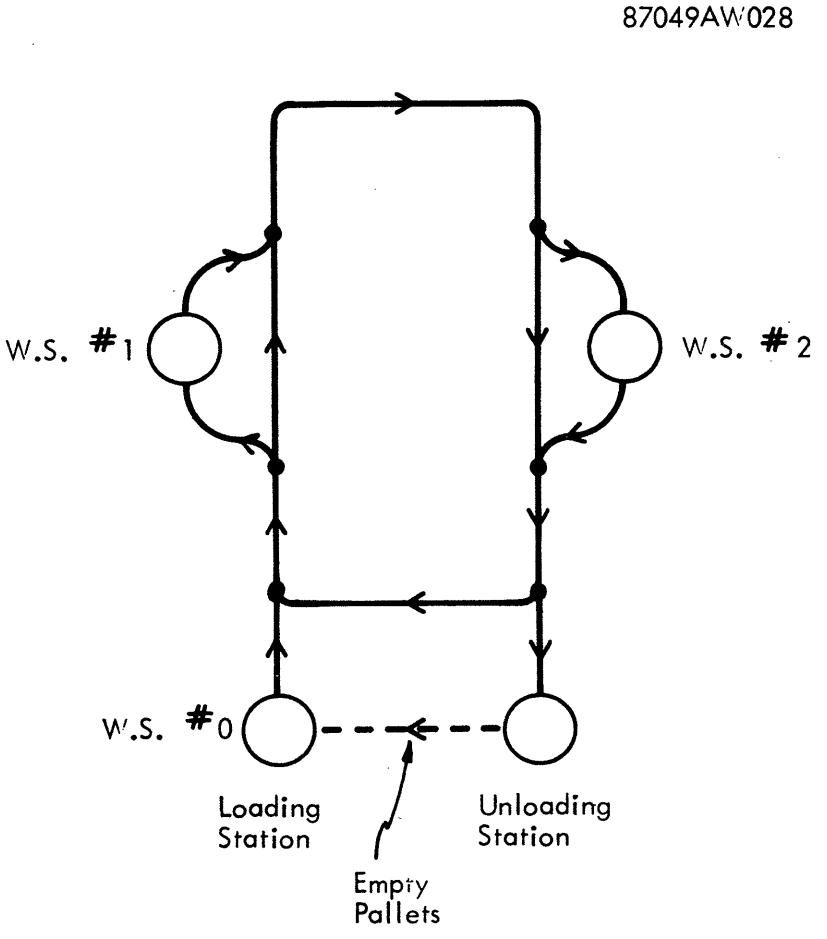


Fig. 4.1. A Two-Workstation System

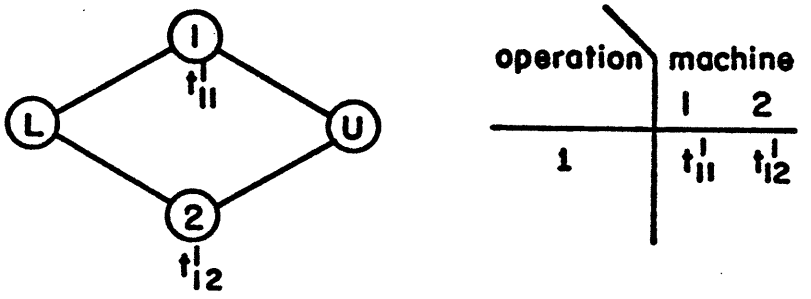


Fig. 4.2. Machining Options for Type 1 Pieces

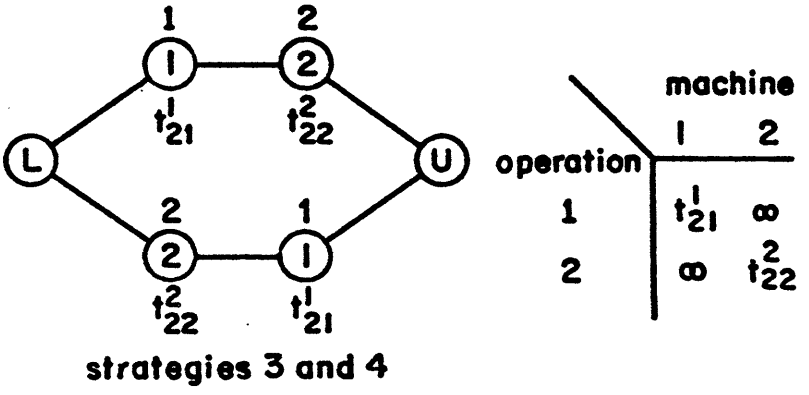


Fig. 4.3. Machining Options for Type 2 Pieces

strategy	1	2	3	4
τ_ℓ	4τ	4τ	3τ	9τ

Table 4.1: Average Time τ_ℓ on the Transportation Network for each Strategy, Two-Workstation Example

Workstation j	speed μ_j in pieces/hr.	Average Operation time (minutes) $60/\mu_j$
loading station	30	2
1	6	10
2	5	12

Average travel time on each arc 1.2 minutes

Table 4.2: System Parameters, Two-Workstation Example

where M(j) is the set of strategies that use workstation j.

The loading station is labelled j = 0. An arrival at the loading station is considered to be an order by the system controller to load a particular piece. The queue at the loading station is thus the pieces and pallets awaiting service because their loading orders cannot be carried out immediately. The empty pallets and raw material inventory are thus not included in $q_o(y)$. An accurate assessment of the utilization of the loading station is obtained since its service rate is matched against the arrival rate of load commands.

Combining (4.3) and (4.4) gives the total average inprocess inventory as

$$I_y(y) = \sum_{j=0}^{2} q_j + \tau[4(y_1 + y_2) + 3y_3 + 9y_4] \tag{4.5}$$

The optimization problem is to maximize the production rate while keeping the average in-process inventory below a set level Q. This is stated as

NLP 4.1 $$\text{Maximize } R = \sum_{\ell=1}^{4} y_\ell \tag{4.6}$$

subject to (4.1) and

$$I_y(y) \leq Q \tag{4.7}$$

$$y_\ell \leq 0 \tag{4.8}$$

The constraint on the average level of in-process inventory is expressed in (4.7) where Q is the desired level and is motivated by the desire to keep the input flow rate $\sum_{\ell=1}^{4} y_\ell$ into the system at a level which does not overwhelm the system. If a non-deterministic system is offered work at a rate very close to its service rate, the result is a rapid build up in the queue lengths at the servers (Kleinrock, 1975). If a complete congestion model were available that took into account queue blocking, overflow onto the transportation system and congestion effects on the network links there would be no need to limit input flow rate by use of constraint (4.7). However, limiting the size of the in-process inventory is often desirable because floor space and the pallets needed add to the cost of the system.

The parameter values used in the experiment are shown in Table 4.2. In the first experiment the value of Q was varied from 2 to 10 and the resulting optimal strategies, production rates, and station utilizations are shown in Figs. 4.4 - 4.8.

The optimization was carried out using the augmented Lagrangian algorithm. The speed of the algorithm is found to depend on how fast the estimate to the Kuhn-Tucker vector converges to the true value and on the number of line searches needed in the unconstrained minimization of the Lagrangian function. Typically in this example with two explicit constraints, it takes less than five Lagrangian minimizations to find the maximum to an accuracy of 10^{-5} in the constraint function value. Execution time is about 2 seconds (C.P.U) but can be improved significantly by the use of an improved unconstrained minimization algorithm.

It should be noted that if there is no constraint on the average in-process inventory, the problem becomes a linear program. The optimal solution of this program is the limiting maximum production rate of the system as $Q \to \infty$.

For all values of Q, type 2 pieces always follow strategy 3. That is, they go to station 1 first and then to station 2. This is because strategy 4, which involves extra travel, increases the in-process inventory without a corresponding increase in production.

The proportion λ of type 1 pieces that are sent to workstation 1 (referred to as the optimal split) is shown in Fig. 4.4 as a function of Q. When the average in-process inventory is low, the optimal split is high since workstation 1 is the fastest station. As the number of pieces in the system increases, more type 1 pieces are diverted to workstation 2. Secco-Suardo (1978) found a similar change for a system modeled as a closed network. In his formulation, the optimal split depends on the number of pallets available. As can be expected, the production rate increases with Q (Fig. 4.5.) but a saturation effect is in evidence. The maximum possible production rate is 6.6 pieces per hour when the restriction on the average level of in-process inventory is lifted (i.e., as $Q \to \infty$). Both stations are then fully utilized.

The effect of increasing Q on the utilizations of the workstations and their queue lengths is shown in Figs. 4.6 and 4.7 respectively. The utilization of the workstation increases in a way which keeps the queue lengths approximately in constant proportion. Thus as the average level of in-process inventory is increased, the optimal split changes in a way that keeps the workstations balanced in the sense that their levels of utilization are approximately equal.

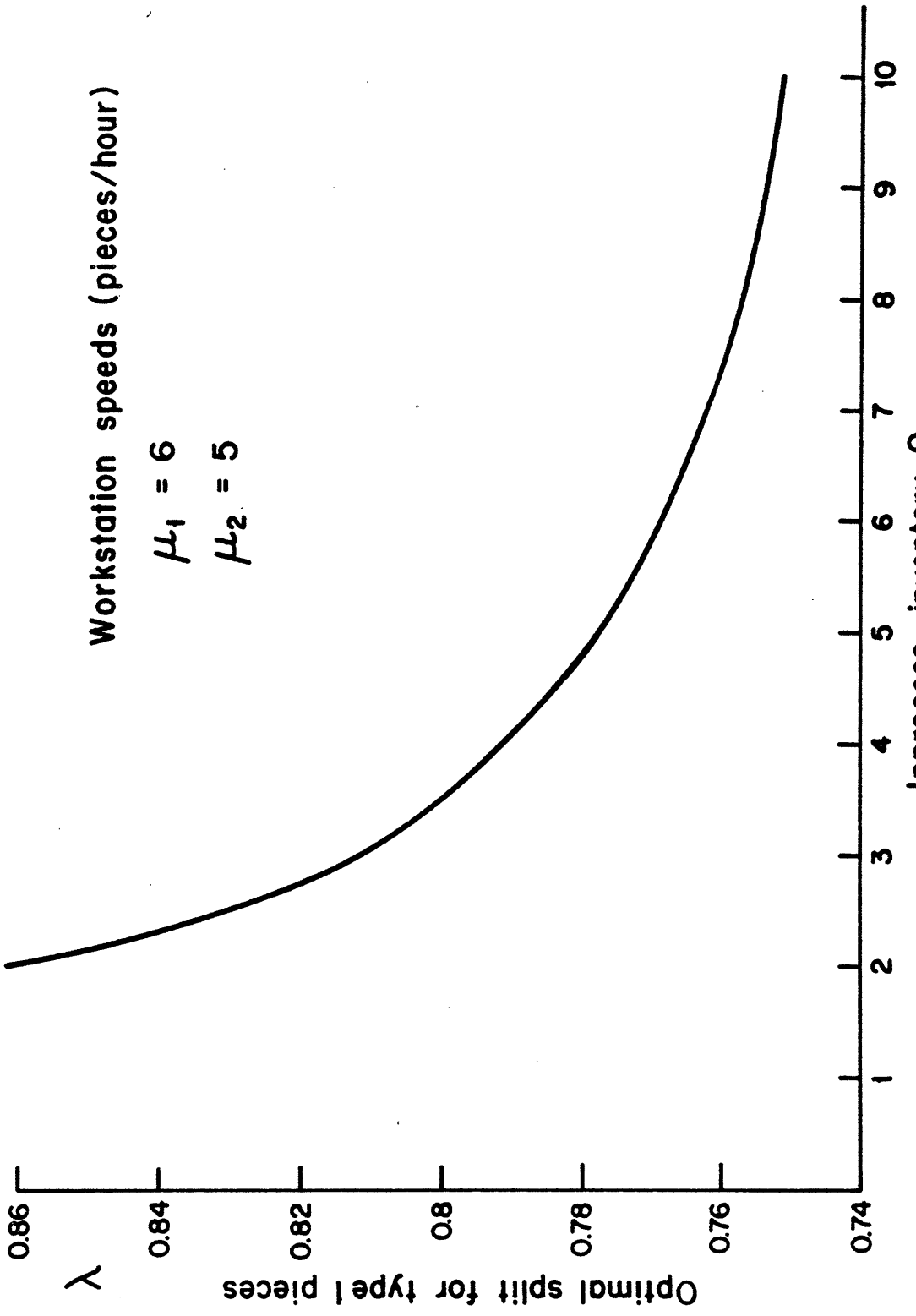


Fig. 4.4. Optimal Mix λ as a Function of In-process Inventory, Q

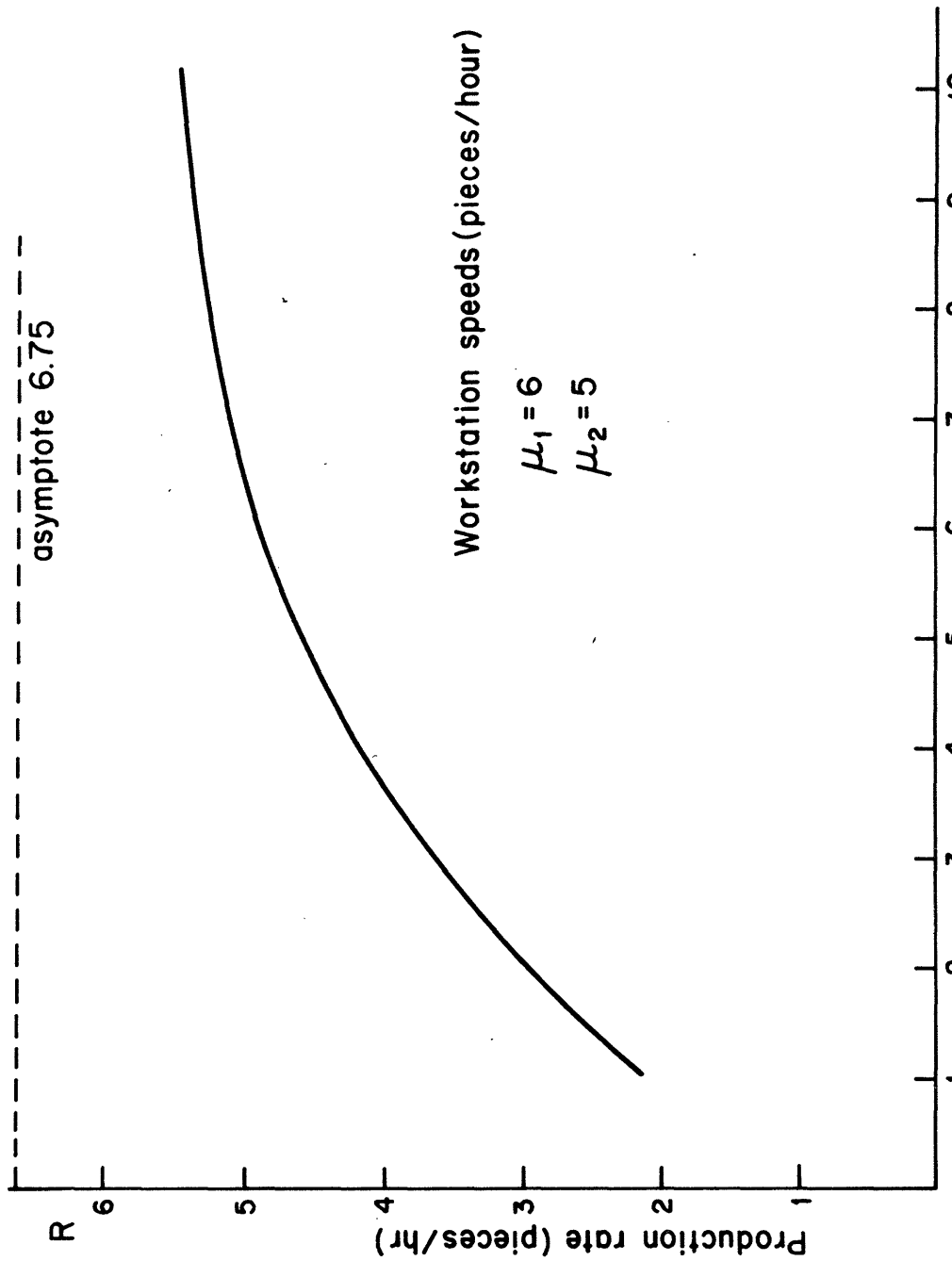


Fig. 4.5. Optimal Production Rate As a Function of In-Process Inventory

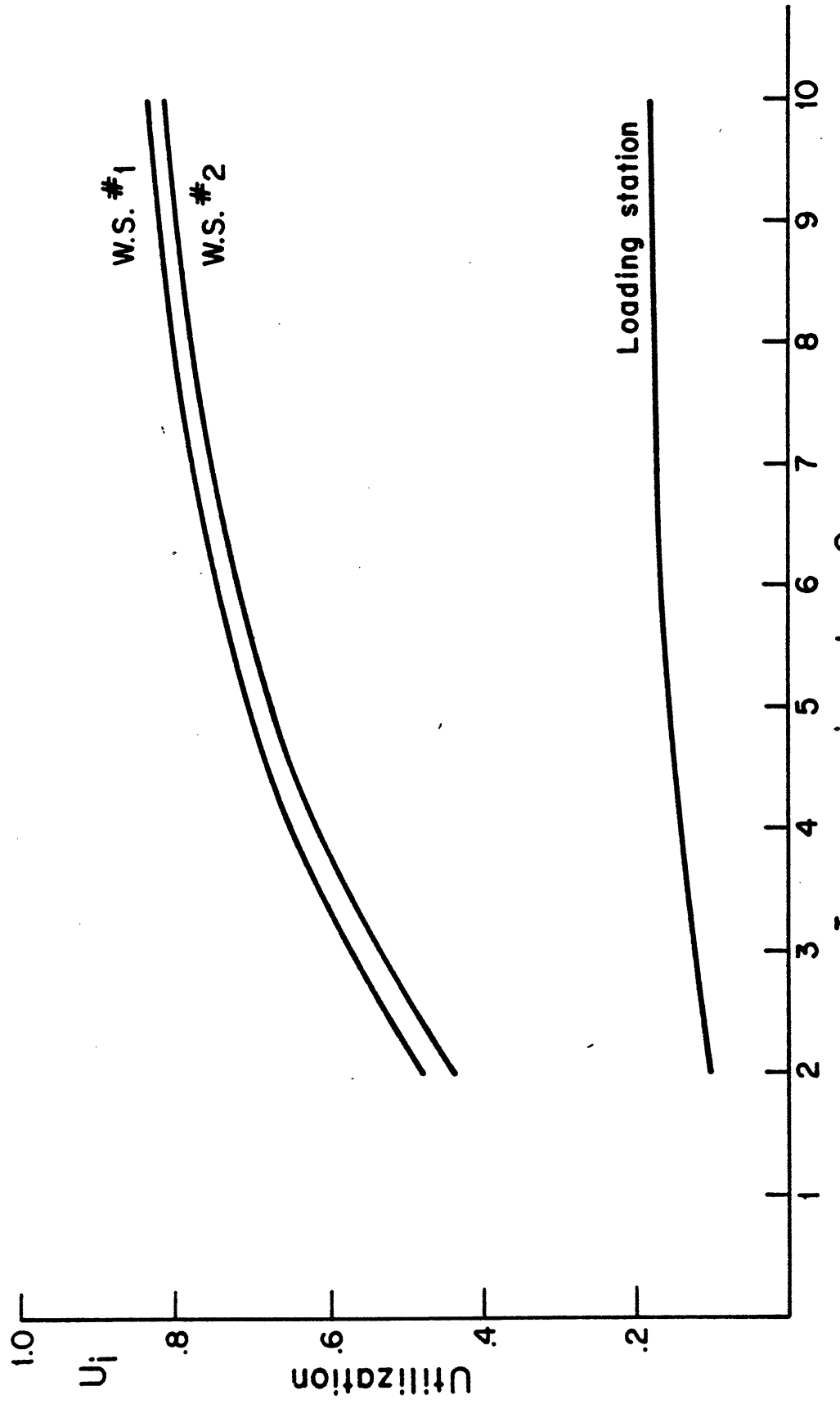


Fig. 4.6. Optimal Workstation Utilization as a Function of In-process Inventory

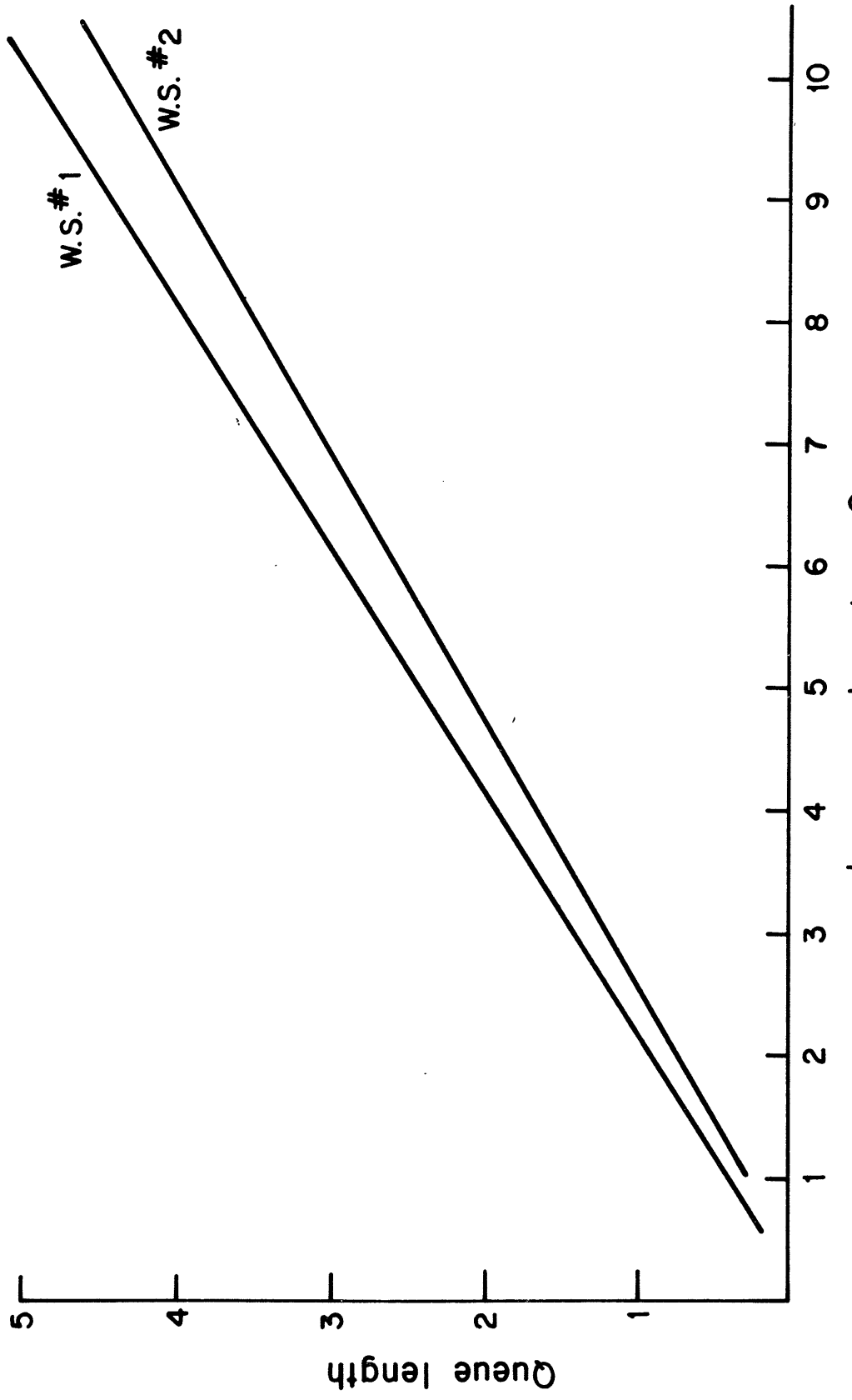


Fig. 4.7. Optimal Average Queue Lengths As a Function of In-Process Inventory

A useful output of the optimization program is the value of the Lagrange multiplier associated with the constraint on average in-process inventory. As can be seen in Fig. 4.8, the multiplier decreases as Q is increased. Since the multiplier can be interpreted as the rate of change in the optimal objective function, with Q, the return in terms of increased production rate decreases dramatically as Q is increased.

In the second set of results, the average value of in-process inventory Q is required to be 10. The speed μ_2 of workstation 2 is fixed at 5 pieces per hour, and that of workstation 1, μ_1, is varied from 2 to 10 pieces per hour. The results are compared to the asymptotic case in which there is no limit on Q.

The optimal split for type 1 pieces for $Q = 10$ and $Q = \infty$ are shown in Fig. 4.9. The difference between the two is small. There are three operating regimes. When μ_1 is very small compared to μ_2 all type 1 pieces are sent to workstation 2. Similarly if μ_1 is large compared to μ_2 the optimal split is unity and all type 2 pieces to to workstation 1.

This would indicate that when the difference in speed between the two workstations is great, it is not worthwhile making the slower station flexible. Even if it has the capability of performing operations on a type 1 piece, it is not utilized. On the other hand this flexibility may be valuable when the faster machine is unavailable due to a failure or to routine maintenance.

In the range where the speed of workstation 1 is about $\pm$ 40% that of workstation 2, the optimal split changes rapidly from zero at the lower speed to unity at the higher speed.

The three regions are evident in the effect on utilization and average queue lengths shown in Figs. 4.10 and 4.11 for $Q = 10$. The change in optimal split keeps the utilizations of the two stations close to each other. For this system, at least, the optimization produces a balanced load on the two workstation.

The utilization u_1 of workstation 1 does not decrease monotonically as μ_1, the speed of the station, increases. This counter-intuitive behavior can be examined by changing the variables in NLP 4.1. Let R be the production rate

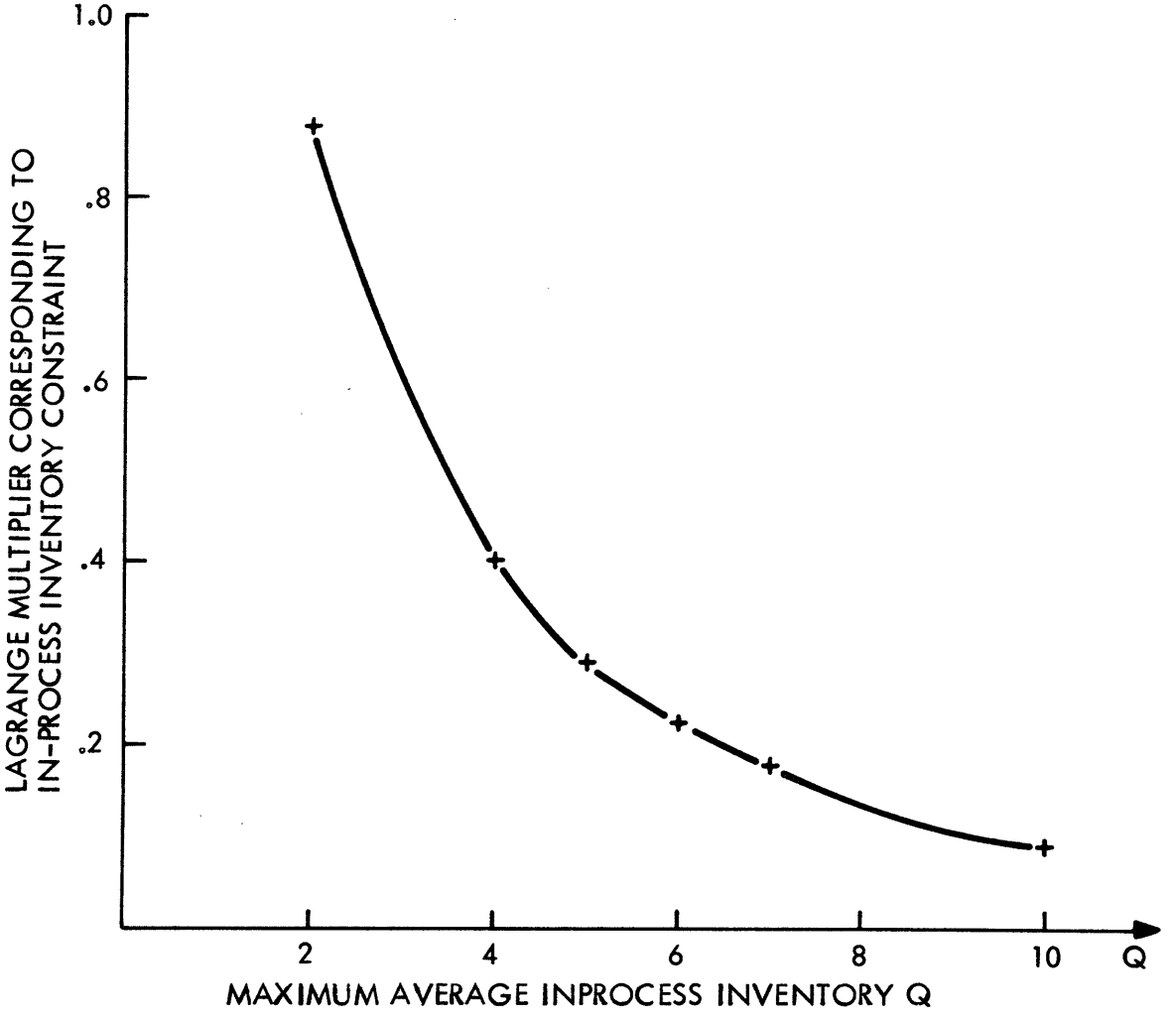
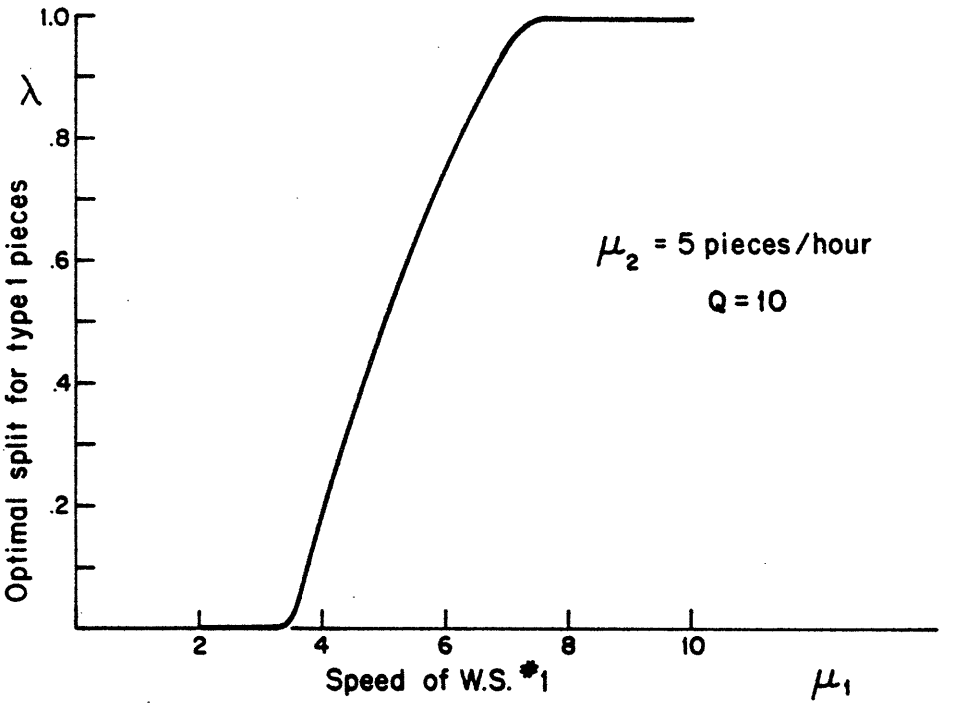


Fig. 4.8. Optimal Value of the Lagrange Multiplier as a Function of In-Process Inventory



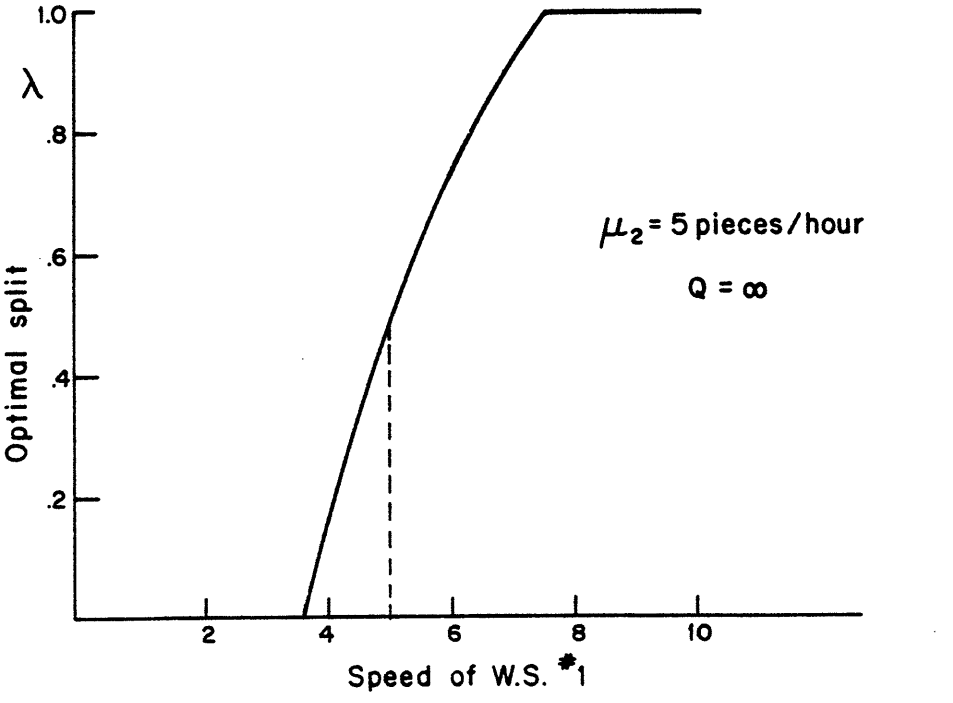


Fig. 4.9. Optimal Split λ as a Function of μ_1

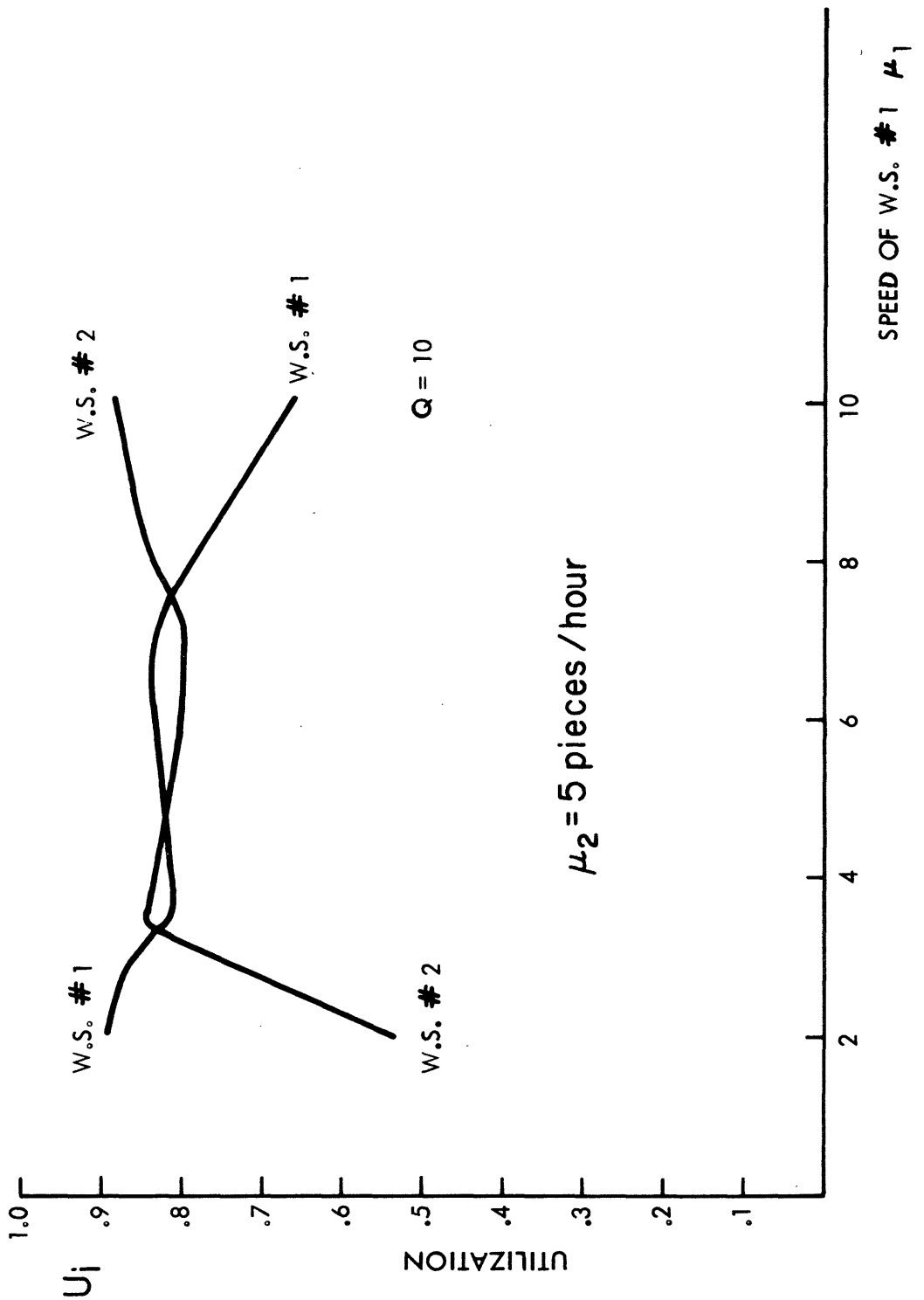


Fig. 4.10. Optimal Workstation Utilization as a Function of μ_1 with Q=10

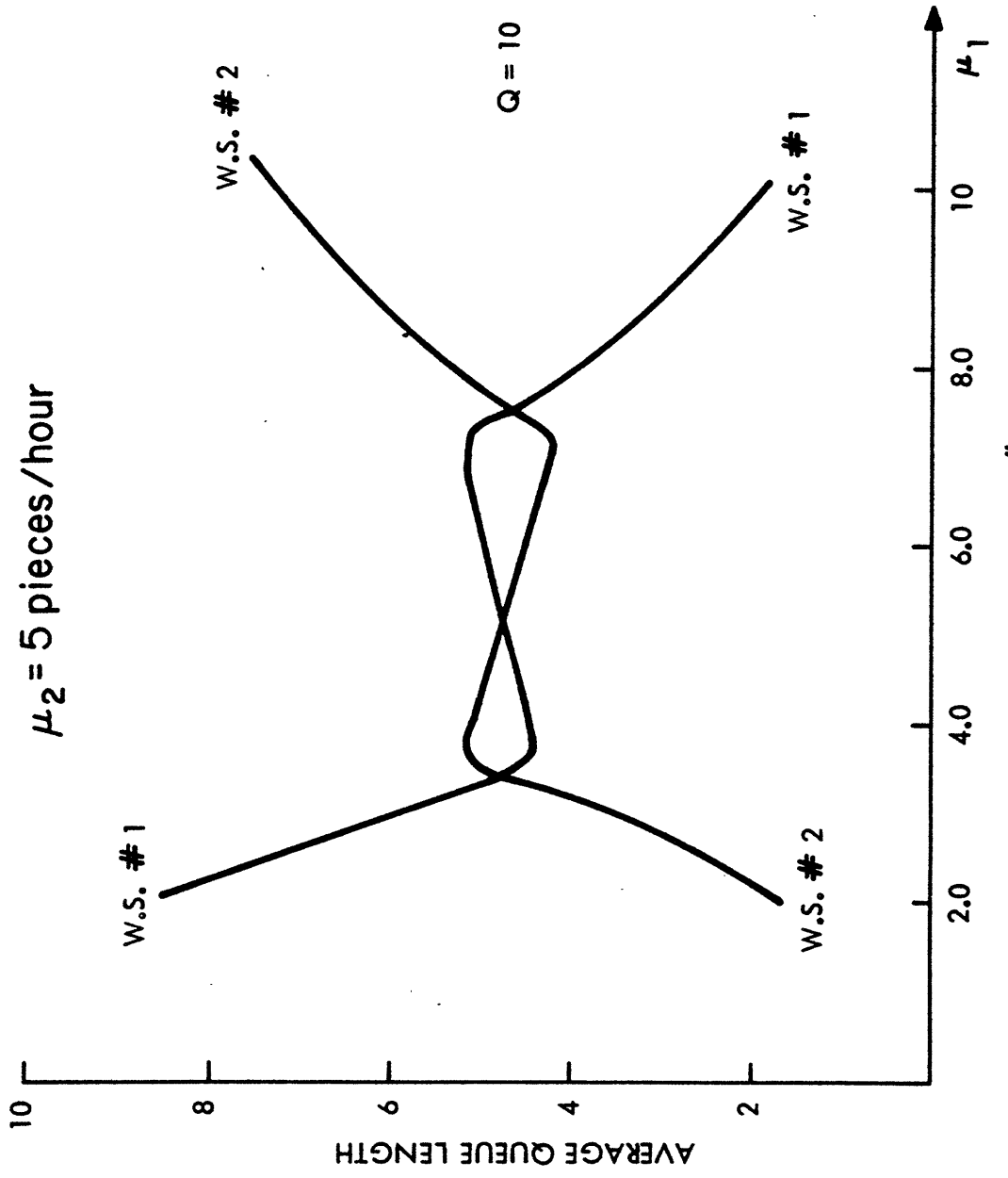


Fig. 4.11. Optimal Average Queue Lengths As a Function of μ_1. With Q=10

and λ the split for type 1 pieces. Since strategy 4 is never used, the strategy flow rates can be expressed using the ratio requirement constraint as

$$y_1 = \frac{\lambda R}{3} \tag{4.9}$$

$$y_2 = (1-\lambda)\frac{R}{3} \tag{4.10}$$

$$y_3 = \frac{2}{3} R \tag{4.11}$$

The utilizations u_1 and u_2 of the workstations are, from (4.9) - (4.11),

$$u_1 = \frac{R}{3\mu_1} \tag{4.12}$$

$$u_2 = \frac{R}{3\mu_2}(3-\lambda) \tag{4.13}$$

where μ_1 and μ_2 are the service rates at workstations 1 and 2, respectively.

The number of pieces on the transportation network and at the loading station is negligible compared to those queuing at the workstations. The optimization problem NLP 4.1 can thus be stated as

NLP 4.2

$$\underset{R,\lambda}{\text{Maximize}}\ R \tag{4.14}$$

subject to

$$I_\lambda(R,\lambda) = \frac{u_1}{1-u_1} + \frac{u_2}{1-u_2} \leq Q \tag{4.15}$$

$$\lambda \leq 1 \tag{4.16}$$

$$\lambda \geq 0 \tag{4.17}$$

$$R \geq 0 \tag{4.18}$$

The in-process inventory constraint (4.15) limits the total average number of pieces queueing at both workstations. It results from the substitution of (4.9) - (4.13) into (4.4).

The problem may be solved algebraically by applying the Kuhn-Tucker optimality conditions. The Lagrangian function is

$$L(R,\lambda) = R + \pi_1(I_\lambda(R,\lambda)-Q) + \pi_2(\lambda-1)-\pi_3\lambda \tag{4.19}$$

where $(\pi_1,\pi_2,\pi_3,)^T \geq 0$ is the Kuhn-Tucker vector.

Firstly, it should be noted that the optimal solution always occurs on the boundary $I_\lambda(R,\lambda) = Q$. By equating $I_\lambda(R,\lambda)$ to Q, for fixed λ, a

quadratic equation in R results

$$A_1A_2(Q+2)R^2 - (Q-1)(A_2\mu_1+A_1\mu_2)R + \mu_1\mu_2Q = 0 \qquad (4.20)$$

where

$$A_1 = (2+\lambda)/3$$

$$A_2 = (3-\lambda)/3$$

If $\lambda = 0$, necessary conditions for optimality are $\pi_2=0, \pi_3>0$ and $\partial L(R,\lambda)/\partial\lambda > 0$. Using (4.19), these conditions imply that

$$\left[\frac{\mu_1}{\mu_2}\right]^{1/2} > (\mu_1 - \frac{2}{3}R_0)/(\mu_2 - R_0) \qquad (4.21)$$

with R_0 being the solution of (4.20) for $\lambda=0$. The solution of (4.21) depends on the average in-process inventory Q through (4.20). For fixed μ_2, the range of μ_1 in which it is optimal to have a mix of strategies for type 1 pieces thus depends on the average level of in-process inventory Q.

The roots of (4.20) as a function of μ_1 are plotted in Fig. 4.12 with the split, λ, as a parameter, $Q = 10$ and $\mu_2 = 5$. For workstation 1 speed $\mu_1 \leq 3.3$, the highest production rate is achieved with $\lambda = 0$. In this range of μ_1 where $\lambda = 0$ is optimal, the production rate growth is approximately linear with μ_1 from 0 to 3, but falls off thereafter. Larger values of λ then become optimal as μ_1 increases beyond 3.

The effect of μ_1 on workstation utilizations u_1 and u_2 is shown in Figs. 4.13 and 4.14, with λ as a parameter. For the $\lambda = 0$ case, all type 1 pieces are sent to workstation 2, and workstation 1 handles only type 2 pieces. Because of the production ratio requirement, workstation 1 is the bottleneck station for low values of μ_1, and constrains both u_2, the utilization of workstation 2, and the production rate, R. Both u_2 and R rise with μ_1, linearly at first, and then more slowly as u_2 approaches a value of 0.9 (limited by the in-process inventory constraint) and workstation 2 becomes the bottleneck. This in turn causes u_1 to decrease.

A similar argument holds when $\lambda=1$. By applying the Kuhn-Tucker conditions to (4.19) it is shown that

$$\left[\frac{\mu_1}{\mu_2}\right]^{1/2} \leq (\mu_1-R_1)/(\mu_2 - \frac{2}{3}R_1) \qquad (4.22)$$

where R_1 is the solution of 4.20 for $\lambda=1$

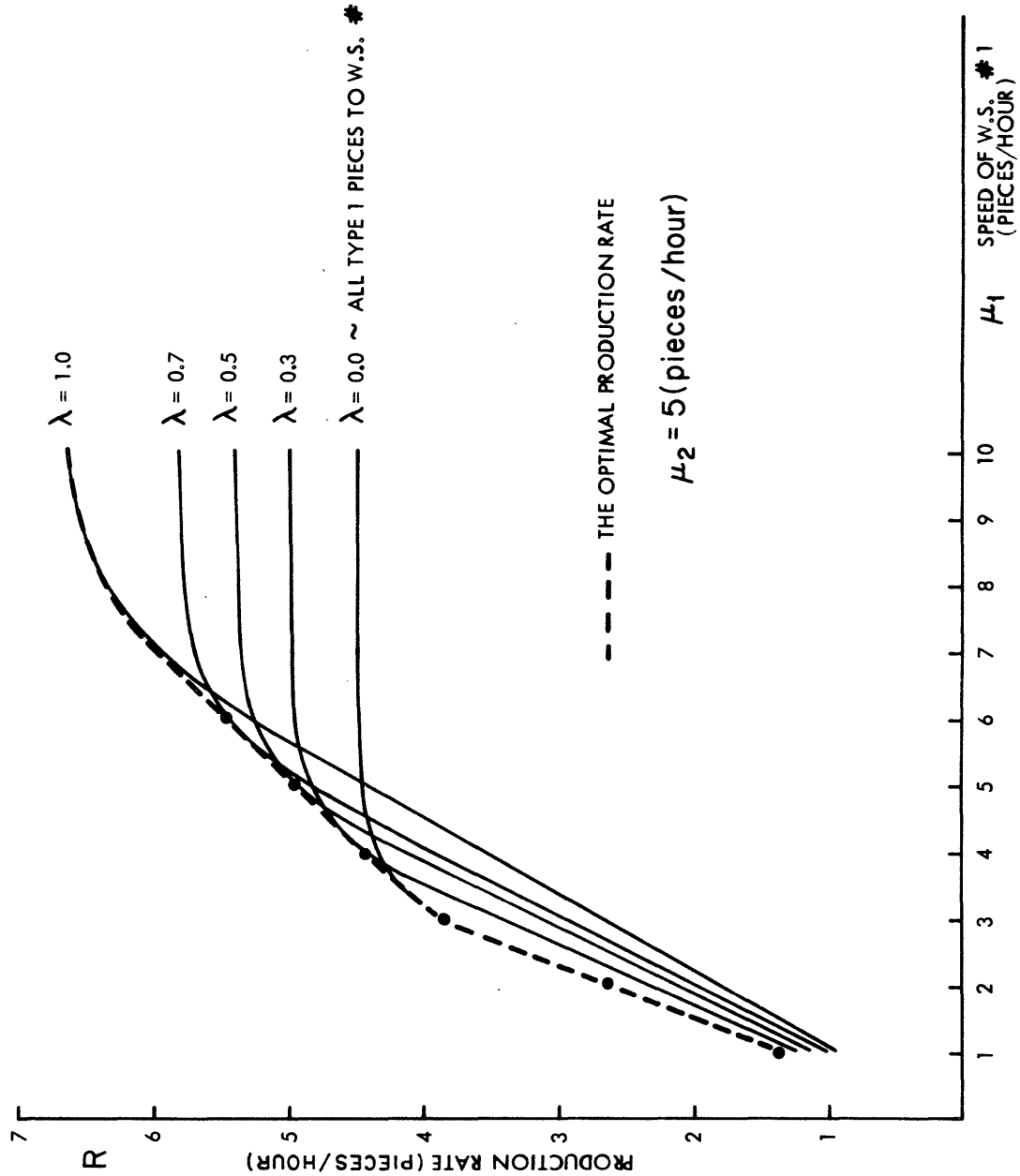


Fig. 4.12. Production Rate as a Function of μ_1 with λ as Parameter

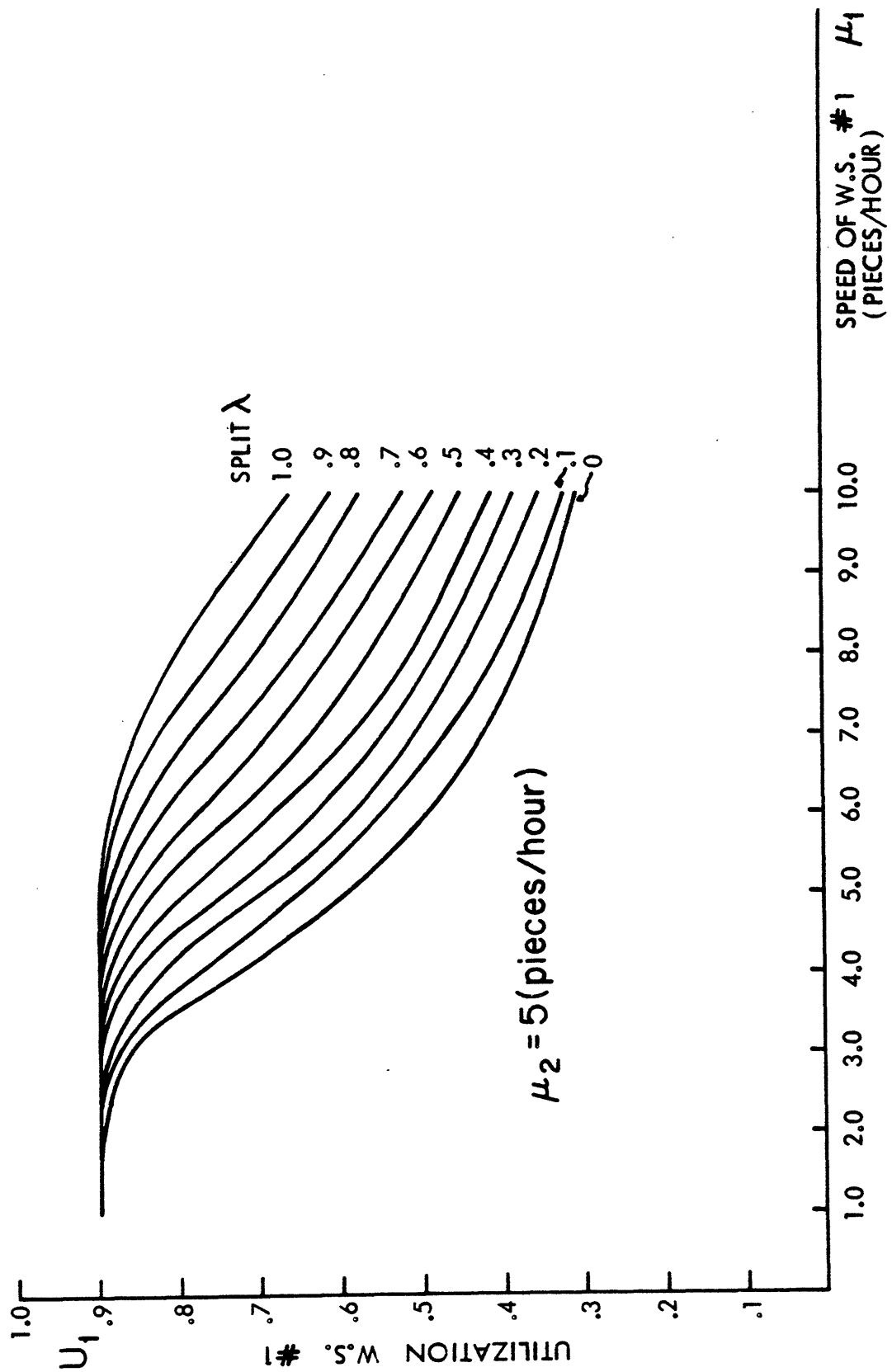


Fig. 4.13. Utilization of Workstation 1 as a Function of μ_1 with λ as Parameter

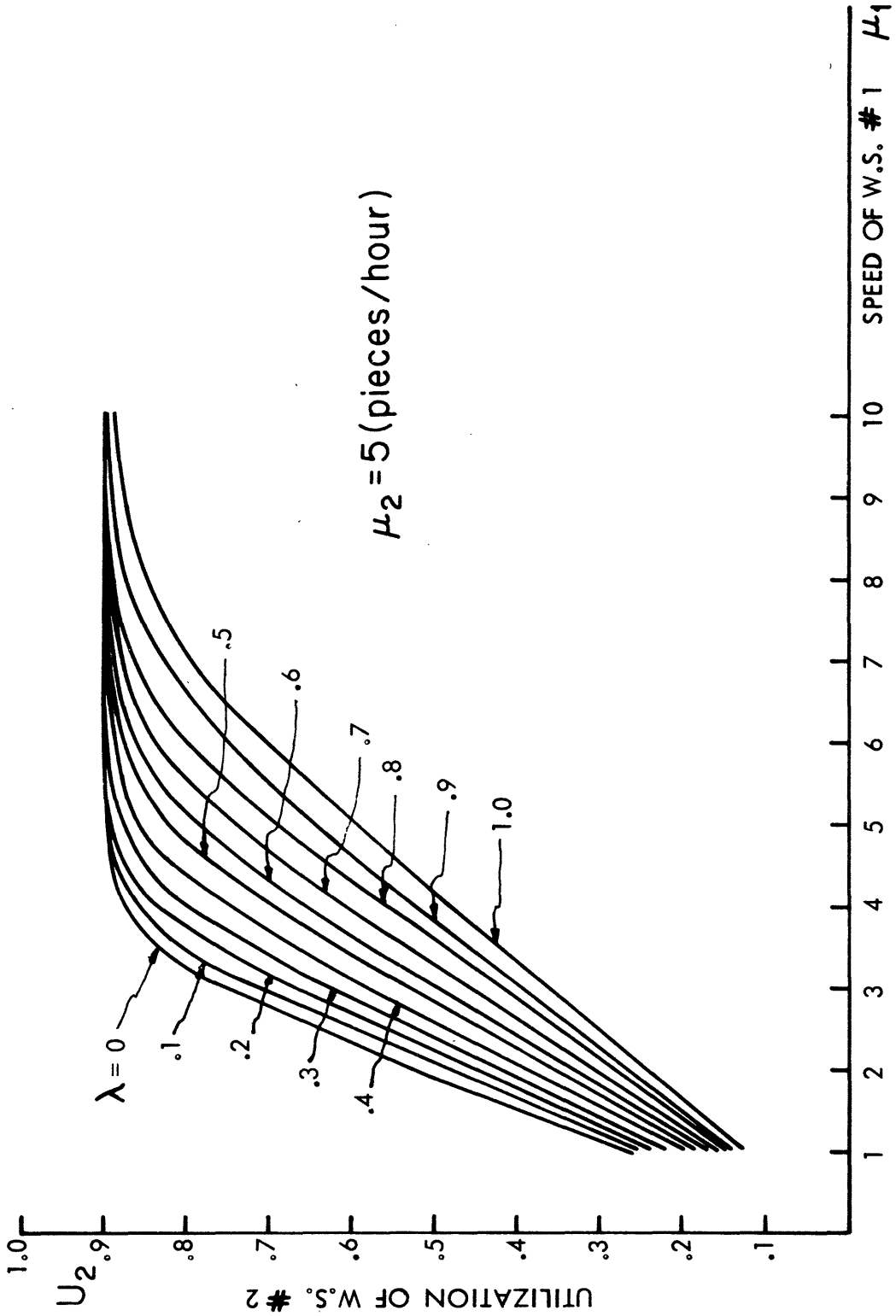


Fig. 4.14. Utilization of Workstation 2 as a Function of μ_1 with λ as Parameter

In this region, all type 1 pieces go to workstation 1. As its speed increases, its utilization drops. When $\mu_1 > \mu_2$, station 2 is the bottleneck and it determines the maximum production rate.

From (4.19), if $0 < \lambda < 1$, then $\pi_2 = \pi_3 = 0$. The Kuhn-Tucker optimality conditions are, in this case:

$$\frac{\partial L}{\partial R} = \frac{\partial L}{\partial \lambda} = 0 \quad . \tag{4.23}$$

Some algebraic manipulation reveals that

$$u_1 = 1 - \frac{1}{Q}\left[1 + \left(\frac{\mu_1}{\mu_2}\right)\right] \quad . \tag{4.24}$$

Differentiating with respect to μ_1

$$\frac{du_1}{d\mu_1} = \frac{1}{\mu_1 Q}\left(\frac{\mu_1}{\mu_2}\right)^{\frac{1}{2}} > 0 \quad . \tag{4.25}$$

Thus, in this region the utilization of workstation 1 increases with μ_1. A similar argument shows that u_2 drops as μ_1 increases.

There are two changes taking place as μ_1 increases. The production rate R and the split λ are both increasing. The rising production rate tends to increase the utilization of both stations. The change in λ means that type 1 pieces are being switched from station 2 to station 1. It appears that the changing split has the effect of raising the throughput of station 1 at a faster rate than the increase on μ_1. At station 2, the effect is to actually reduce the throughput of the station, thereby lowering the utilization. The gradient (4.25) varies inversely with Q, and when Q is large, is close to zero. The graph of Fig. 4.15 shows the utilization of both stations for $Q = \infty$. For $3.3 < \mu_1 < 7.5$, the utilizations of both stations are 1, as predicted by (4.24) and (4.25).

The non-monotonic shapes of the curves in Fig. 4.10 are determined by the solution of NLP 4.2. One might ask whether or not this kind of behavior results only from the model or reflects phenomena which can be observed in actual systems.

The optimization method suggested by Secco-Suardo (1978) makes use of the exact solution of the closed network-of-queues model. For the two-workstation system, the problem may be stated as (see the Appendix):

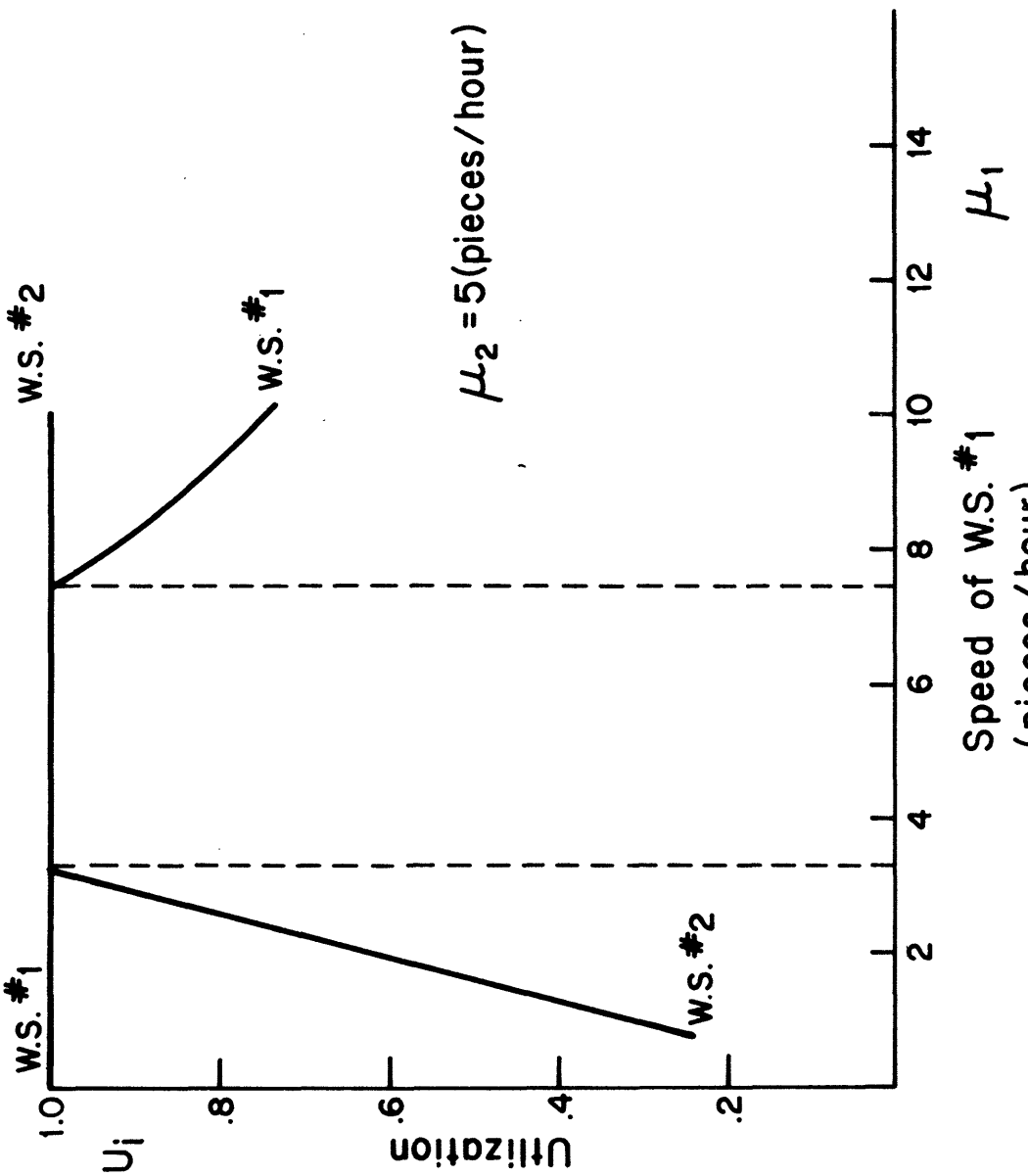


Fig. 4.15. Optimal Workstation Utilization as a Function of μ_1, with $Q=\infty$

NLP 4.3

$$\underset{\lambda}{\text{Maximize}} \quad f(\lambda) \tag{4.26}$$

subject to

$$0 \leq \lambda \leq 1 \tag{4.27}$$

where $f(\lambda) = G(M,N-1)/G(M,N)$ and is the ratio that determines the throughput of the network. It is defined by equation (2.19). If the Kuhn-Tucker optimality conditions are applied, three conditions are again found to govern the optimal choice of λ:

$$\frac{df(\lambda)}{d\lambda} \begin{cases} < 0 & \lambda = 0 \\ = 0 & 0 < \lambda < 1 \\ > 0 & \lambda = 1 \end{cases} \tag{4.28}$$

The solution of (4.28) determines for what range of μ_1 it is best to have a mix of strategies for type 1 pieces.

Intuitively, at either boundary, $\lambda = 0$ or $\lambda = 1$, the behavior of the two models should be the same. As the speed of workstation 1 increases for fixed λ, the production rate grows until station 2 becomes the bottleneck. As a result, the utilization of station 1 drops while that of station 2 increases. This is indeed the case, as is shown in the Appendix.

In the interior of the constraint set $0 < \lambda < 1$, behavior is determined by the solution of (4.28). The defining relationship for $G(M,N-1)/G(M,N)$ indicates that this involves finding the roots of a high-order polynomial.

In the Appendix, the behavior of the closed network model for the two-workstation system is investigated and is found to be remarkably similar to that of an open network model with a constraint on the average level of in-process inventory.

The graphs of Figs. 4.12 and A.1 show the effect of varying the speed μ_1 of workstation 1 on the production rate R for different values of the split λ. Both models exhibit the approximately linear growth in R when μ_1 is small, followed by a saturation effect. The closed network-of-queues model predicts a higher throughput than the open network model. For $\lambda = 0$, for example, the asymptotic throughput for the former is five pieces per hour, compared to 4.5 pieces for the latter. Similarly, the

utilization μ_1 of workstation 1 is higher for the closed network model (Figs. A.3, 4.10 and 4.13).

The effect of λ on the production rate for fixed values of μ_1 is similar for both models, as is shown in Figs. A.2 and 4.16. The maximum throughput for any μ_1 occurs at about the same value of λ for both cases.

The two models apply under different assumptions. The open network model looks at the system from the point of view of the workpieces. The average queue length in an open system grows very rapidly as the arrival rate at the server approaches the service rate. This effectively limits the throughput of the bottleneck station and hence the production rate when there is a constraint on the in-process inventory.

In the closed queueing network model, the number of customers in the system is fixed. This model can be viewed as modelling the system from the point of view of the fixed number of pallets circulating in the system. The utilization of the bottleneck station can approach unity if the number of pallets is large enough. It is this fact that explains the higher throughput and utilization in the closed network-of-queues model.

The variation of the throughput of the two-station system with the relative speed of the two stations and the split should be investigated by means of a simulation. It can then be seen whether or not the counter-intuitive behavior of the system model reflects a phenomenon that occurs in actual systems, or is in fact only a property of the mathematical models.

The sensitivity of the production rate to changes in the split can be judged from Figs. 4.12 and 4.16. Figure 4.12 shows the effect of μ_1 on the production rate with the split as a parameter and $Q = 10$, and the optimal production rate in Fig. 4.17 is seen to be the envelope of the curves in Fig. 4.12. The three operating regions can be seen. For $3.3 \leq \mu_1 \leq 7.5$, the optimal split is sensitive to the relative speeds μ_1 and μ_2 of the two workstations. The peaks of the curves in Fig. 4.16 are fairly flat, which indicates that small variations in λ about the optimal do not reduce the production rate greatly.

The graphs of Fig. 4.17 showing the effect of μ_1 on the production rate and Fig. 4.10 on station utilizations, emphasize the importance of analyzing a flexible manufacturing system as an interconnected system. The results here illustrate the effect of changing two system parameters; the speed of workstation 1 and the split. In an actual system, many more

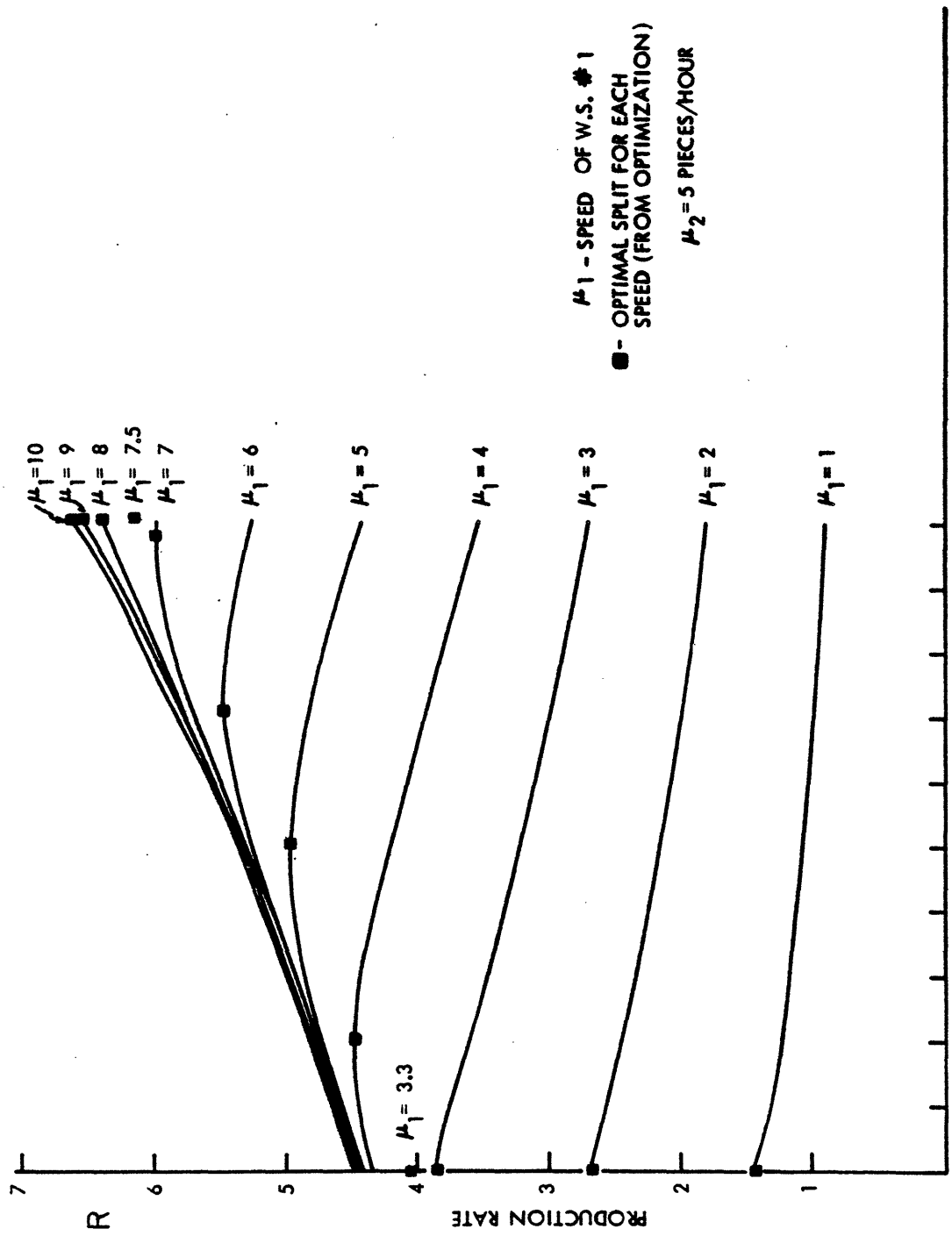


Fig. 4.16. Production Rate as a Function of λ with μ_1 as Parameter

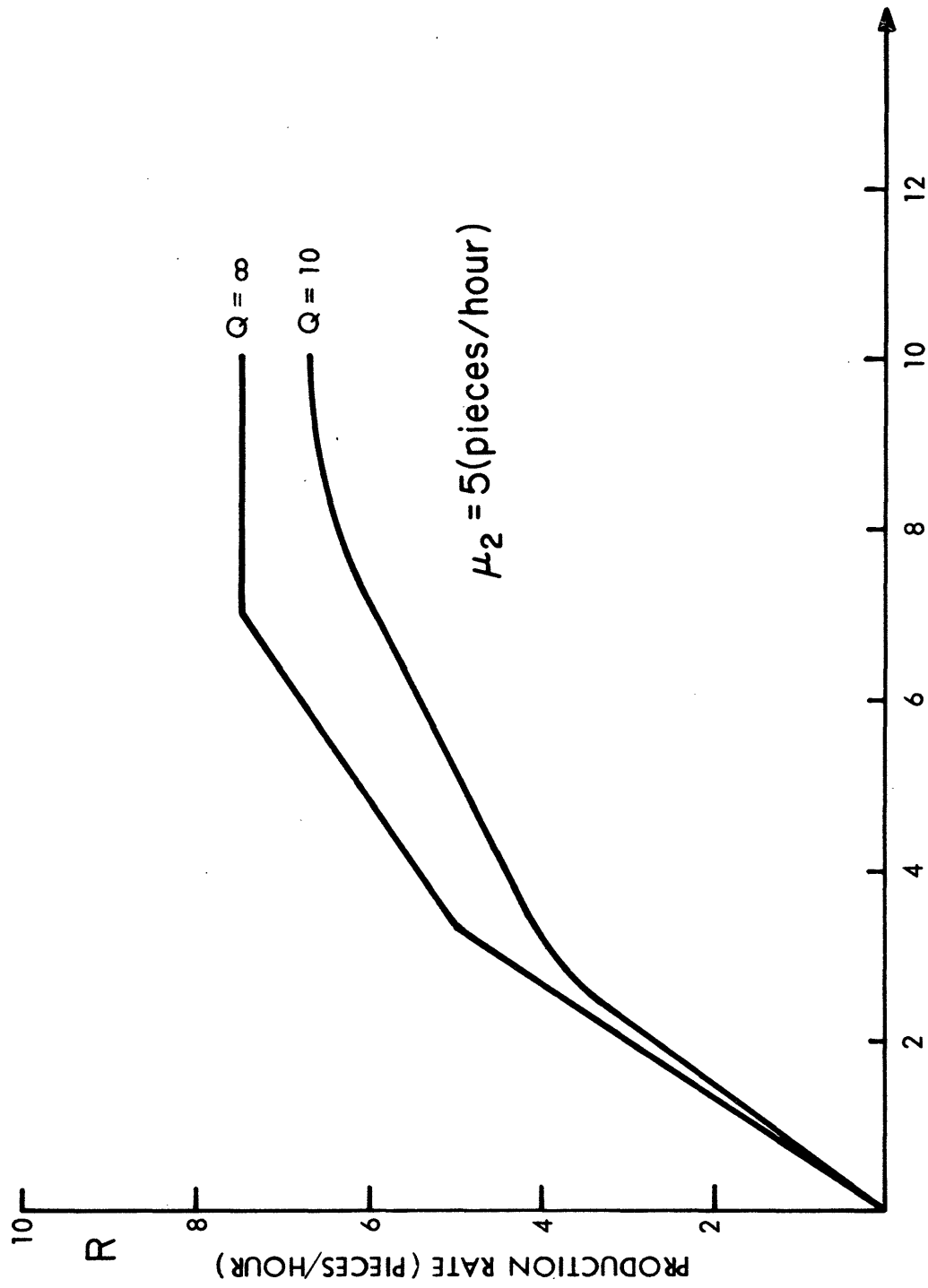


Fig. 4.17. Optimal Production Rates as Functions of μ_1

parameters can be varied, and a trade-off between the many conflicting requirements is needed before an optimum choice of parameter values can be made. An analytic technique such as the flow optimization method for evaluating the performance of a given system is an essential tool in the planning and operation of a flexible manufacturing system.

If two different systems are being compared, it is essential to choose the correct split for each, otherwise the comparison would not yield the correct result. For example, assume that in the design of the two-machine system, two versions of workstation 1 are available. The cheaper version works at a rate $\mu_1 = 4$ pieces per hour and the more expensive works at $\mu_1 = 6$ pieces per hour. Using a fixed value of $\lambda = 0.2$ would show the faster workstation producing only a 6% improvement over the slower machine. However, using the optimal values of λ ($\lambda = 0.19$ for $\mu_1 = 4$ and $\lambda = 0.75$ for $\mu_1 = 6$) shows the true improvement to be 22%. If installing the faster workstations costs 10% more than the slower, the wrong decision would be made if the comparison were made with $\lambda = 0.2$ for both versions.

4.3 Results for a Four-Workstation Deterministic System

4.3.1 Five-Part Example with Strategies Enumerated in Advance

The system of Fig. 4.18 has four workstations, and is simulated as a discrete step process on a digital computer (Horev et al., 1978). There are five different types of pieces to be manufactured.

In operating the simulation, the policy was to give each piece two alternative paths through the system. The first priority route is the preferred one, and the second is available if for some reason the first cannot be used.

The first and second priority routes can be viewed as strategies in the flow-optimization approach. The strategies, two for each piece, are shown in Table 4.3. All machining times are deterministic and the system is assumed never to fail. Using only the first priority route for each piece, the relative workload on each workstation is given in Table 4.4. The system does not have equal loads at the workstations and is said to be "unbalanced" (Ward, 1980). The maximum production rate of the system operated in this manner is easily calculated. Under this policy workstation

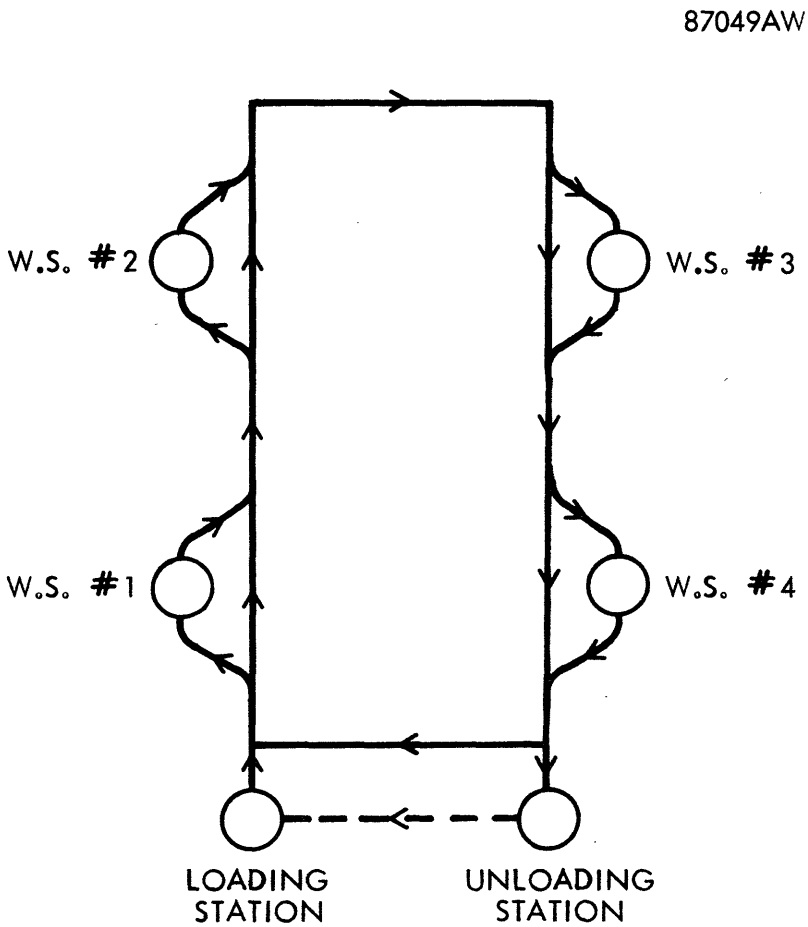


Fig. 4.18. A 4-Workstation System

part no.	ratio requirement	Optimal split % to follow 1st route	ROUTES AND SERVICE TIMES: 1st priority route	ROUTES AND SERVICE TIMES: 2nd priority route
1	0.20	1.00	L → 1 → 4 → U 6 10	L → 2 → 3 → U 10 10
2	0.20	0.99	L → 2 → 3 → 1 → U 5 4 7	L → 1 → 4 → U 9 10
3	0.12	0.93	L → 3 → 2 → U 9 10	L → 1 → U 26
4	0.08	1.00	L → 2 → 4 → 1 → U 7 10 4	L → 4 → 3 → U 1 14
5	0.40	0.60	L → 3 → U 7	L → 1 → 2 → 4 → U 2 5 4

Table 4.3

Strategies and Optimal Splits for 4-Machine Case with 5 Part Types

	First Priority Route Only		Optimal Splits	
	Predicted Loading	Utilization (simulation) interval (0-1500)	Predicted loading	Utilization (simulation) interval (0-1500)
Workstation				
1	.62	.646	1.00	.972
2	.58	.576	1.00	.965
3	1.00	.967	1.00	.963
4	.60	.578	1.00	.965

Table 4.4 - Predicted and Simulation Utilization Using lst Priority Routes only and Using Optimal Splits

	First Priority Routes Only		Optimal Splits	
	Predicted	Simulation	Predicted	Simulation
production in 1500 time steps	320	302	433	406
improvement			35%	34%

Table 4.5 - Predicted and Actual Production in 1500 Time Step Interval

3 is fully utilized. Let R be the total production rate. Pieces 2, 3, and 5 use station 3; their respective ratio requirements are 0.2, 0.12, and 0.4. The time τ_i each piece spends at workstation 3 is given in Table 4.3. The production rate R thus satisfies:

$$R(\alpha_2 \tau_2 + \alpha_3 \tau_3 + \alpha_5 \tau_5) = 1 \qquad (4.28)$$

where α_i is the ratio requirement for a type i piece. Using the values shown, R = 0.2137 pieces per unit time.

The optimization method of Section 2.3.2 can be applied to this case to determine the optimal proportion of each type of piece to follow the first priority route. There are 10 strategies. Using the variables y_ℓ ($\ell = 1,\ldots,10$) to indicate the flow rate into the network of pieces following strategy ℓ, the linear program LP 2.1 applies.

The optimal split for each type of piece is shown in Table 4.3. The results are intuitively satisfying. Workstation 3 is the bottleneck station when only the first priority routes are used. The production rate is increased by using the alternative path for a proportion of the pieces that require station 3 on the first priority route. This is most evident for type 5 pieces, for which 40% are diverted to the second path. As a result, the workstations are balanced with equal loads at each station.

The total production in an interval of 1500 time steps is predicted by the optimization to be 433 pices (Table 4.5), which is a 35% improvement over the output calculated from (4.28) using first priority routes only.

4.3.2 A Scheduling Procedure for the Loading Station

A major problem to be solved before flexible manufacturing systems can be used to their full potential is the tactical problem of deciding precisely when a particular piece should be loaded into the system. In order to examine the effect of implementing the strategies suggested by the optimization, a simple loading strategy has been improvised. The variable $\bar{y}_\ell$ is the optimal flow rate of strategy ℓ pieces through the network. To achieve that flow rate, a strategy ℓ piece should be loaded into the system every $\bar{t}_\ell = 1/\bar{y}_\ell$ time units on the average. If a piece is to be loaded at its appointed time, and the loading station is busy,

it is put into a queue at the loading station. Since the loading operation is generally much faster than the operations at the workstations, the utilization of the loading station is low, and consequently the proportion of time it is idle is high. Thus, on average, the number of pieces that cannot be loaded immediately when they are required is small.

This loading strategy was tested on the discrete simulation using the first priority routes only and the optimal split for the strategies. The results are summarized in Table 4.4. Using the first priority routes only, 302 pieces were produced in the simulation interval of 1500 time units. The utilization of the workstations is very close to the predicted levels. When the optimal strategy mix is implemented, 406 pieces are produced in the same interval, an improvement of 34%. The workstations are balanced with utilization close to unity. The production rate predicted by the optimization result and the simulation production rate are within 6% of each other, using first priority routes only and 7% with optimal splits. The 35% improvement in production rate predicted by the optimization is achieved to within 1% when the optimal splits are used in the discrete simulation.

There are two factors which account for the differences between the predicted and simulation results. First, the simulation interval (0,1500) covered an initial startup period when there were no pieces in the system. This had the effect of lowering the average utilizations of the workstations and the production. Second, congestion effects in the transportation system, which are not included in the linear model, have an adverse effect on the performance of the system.

The simulation results of Table 4.4 were achieved with a large number of pallets (10 for each piece strategy) and at least 10 queueing (buffer) spaces at each workstation. The actual production rate, when the number of pallets is limited, may be approximated by the method of Section 2.4. If the asymptotic production rate is R, the true production rate is approximately

$$P = \frac{N}{N + M_B - 1} R \qquad (4.29)$$

where N is the number of pallets and M_B is the number of stations with equally high relative utilizations. For the four-machine system, the approximation is compared to the network-of-queues solution (Ward, 1980) for optimal and non-optimal strategies in Fig. 4.19. The approximation is closer to the network-of-queues result as N, the number of pallets, grows. The transportation system in the network-of-queues model is treated as a multi-server station with a service time proportional to the time that workpieces spend on the conveyor. The relative utilization of the transportation system therefore drops as its speed is increased. The approximate solution in Fig. 4.19 is closer to the network of queues solution for the higher line (transportation system) speed, illustrating the point that the accuracy of the method is highest when the relative utilization of the non-bottleneck workstation is low compared to that of the bottleneck stations.

The occupancy of the queueing space at each of the workstations is shown in Fig. 4.20, for both the first-priority routes only, and with the optimal splits applied. For each station, the proportion of time P(n) during the simulation interval when there were n pieces in the workstation buffer is plotted as a function of n.

For the case in which the workstations are not balanced, two buffer positions is the maximum requirement. This occurs at workstation 2, where two pieces were waiting for service for 12% of the simulation interval. At all other stations, buffer occupancy was never more than one piece. The buffers are empty more than 80% of the time at stations 3 and 4, and 78% at station 1.

For the balanced case using optimal splits, the buffer requirements change drastically only at workstation 1. In this case, up to seven buffer positions are required. Despite the fact that all stations have the same utilization, the use of the buffers is quite different. This might perhaps be explained by looking at the strategy diagrams of Table 4.3. Station 1 is required by five of the strategies for a relatively short time period. However, one of the strategies (part 3 on the second route) requires an operation at station 1 which takes 26 time units. This relatively long

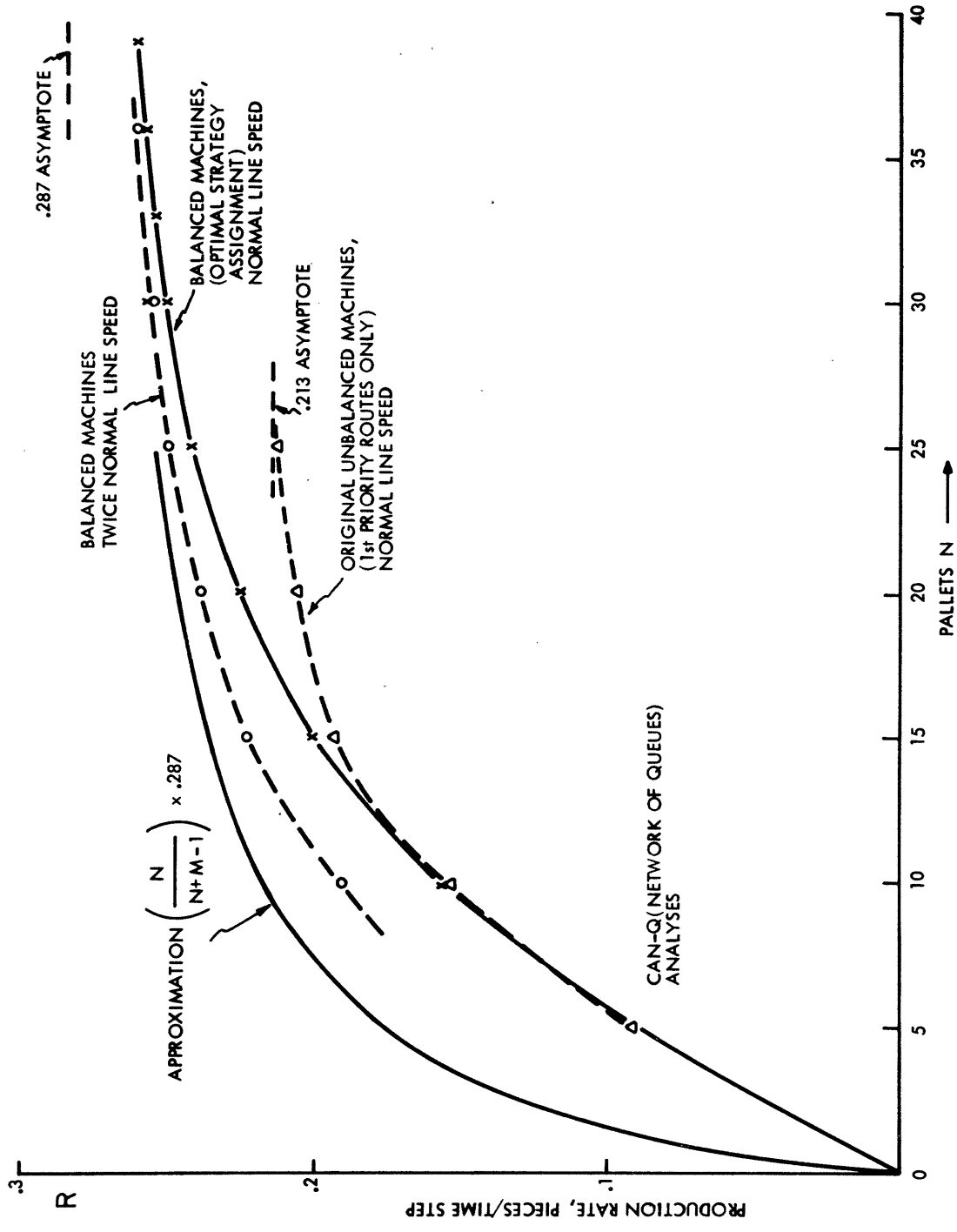


Fig. 4.19. Production Rates for 4-Machine 5-Piece Example as a Function of the Number of Pallets N

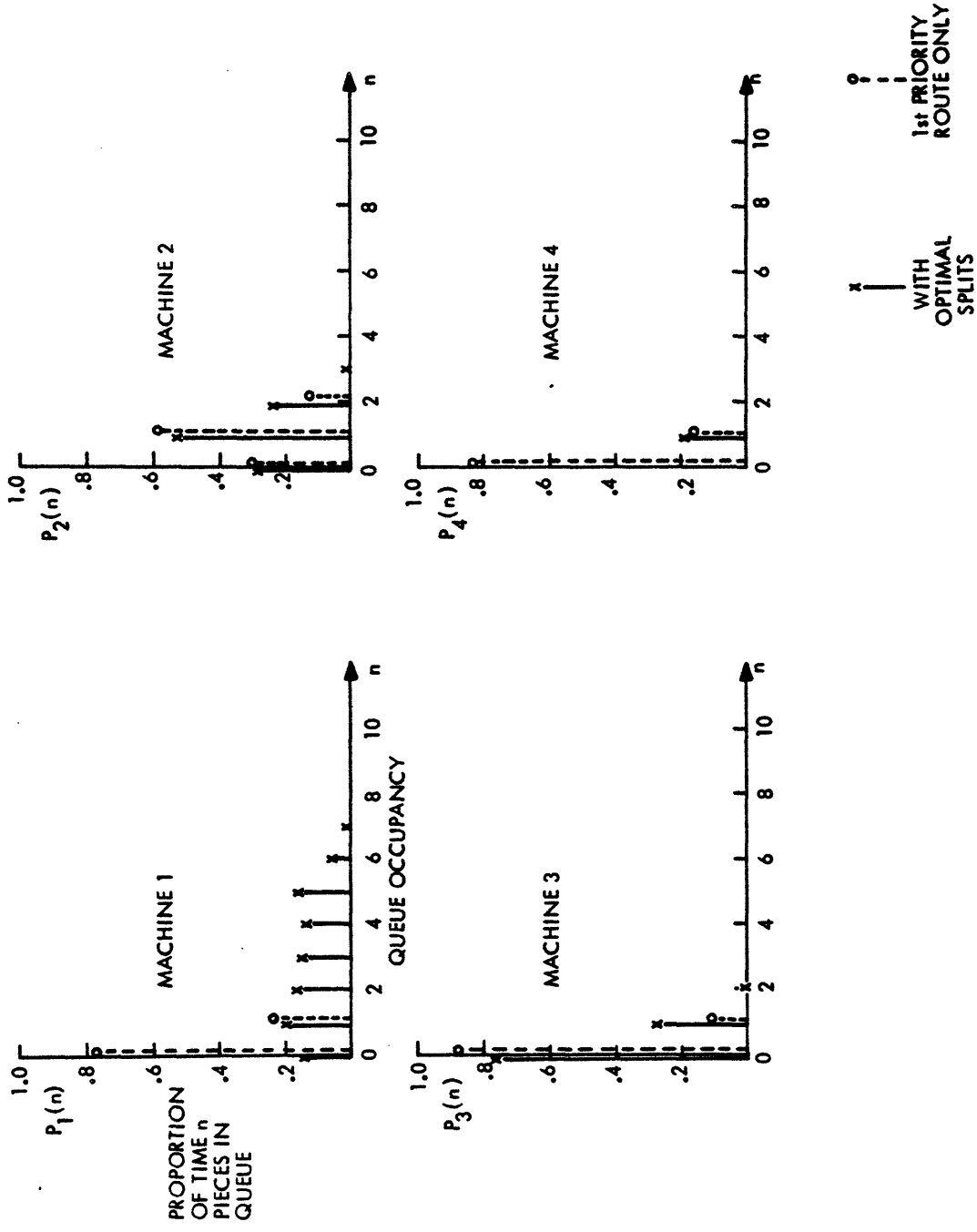


Fig. 4.20. Queue Occupation for 4-Machine 5-Piece System

operation is a likely cause of the large buffer requirement at station 1. The utilization of the other buffers is not different from the unbalanced case, except that there were three pieces waiting at workstation 2 in one instance.

The graphs of Fig. 4.20 show the limitations of the simple loading strategy. In order for it to be effective, there must be adequate waiting room at the stations. This might not present too great a problem for a system producing small, inexpensive pieces, but where the cost of providing buffer spaces and additional pallets is considerable, an improved loading strategy would offer a substantial advantage. This point is further discussed in Section 5.3.2.

4.3.3 Six-Part Examples: Strategies Not Enumerated in Advance

The linear program LP 2.1 is suitable for cases in which the number of strategies is too large to be enumerated in advance. The processing time matrices T_i are shown in Table 4.6 for six parts to be manufactured in the four-workstation system of Fig. 4.18. This is an example of an extremely flexible system; most of the operations can be performed at any machine. If, in the example, the operation numbers k were to denote strict precedence constraints, there would be 207 possible strategies. The number becomes much bigger if the precedence constraints are relaxed.

The formulation of LP 2.1 produces a linear program with 56 x_{ij}^k variables. The number of constraints is 19, of which four are inequality constraints and 15 are equality constraints due to flow conservation and ratio requirement constraints.

The problem was solved for two different product ratio requirements by a commercial linear programming code.[1] In this example, the strategies could be easily identified from the optimal solution $\hat{x}_{ij}^k$ and are given in Tables 4.7 and 4.8. In general, the $\hat{x}_{ij}^k$ variables do not produce unique strategy assignments. For a type 2 part with the ratio requirement of Table 4.7, the optimal flow rates through the workstations are given in Table 4.9. The strategy diagram for this part type can thus be drawn as in Fig. 4.21. Several combinations of the four possible strategies can result from the flow rates of Table 4.9. The three strategies shown in Table 4.7 are one

---

1 International Mathematical and Statistical Library (IMSL) on IBM (VM) 370/168.

k-operation	1	2	3	4
1	0.95	1.0	1.1	1.0
2	5.9	4.8	5.6	6.0
3	3.5	3.7	3.2	∞

t^k_{1j} - part type 1

op-k	1	2	3	4
1	3.0	3.4	3.2	3.9
2	2.0	2.8	1.9	2.8

t^k_{6j} - part type 6

k-operation \ j-workstation	1	2	3	4
1	3.9	4.2	4.1	3.6
2	2.9	2.7	2.3	2.5
3	1.0	0.9	1.1	1.1
4	5.4	6.1	5.6	6.0

t^k_{2j} - part type 2

k-operation \ j-workstation	1	2	3	4
1	∞	3.0	3.2	3.1
2	4.3	4.4	4.3	4.6

t^k_{3j} - part type 3

k-operation \ j-workstation	1	2	3	4
1	1.4	∞	1.4	1.6

t^k_{4j} - part type 4

operation \ j-workstation	1	2	3	4
1	2.0	2.9	∞	3.0
2	3.7	3.9	3.0	3.9
3	4.9	5.9	5.0	5.9

t^k_{5j} - part type 5

Table 4.6 - t^k_{ij} Matrices and Operational Requirements for 6 Part Example

part	1	2	3	4	5	6
ratio requirement	0.2	0.3	0.1	0.1	0.1	0.2

Results

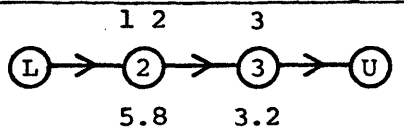
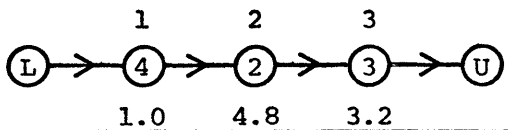
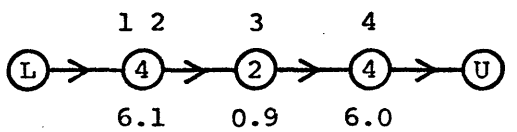
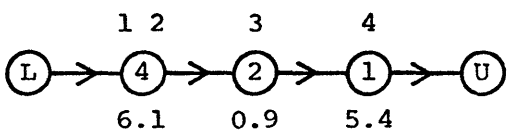
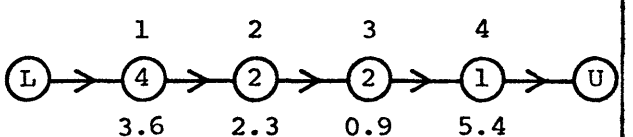
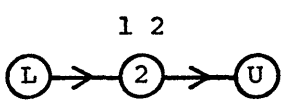
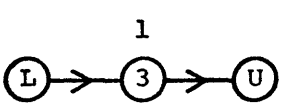
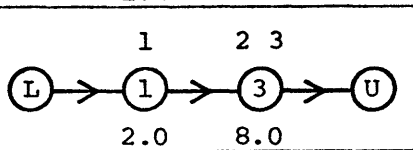
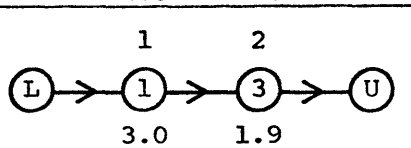
Part	strategies (ℓ,m — (j) — τ_{ij} : Operations, Machine, Total time)	flow rate pieces/min	split
1		.06225	.65
		.03295	.35
2		.02813	.20
		.0856	.60
		.02913	.20
3		.04760	1.0
4		.04760	1.0
5		.04760	1.0
6		.09520	1.0
		Production rate .4760	

Table 4.7 - Example 1 - Optimal-Strategy Assignments

part	1	2	3	4	5	6
ratio requirement	.1	.05	.25	.3	.1	.2

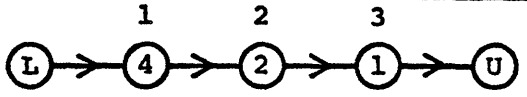
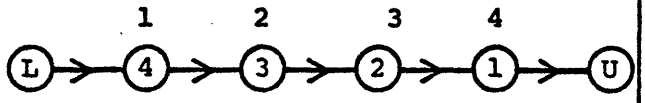
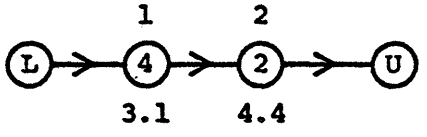
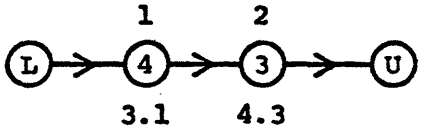
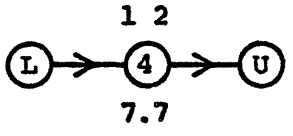
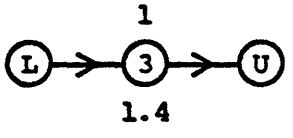
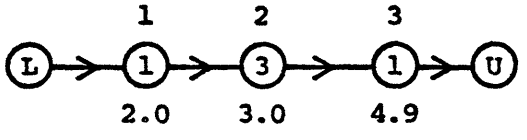
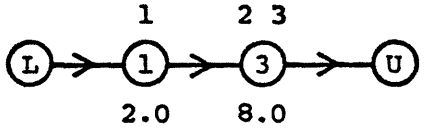
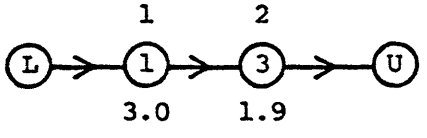
part	strategies (ℓ,m → (j) → Operations; τ_{ij} → Machine · Total time)	pieces/min.	split
1		.06564	1.0
2		.03282	1.0
3		.1490	.76
		.0033	.02
		.04472	1.0
4		.1641	1.0
5		.06075	.93
		.0049	.07
6		.1313	
	Production rate	.6564	

Table 4.8 - Example 2 — Optimal-Strategy Assignments

k operation \ j machine	1	2	3	4
1	0.0	0.0	0.0	0.1428
2	0.0	0.0	0.02913	0.1137
3	0.0	0.1428	0.0	0.0
4	0.1147	0.0	0.0	0.02813

Table 4.9 - Optimal Flowrates x_{ij}^{k} for Type 2 Workpiece, (Six-Part Problem Example 1)

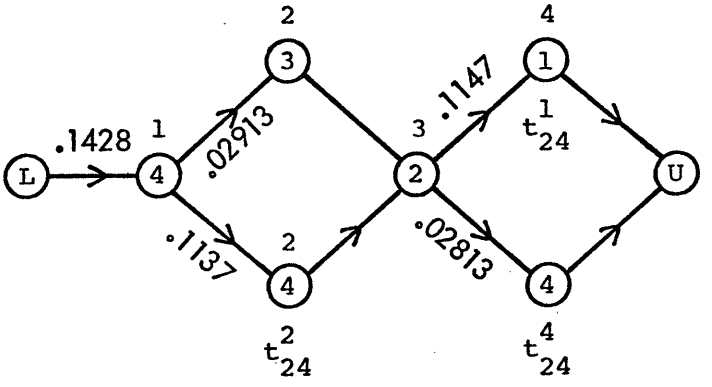


Figure 4.21 Strategy Diagram for Type 2 Workpiece
Six-Part Problem Example 1

such combination which can realize the optimal flow rates.

The fact that different strategy assignments can result in the same optimal flow rates through the workstations means that it is not necessary to assign workpieces to strategies at the loading station. A central controller (or perhaps local controllers) could be left to decide on the location at which the next operation is to be done when a workpiece leaves a workstation. The objective in this case would be to maintain the optimal flow rates $\hat{x}^k_{ij}$. For a type-2 workpiece in this example (Fig. 4.21), on completion of operation 3 at workstation 2, the controller would have a choice of performing operation 4 at either station 1 or 4. If a workpiece is assigned to a strategy at the loading station, there would be no further choices, since each strategy precisely defines the sequence of workstation visits. The controller's task is then the simple one of keeping each piece on its assigned strategy.

The optimal assignments produce balanced workloads at the workstations for both production ratio requirements. It is interesting to note, however, that the ratio requirement of Table 4.8 produces a production rate 38% higher than that of Table 4.7.

In both cases, the pieces with the high ratio requirements are assigned to more than one strategy, whereas the other pieces, in general, seem to be assigned to a single strategy. This is a sensible policy. Suppose, for example, the piece with the highest ratio requirement were to be assigned to a single strategy. The production rate of this piece would then be limited to the service rate of the slowest machine on that route. Because the proportion of each piece in the output is specified, the production rate of the pieces with the smaller ratio requirements is also limited by the same machine. It is likely that the flow rate of these pieces into the system would then be unable to utilize fully the remaining workstations. This would not only lead to an unbalanced system, but would give a production rate below that which the system can attain with an optimal assignment.

The optimal assignments were implemented on the discrete time simulation using the loading strategy discussed in Section 4.3.2. The workstation utilizations and the proportion of time P(n) that there were n pieces in

each queue are illustrated in Figs. 4.22 and 4.23 for the simulation interval of 1500 time steps. The maximum number of pieces at any workstation did not exceed five for either case.

Also shown in Figs. 4.22 and 4.23 are the predicted and actual production of pieces in 1500 time steps, and the percentage differences. In example 1, the optimization predicts performance within 4%. For example 2, the error in the prediction of the production is 9%. Note also that the utilization of workstations 2 and 3 is lower: 0.8791 and 0.8753, respectively. The initial transient period and congestion may partly account for this. The simple loading strategy could also be partly at fault because it loads pieces at predetermined time instants without taking into consideration the conditions prevailing within the system. This may have the effect of increasing congestion on the transportation network.

4.4 Conclusion

The modelling techniques of Chapter 2 have been applied to two- and four-workstation examples, and the results compared to discrete simulations under the same conditions for the latter case. In the two-machine system, stochastic machining times were considered and the performance of the system was evaluated for different parameter values. The optimization results are intuitively pleasing. The optimal strategy assignments in this case produce approximately equal workloads at the workstations, providing that the service rates at the two workstations are not too widely different.

The linear model for deterministic systems was applied to a four-workstation simulated system. The optimization gave a balanced system with an improved production rate. In two 6-part examples, where strategies were not enumerated in advance, the formulation of LP 2.1 proved to be effective in providing enough information to assign strategies to the workpieces. This is a saving in computation, because a search over hundreds of possible strategies was not necessary.

The optimal strategy assignments were implemented on the discrete simulation using a simple loading strategy devised to utilize the optimal flow rates. The production rates and workstation utilizations obtained were close to the values predicted by the optimization results.

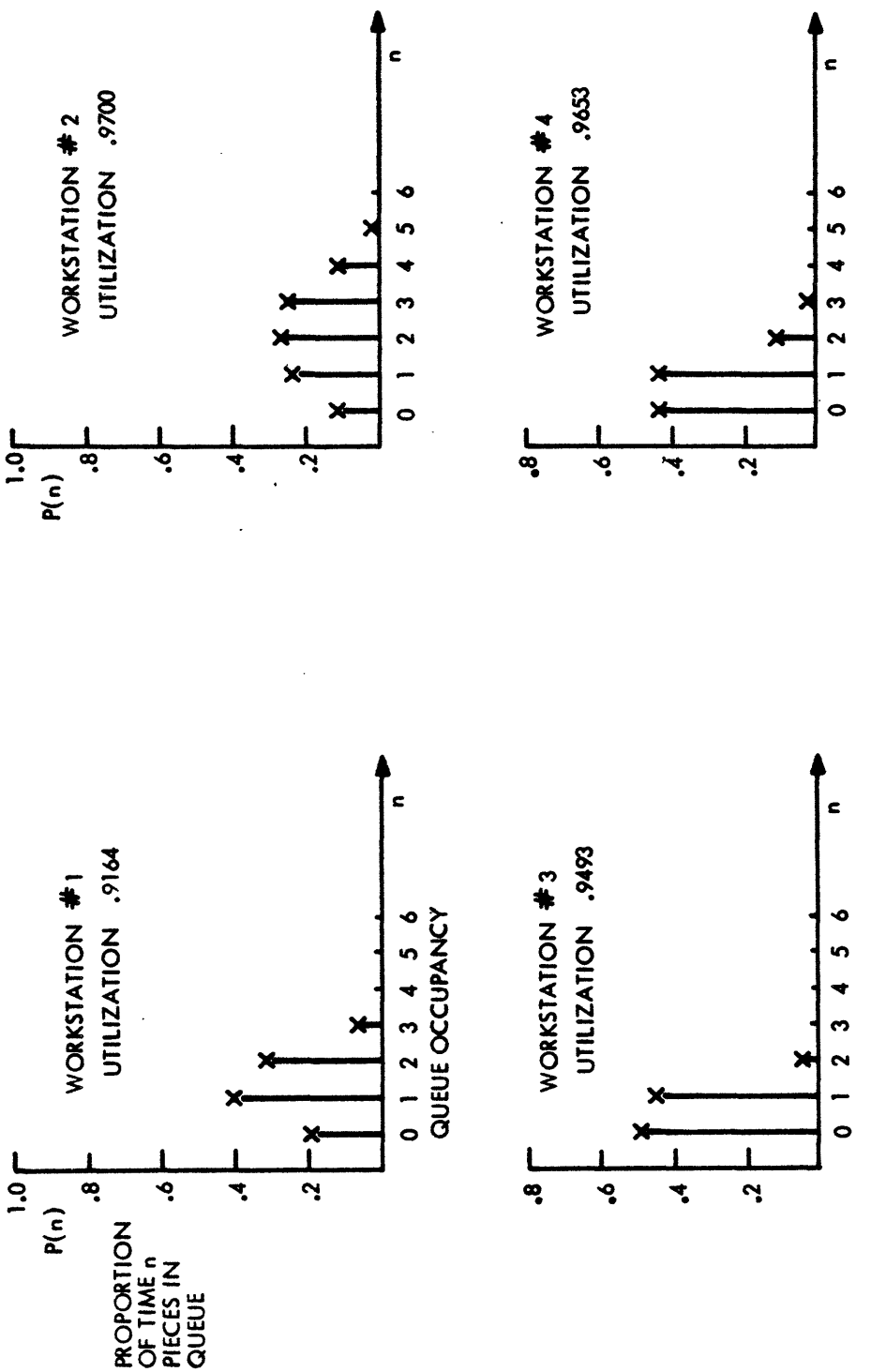


Fig. 4.22. Queue Occupation for 4-Machine 6-Piece System Example 1

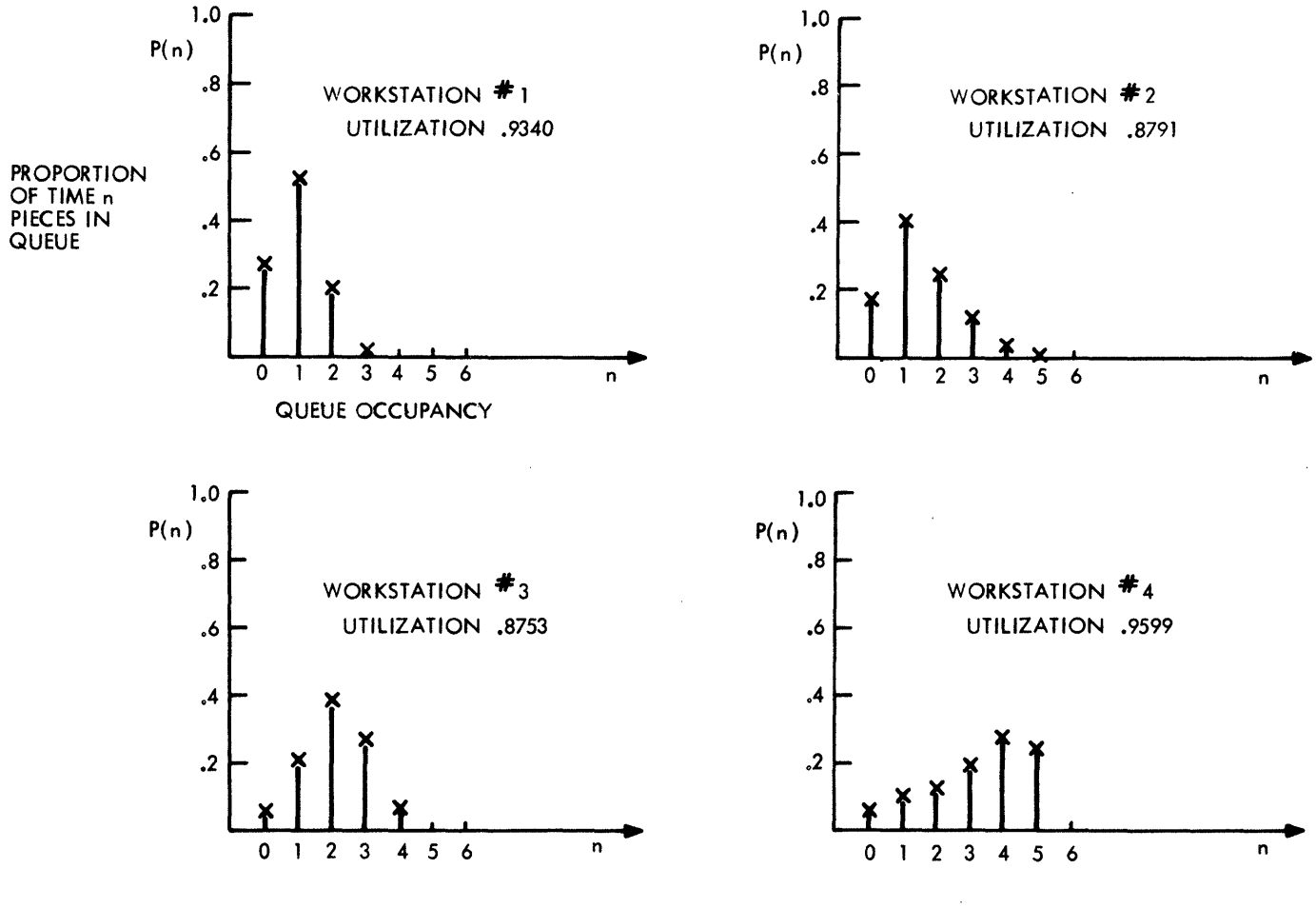


Fig. 4.23. Queue Occupation for 4-Machine 6-Piece System Example 2

The network flow optimization approach has shown itself to be a quick and efficient method of carrying out trade-off studies on a flexible manufacturing system and of choosing optimal strategy assignements. Some further development is necessary before the method can be applied to more general systems. The results presented show that it is a method which can be implemented as part of the control system in flexible manufacturing systems.

5. OPEN AREAS FOR FUTURE RESEARCH

5.1 Introduction

There are a number of problems remaining before the network flow optimization approach can be applied to more general systems. Two related areas discussed in Section 5.2 are concerned with failures of workstations and limited queueing or buffer spaces. The possible roles for the network flow optimization method in solving strategic and tactical problems are examined in Section 5.3.

In operating a flexible manufacturing system, decisions have to be made at various levels depending on the length of time scale involved (Hutchinson, 1977). These levels may be divided for convenience into two categories. At the tactical level, the moment-by-moment functioning of the system is of interest. The state of the workstations, the position of individual pieces and the stage they have reached in their manufacturing process are the kinds of variables that a tactical level controller would monitor and act upon.

Above the tactical level is the strategic decision making level. It is very broad in scope and covers aspects ranging from the planning of production and the configuration of the system to the allocation of work-pieces to strategies. The time scale involved ranges from a few hours to perhaps several months.

The flow optimization method appears promising as a component of decision making schemes at both the strategic and tactical levels; but ways of handling additional effects need to be devised.

5.2 Reliability and Limited Capacity Constraints in Flexible Manufacturing Systems

The network flow analysis described in Chapter 2 assumes that each queue in the system has infinite capacity in the case of an open system, and sufficient space to hold all of the pieces in the case of a closed system. It is further assumed that all workstations are reliable and never fail. Reliability and the capacity of the buffer or queueing space at each workstation are closely related issues (Schick and Gershwin, 1978). Buffers are put at workstations so as to smooth the production in a system subject to

breakdowns. A buffer can prevent a workstation from becoming idle immediately when another station which supplies it with some part has broken down.

Failures in a flexible manufacturing system are of two kinds. Minor breakdowns occur when, for example, a tool breaks and has to be replaced, or when the pallet handler misaligns a pallet and an operator has to be called to align the pallet properly. A workstation in a flexible manufacturing system is a complex device and is subject to many types of failure. The time between failures and the time to repair can thus be modelled as stochastic processes. The exponential distribution, because of its memoryless property, is a good model when failures of all types are considered (Schick and Gershwin, 1978). Other distributions can be used to model cases where for example a system is more likely to fail if it has been in operation for a long time (Sivazlian and Stanfel, 1975). Major failures result in a workstation or the transportation system being out of service for a lengthy period of time, and can also include scheduled down time for maintenance.

Methods of handling failures will depend on their severity, i.e., whether they are minor or major. One way of handling minor failures as described above is to incorporate them as a stochastic component of station processing times. The non-linear programming formulation of Chapter 2 can then be applied.

Major workstation failures require a different approach, since it is unreasonable to incorporate them in stochastic processing times. The best way seems to be a temporary reconfiguration of the operating strategy. A flexible system need not be stopped due to a single workstation failure. If the system configuration is such that there is no operation which can only be performed at a single station, production can continue at a reduced rate using the remaining stations. New optimal strategies would have to be evaluated using the network flow optimization approach. It is also possible to work out contingency plans in advance so that when a particular workstation fails, the optimal operating strategies using only the remaining stations are available for immediate implementation.

When a failure occurs, it may not be immediately known whether it is a minor or a major problem. Improved failure identification procedures are needed. Research is being conducted, for example, into automatic tool wear sensors (Cook and Subramanian, 1977) and using such monitoring devices it might be possible to at least estimate the repair time. Maintenance personnel could also enter their estimates of the repair time when called to a failed machine. The decision rule as to whether or not to switch to a new strategy will require a careful analysis of the causes of workstation failures and the length of time it takes to diagnose and repair the fault.

Finite buffer spaces affect the performance of a production line. A particular problem in a flexible manufacturing system is that if a part attempts to enter a workstation at which the buffer is full, it is rejected and it has to remain on the transportation system. It is important that effects due to finite capacity constraints should be modelled and taken into account in computing optimal strategies.

The analysis of unreliable transfer lines with finite buffer spaces leads to a large system of simultaneous equations which have to be solved (Schick and Gershwin, 1978). The equations have a special structure which can be exploited in order to produce an efficient solution procedure. It is likely however that direct analysis using Markov process (Kleinrock, 1975) techniques will prove untenable for a flexible manufacturing system.

Lavenberg (1975) has studied the stability and maximum throughput of open networks with finite capacity constraints. For a network such as that in Figure 5.1, there is a certain maximum arrival rate λ below which the network is stable in the sense that the steady state average number of customers in the system remains finite. The maximum throughput for such a network is λ, provided that the interarrival times have probability distribution functions with rational Laplace transforms. In all but simple cases, the calculation of λ is very difficult (Lavenberg, 1975).

The behavior of queues with finite capacity constraints may be usefully approximated by a diffusion process. Gelenbe (1975) considers a diffusion process $x(t)$ on the interval $(0,M)$. The assumption is made that at either boundary, $x(t)=0$ or $x(t)=M$, the process remains at the boundary for an

87049AW035

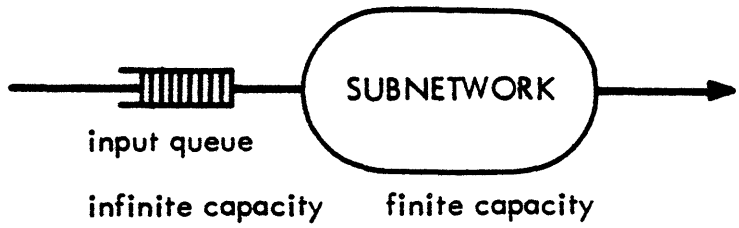


Fig. 5.1. Model of a Queueing System with Capacity Constraints

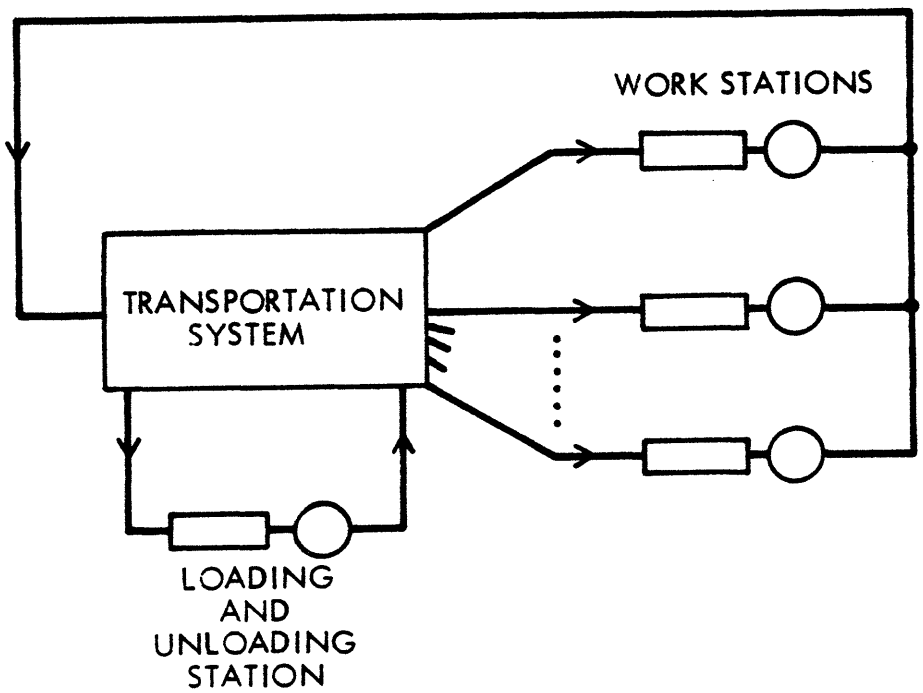


Fig. 5.2. Closed Network of Queues Model for a Flexible Manufacturing System

exponentially distributed time interval before jumping into the interior. The exponential distribution assumption could be relaxed by assuming a distribution with rational Laplace transforms. This suggests a method of approximating a network of queues with capacity constraints. A vector valued diffusion process $\underline{x}(t)$ which is constrained so that $0 \; x_i(t) < m_i$, where m_i is the buffer capacity at workstation i, can be considered (Kobayashi, 1974). Alternatively each queue could be assumed to behave in the network as it does when it is in isolation.

In this manner an approximation to two important quantities can be obtained. The first is the probability that a workstation and its associated queue are empty. From this quantity, the utilization of the workstation can be evaluated and the bottleneck stations can be identified. The second is the probability that the buffer is full. The proportion of pieces arriving at the workstation that are rejected is related to this probability. Thus the additional traffic on the transportation system due to station rejection can be evaluated.

Conceptually, the modelling of closed queueing systems with a limited capacity at each station should not differ significantly from the unlimited capacity case. A valid assumption in modelling a flexible manufacturing systems is that pieces enter the transportation network when they leave a workstation. The model of Figure 5.2 (Solberg, 1977) is then applicable. The effect of limiting the capacity at each workstation is to reduce the number of states in the Markov chain model of the system without altering the interstate transition structure of the Markov chain. It remains to be seen whether a product form solution exists (Gordon and Newell, 1967), (Baskett et al.,1975) for the transition balance equation, or if not, whether a sum-of-products form is appropriate (Gershwin and Berman, 1978), (Gershwin and Schick, 1979).

An important question to be answered is to what extent do limited buffer sizes affect the optimal mix of strategies? A simulation study of a flexible

manufacturing system is needed to investigate this issue. If the optimal mix is insensitive to changes in buffer size, then it could be found assuming that infinite buffer spaces are available. The maximum production rate would then be found by solving the limited capacity model with known strategies. A saving in computation results because the complicated limited capacity model is not solved repeatedly.

5.3 Application of Network Flow Optimization to Strategic and Tactical Problems

Production planning and inventory control are problems which have had considerable attention (Lee, 1978). The advent of flexible manufacturing systems capable of producing several different products simultaneously is expected to have a considerable impact on this area of production management (McRainey, 1977).

Traditionally, the production manager is faced with the problem of scheduling a number of products through a manufacturing facility so as to satisfy a forecast demand which might not be perfectly known, while maintaining a certain level of in-process inventory (Sivazlian and Stanfel 1975). A considerable cost is incurred in changing from the production of one product to the production of another.

A number of methods have been applied to this problem. Gorenstein (1970) finds economic lot sizes for tire production in a eight week period by a linear programming method. In his case the set-up time for the molds is substantial, and furthermore, each mold can only produce one type of tire at a time. Linear and integer programs have been used for multi-product scheduling in chemical plants (Eilon, 1969) (Royce, 1970). Each product is produced in batches and the reactors are not capable of handling more than one product at a time.

A hierarchical approach to production scheduling has been suggested (Bradley et al., 1977) (Gabbay, 1975). At the highest level, an aggregated plan is made over a relatively long horizon, taking into account factors such as estimated demand patterns and costs which are usually the concern of top management. Decisions made at the highest level act as constraints on middle and shop floor management. This approach is motivated by the desire to avoid large production planning problems which result if all factors are

included in one problem. Furthermore, where future demands are not accurately known, the detailed model is solved with parameters which may fluctuate after the plan is established, leading to a poor utilization of resources (Gabbay, 1975).

The hierarchical planning procedure is suited to flexible manufacturing systems (Hutchinson, 1977). At the lowest levels, however, management is by computer control with a minimum of human intervention. As a result, great attention will have to be paid to the flow of information between and within the various hierarchies.

The flow optimization technique finds the mix of operating strategies that maximize the production rate or any other performance index, given a system configuration and parts specifications. As such it could be incorporated into decision making schemes which involve flexible manufacturing systems.

Although it is possible to manufacture several kinds of products simultaneously, there will be cases where it is impracticable to maintain machine tooling for all of the required types at the same time. One is then faced with the problem of scheduling subsets of the part types to manufacture in rotation, each for n shifts of a given month, for example, so that a certain production requirement is met and inventory is maintained at desired levels. The flow optimization method can be used as a component of a scheme which searches over possible part combinations in order to find an optimal production plan.

Typically the decision variables in a planning problem are $x_i(j)$, the number of type i units to be produced during period j of the planning horizon. The period could be a shift of eight hours and the horizon a week, for example. To establish what the actual production would be in period j with the assignment $\hat{x}_i(j)$, the flow optimization could be solved. The ratio requirement constraint for type i piece would be

$$\alpha_i(j) = \frac{x_i(j)}{\sum_i x_i(j)} \tag{5.1}$$

The output of the flow optimization is then $R_i(j)$, the production rate of type i pieces in period j. Thus if the period has a length of time T, the actual number of pieces produced is $\hat{x}_i(j) = TR_i(j)$. The total production over the entire horizon is then given by $\sum_i \sum_j \hat{x}_i(j)$.

Starting with an initial assignment, the total production could be evaluated by solving the flow optimization problem for each period. A new assignment would then be evaluated by calculating, for example, the gradient of the objective function and finding a feasible direction which improves on the value of the objective. The process would be repeated until an optimal point is reached. The change over from one product mix to another might have to consider the set-up costs involved in changing tools and control programs. If set-up costs are negligible and the future demand is known perfectly, the problem may be formulated as a deterministic, discrete time optimal control problem. In the face of uncertain demands, it becomes a stochastic problem.

A problem unique to flexible manufacturing systems is that of configuring a system for a given parts mix, i.e., which workstations to tool for which part types. Each workstation in a flexible system has a limited tool magazine capacity. The problem is, given all the manufacturing requirements of the pieces, how should the configuration of operational capabilities be chosen so as to attain a maximum production rate while at the same time maintaining enough flexibility in the system so that the system is relatively immune to failures. The optimal solution $\underline{x}$ to the flow optimization problem may include certain x_{ij}^k variables which are zero. In this case, operation k on a type i piece is not carried out at workstation j. This indicates therefore that the necessary tools should not be loaded at that station. An extension of the flow optimization method can be made so as to include tool capacity constraints and set-up costs. There would also have to be constraints to ensure that there is enough flexibility in the system to guard against workstation failures. A mixed integer programming problem is likely to result.

The operational problem is concerned with the instant-to-instant control of a flexible manufacturing system. At this level, such things as the precise loading schedules, the location of pieces in the system, and their

next operation are monitored, similar to the control of a job shop. The general job shop problem is in a class of problems termed NP-complete (Kanellakis, 1978) which are extremely difficult to solve. Heuristic algorithms which exploit the special features of a flexible system will therefore have to be developed.

The loading strategy described in Chapter 4 uses the strategy flow variables y_ℓ to determine loading intervals for each type of piece. When tested in a discrete simulation, the method achieved the predicted high utilization rates at the workstations. However it needs fairly generous buffer sizes at the workstations. This might be a problem in a system producing large heavy pieces. The loading strategy may be improved upon. Under the simple procedure, a type i piece should be loaded at the time instants $t_i + n\zeta_i$ (n=0,1,2,...,), where t_i is the initial loading time and ζ_i is the interval evaluated from the optimal flow rate. However, so long as a type i piece is loaded within the interval $(t_i + n\zeta_i, t_i + (n+1)\zeta_i)$, the average flow rate can be maintained. Thus the precise instant within the interval at which the piece should be loaded could be evaluated from data about the state of the system, thereby realizing a closed-loop control policy.

A periodic scheduling technique described in Section 2.3.2 (Hitz, 1979) finds a schedule for the minimum integer number of parts satisfying the ratio requirement. The schedule is required to leave no idle time at the bottleneck workstation and to be such it can be repeated without idle time. In order to generate strategies and identify the bottlenecks in the system, a preliminary step might be to apply the flow optimization algorithm to the problem.

The scheduling problem is important, particularly if the system is subject to disturbances. Minor random failures and other uncertainties which delay workpieces during their passage through the system preclude detailed schedules over long time horizons. A good scheduling policy for this kind of system is one which quickly attains the maximum production rate starting from some initial condition. It should be flexible enough to accommodate

random disturbances such as tool failure and blockages. This means that an effective policy will most probably be closed-loop control. Because of the computational complexity of scheduling problems (Kanellakis, 1978), heuristic algorithms are necessary. This is an aspect of flexible manufacturing systems that requires a considerable amount of further research.

5.4. Summary of Open Areas

The problem of modelling of flexible manufacturing systems with finite buffers at the workstations has been discussed. This is a difficult problem to treat analytically and approximate techniques should be developed. Simulation studies should be made so as to gain an understanding of the effect of limited buffer sizes on the optimal strategy mix.

The effect of flexible manufacturing systems on production management and inventory control has also been discussed as an open area for investigation. The problem of how the best configuration of operational capabilities in a flexible system should be chosen may be answered by the flow optimization approach. Integer variables and additional constraints and cost terms will have to be included in the problem formulation for practical application.

Algorithms for the real-time control of a manufacturing system are an important element in the management hierarchy, and particularly important for flexible manufacturing systems. The optimal flow rates may provide a good starting point in evaluating loading strategies which achieve high production rate while keeping the required buffer capacities small.

6. CONCLUSION AND SUMMARY

The analysis of the movement of individual pieces through a manufacturing system leads to combinatorial problems which are known to be difficult to solve. This report has presented a network flow optimization approach to the problem of choosing the best mix of operating strategies in a flexible manufacturing system. An operating strategy is a sequence of operations required to manufacture a workpiece and defines a path through the system. All possible routes do not have to be identified in advance. The solution method of Chapter 3 generates the strategies for each type of piece as part of the solution. The optimal proportion of each type of piece to be routed along each path is provided by the algorithm. Only a subset of all possible strategies need to be considered in order to arrive at the optimal combination.

Systems in which the machining times are non-deterministic give rise to non-linear programs because of the build up of queues at the workstations. Deterministic machining times and arrival processes result in linear programs. The asymptotic maximum production rates of systems where the processing times have general probability distributions are found by identical linear programs.

The non-linear programs are solved by the augmented Lagrangian algorithm which adjoins the nonlinear constraints to the objective function to form a Lagrangian function. The Lagrangian is minimized subject to linear constraints by considering convex combinations of the extreme points of the feasible flow set. A decomposition method which results in a set of strategy generating subproblems, each involving only one type of piece, is used to generate the extreme points as they are needed. This reduces the computational requirement because the nonlinear optimization is carried out with fewer variables than the original problem.

Numerical results presented in Chapter 4 are intuitively pleasing. For a two-workstation system, choosing the routing so that the utilizations of the two workstations are equal, or nearly so, is optimal when the difference in the speeds of the two workstations is not great. However, when the speed difference is large, the optimal assignment does not produce equal loads at the two stations. The optimal mix of strategies is thus found to be sensitive to the relative speeds of the two workstations.

The linear programming formulation gave good predictions for the performance of a discrete simulation of a four-workstation system. For the parts specification used, the maximum production rate was given by a route assignment with equal loads at all four machines. A simple loading strategy devised to produce the optimal flow rates into the system resulted in high utilization of the workstations when applied to the discrete simulation.

A number of problems remain before the network flow optimization approach presented here can be applied to more general systems. In calculating the average in-process inventory, the assumption was made that there are infinite buffer spaces at each workstation. In general this is not the case. Analytic methods for dealing with finite buffers in flexible manufacturing systems are needed.

The issue of reliability was raised in Chapter 5. The best course of action in the event of a workstation failure will depend on a number of factors, including the expected time to repair the machine. Decision rules will have to be developed so that the system controller can decide which strategies to use when a station drops out of service. It is clear also that in choosing the best operating strategies, the reliability of the workstations should be taken into account.

The optimization calculates the best mix of operating strategies given a set of part specifications and a system configuration. Production planning in an organization with a flexible manufacturing system involves choosing both the numbers of different types of pieces to be made during a certain period of time, and the configuration of operational capabilities within the system. The network flow optimization method appears to be a promising component of a scheme to tackle such a problem.

APPENDIX

The Closed Network of Queues Optimization Model Applied to the Two-Workstation System

In Section 2.4, it is shown that

$$G(M,N) = \sum_{s} \prod_{i=1}^{M} x_i^{n_i} \tag{A.1}$$

where x_i is the relative utilization of station i. The production rate of a flexible manufacturing system modelled as a closed queueing network is given by [Secco-Suardo, 1978]

$$\frac{G(M,N-1)}{G(M,N)} x_L \tag{A.2}$$

where x_L is the relative utilization of the loading station.

It can be shown that

$$G(M,N) = \sum_{i=1}^{M} A_i x_i^N \tag{A.3}$$

where

$$A_i = \prod_{\substack{j=1 \\ j \neq i}}^{M} \left(1 - \frac{x_j}{x_i} \right)^{-1}$$

provided that each workstation can be modelled as a single server in the network model.

If the relative utilizations for the two-workstations are scaled so that x_i is the arrival rate at station i due to a unit throughput, the production rate can be written, using (A.3), as

$$P = \frac{G(M,N-1)}{G(M,N)} = \frac{x_1^N - x_2^N}{x_1^{N+1} - x_2^{N+1}} \tag{A.4}$$

The split for type 1 pieces (the proportion going to workstation 1) is λ. From (4.12) and (4.13), the relative utilizations of the two workstations when there is a flow rate of 1 piece per hour is

$$x_1 = \frac{1}{3\mu_1}(2+\lambda) \tag{A.5}$$

$$x_2 = \frac{1}{3\mu_2}(3-\lambda) \tag{A.6}$$

The problem NLP 4.2 follows direction from (A.4). The reader is referred to (Secco-Suardo, 1978), (Ward, 1980) and (Solberg, 1971) for a complete discussion of the evaluation of G(M,N) using generating functions.

The function (A.4) is plotted in Fig. A.1 as a function of μ_1 for $\mu_2 = 5$, $N = 10$, and with λ as a parameter. This should be compared to Fig. 4.12 where the production rate for the open network model is plotted under the same conditions. In Fig. A.2, the production rate as a function of λ with μ_1 as parameter is shown. A comparison should be made between Figs. A.2 and 4.16. Figure A.3 shows the variation of the utilization of workstation 1 with μ_1 for two values of λ. The value of λ which gives the highest production rate for each value of μ_1 can be determined from Fig. A.2. The utilization of station 1 when the optimal λ is used is superimposed on Fig. A.3. A comparison between Fig. A.3 and Figs. 4.10 and 4.13 can be made. A discussion of the similarities and differences of the open and closed network models is given in Section 4.2.

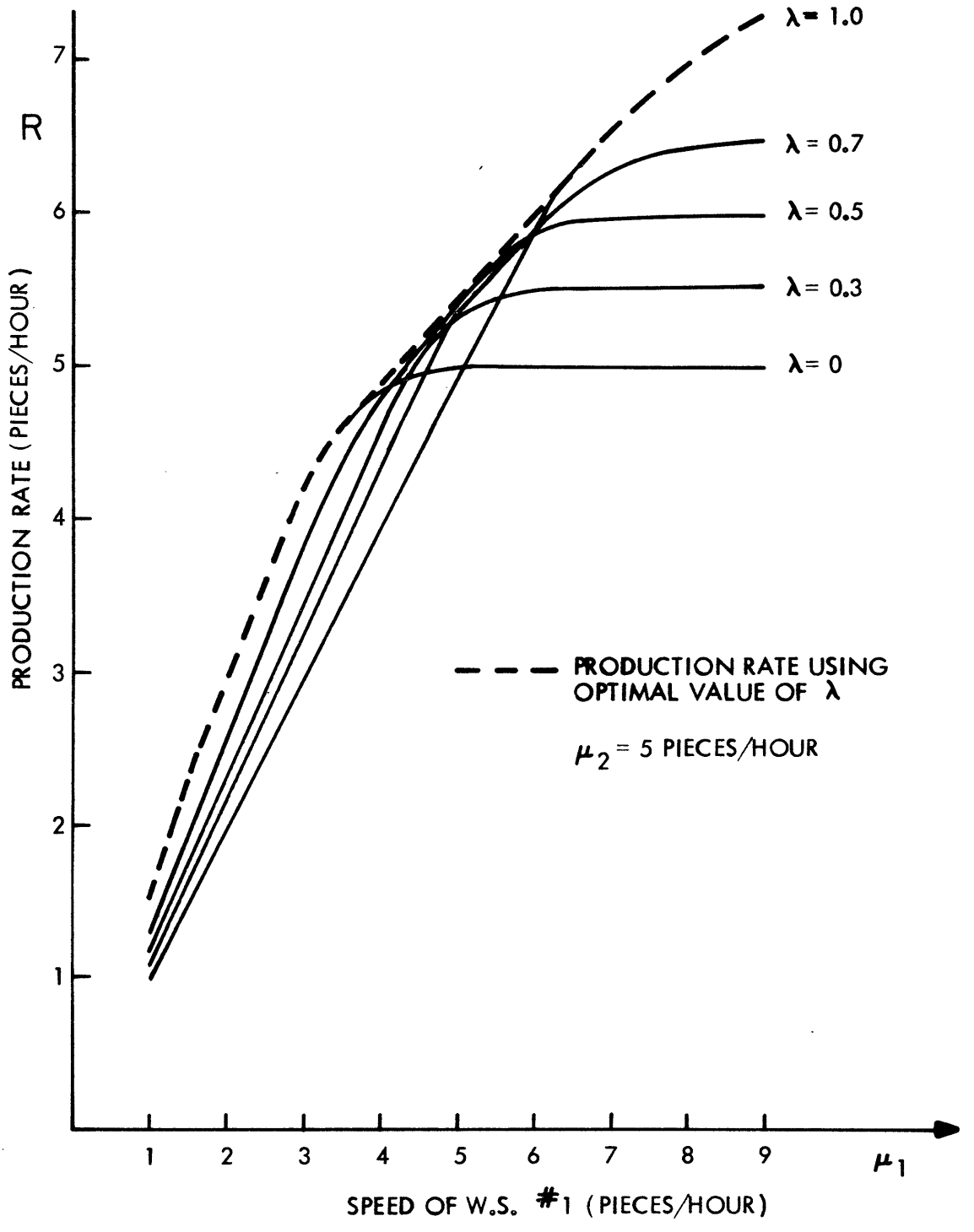


Fig. A.1. Production Rate as a Function of μ_1 with λ as the Parameter

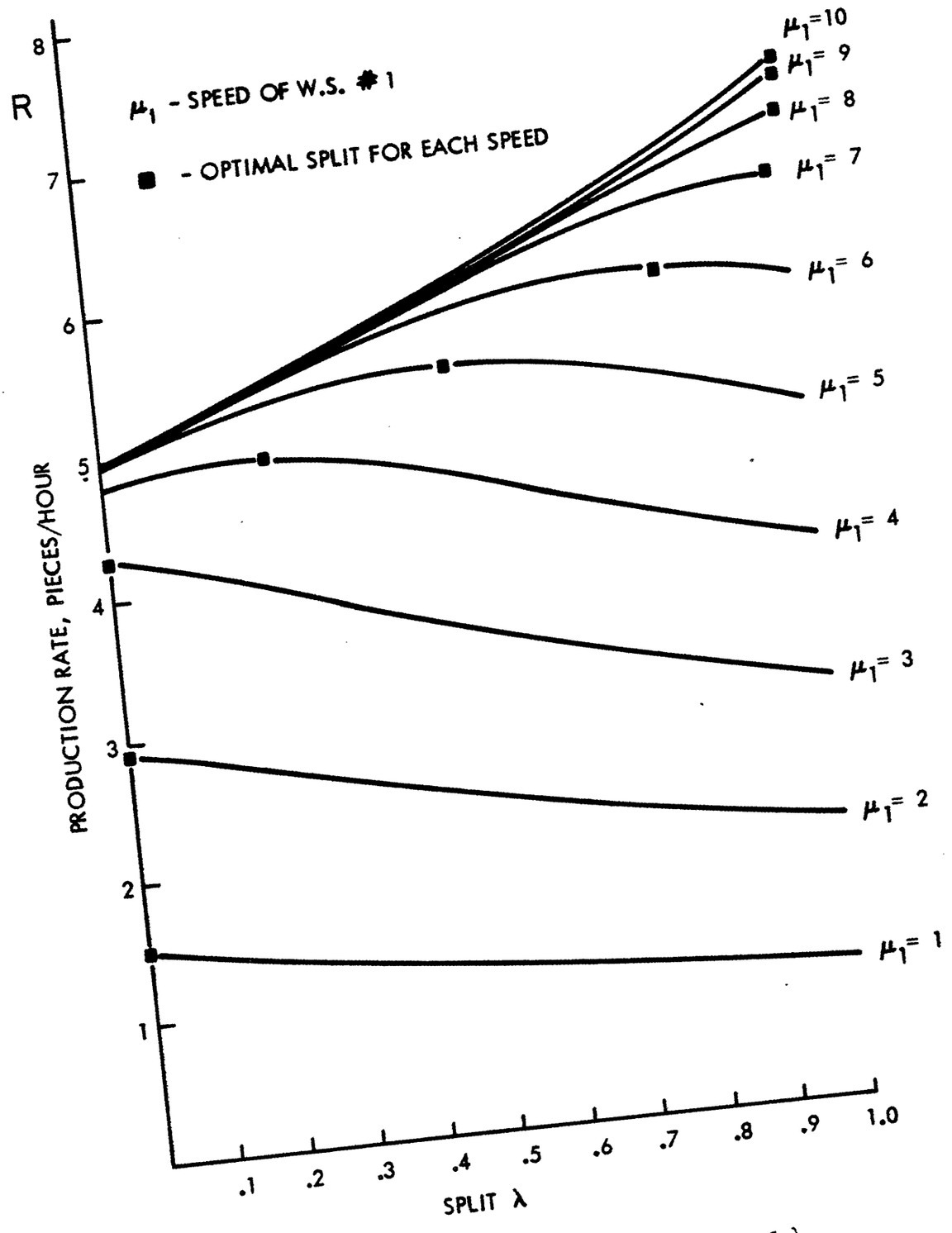


Fig. A.2. Production Rate as a Function of λ with μ_1 as Parameter

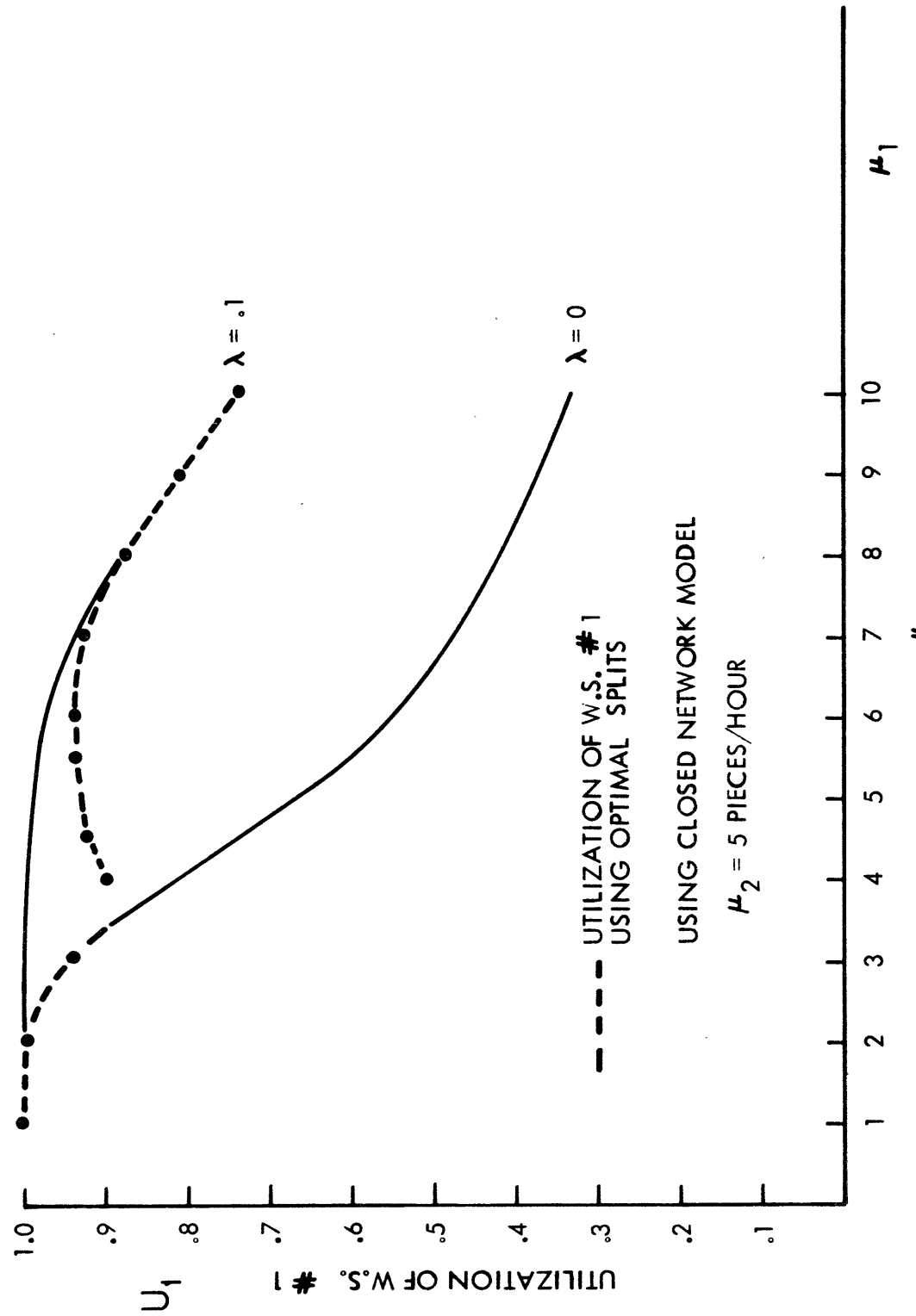


Fig. A.3. Utilization of Workstation 1 as a Function of μ_1

REFERENCES

Assad, A.A., "Multi Commodity Network Flows - A Survey," M.I.T. Ops. Research Center. Working Paper, Dec. 1975.

Baskett, F., Chandy, M., Muntz, R. and Palacios, F., "Open Closed and Mixed Networks of Queues with Different Classes of Customers," Journal of the A.C.M. Vol. 22, No. 2, April, 1975.

Bazaraa, M.S. and Shetty, C.M., Foundations of Optimization, Springer Verlag Lecture notes in Economics and Mathematical Systems, No. 122 (1976).

Bertsekas, D.P., "On Penalty and Multiplier Methods for Constrained Minimization," SIAM Journal on Control and Optimization Vol. 14, No.2, Feb. 1975.

Bertsekas, D.P., "Combined Primal - Dual and Penalty Methods for Constrained Minimizations," SIAM J. Control No. 3, May, 1975.

Betts, J.T., "An Accelerated Multiplier Method for Non-Linear Programming" Journal of Optimization Theory and Applications Vol. 21, No. 2, Feb. 1977.

Bradley, S., Hax, A. and Magnanti, T., Applied Mathematical Programming, Addison Wesley Publishing Co. 1977.

Cantor, D.G. and Gerla, M., "Optimal Routing in a Packet Switched Computer Network," I.E.E. Trans. on Automatic Control, Vol. c-23, No. 10, Oct. 1974.

Chandy M., "The Analysis and Solution for General Queueing Networks," Proc. of the 6th Annual Princeton Conference on Information Sciences and Systems, March, 1972.

Chandy, M., Hertzog, V. and Woo, L., "Approximate Analysis of General Queueing Networks," IBM Research report RC4931, July, 1974.

Chandy, M., Hertzog, V. and Woo, L., "Parametric Analysis of Queueing Networks," IBM Journal of Research and Development, Jan. 1973.

Chandy, M. Howard, J.H. and Towsley, D.F., "Product Form and Local Balance in Queueing Networks," Journal of the A.C.M. Vol. 24, No. 2, April, 1977.

Chang, A. and Lavenberg, S.S., "Workrates in Closed Queueing Networks with General Independent Servers," IBM Research report RJ 989, 1972.

Cook, N.H. and Subramanian, "Micro - Isotope Tool Wear Sensor," Materials Processing Laboratory M.I.T. Dept. of Mech. Eng. Sept. 1977.

Courtois, P.J., "Decomposability Instabilities and Saturation in Multiprogramming Systems," Comm. of the A.C.M. Vol. 18, No. 7, July, 1975.

Dantzig, G., "Linear Programming and Extensions," Princeton University Press, 1963.

Dafermos, S.C. and Sparrow, F.T., "The Traffic Assignment Problem for a General Network," Journal of Research, National Bureau of Standards, Vol. 73B, No. 2, 1969.

Defenderfer, J.E., "Comparative Analysis of Routing Algorithms for Computer Networks," M.I.T. Electronics Systems Lab. Report, No. ESL-R-756, March 1977.

Denning, P.J. and Buzen, J., "Operational Analysis of Queueing Systems," Third Int. Symposium of Modeling and Performance Evaluation of Computer Systems, Bonn, W. Germany, Oct. 1977.

Disney, R., "Random Flow in Queueing Networks: A review and critique," AIIE Transactions, Vol. 7, No. 3, Sept. 1975.

Dreyfus, S.E., "An Appraisal of Some Shortest Path Algorithms," Ops. Research 17, 3, 1969, pp. 395-415.

Eilon, S., "Multi - Product Scheduling in a Chemcial Plant," Management Sci. Vol. 15, No. 6. Feb. 1969.

Fisher, M.L., "Optimal Solution of Resource Constrained Network Scheduling Problems," Technical report, No. 56, M.I.T. Operations Research Center, Sept. 1970.

Frank, H., and Chou, W., "Routing in Computer Networks," Networks 1 pp. 99-112, 1971.

Gabbay, H., "A Hierarchical Approach to Production Planning," Technical Report No. 120, M.I.T. Operations Research Center, December 1975.

Gershwin, S. and Berman, O., "Complex Materials Handling and Assembly Systems, Vol. 7 - Analysis of Transfer Lines Consisting of Two Unreliable Machines with Random Processing Times and Finite Storage Buffers," M.I.T. Electronic Systems Lab. Report, No. ESL-FR-834-7, 1978.

Gershwin, S. and Schick, I.C., "Complex Materials Handling and Assembly Systems, Vol. 9 - Analysis of Transfer Lines Consisting of Three Unreliable Machines and Two Finite Storage Buffers," M.I.T. Electronic Systems Lab. Report No. ESL-FR-834-9, 1979.

Gelenbe, E., "On Approximate Computer Systems Models," Journal of the A.C.M. Vol. 22, No. 2, April, 1975, pp. 261-269.

Gelenbe, E. and Muntz, R., "Probabilistic Models of Computer Systems part 1," Acta Informatica Vol. 7, pp. 35-60, 1976.

Gelenbe, E. and Pujole, "Probabilistic Models of Computer Systems part 2; approximations to a single queue," Institut de Reserche d'Informatique et d'Automatique, Le Chesnay France, Research report 147, Dec. 1975.

Gordon, W., Newell, G.F., "Closed Queueing Systems with Exponential Servers," Operations Research, Vol. 15, pp. 254-254, 1967.

Gorenstein, S., "Planning Tire Production," Management Sci. Vol. 17, No. 2, October, 1970.

Hestenes, M.R., "Multiplier and Gradient Methods," Journal of Optimization Theory and Applications, Vol. 4, No. 5, Nov. 1969.

Himmelblau, D.M., Applied Non-Linear Programming, McGraw-Hill, 1972.

Hitz, K., "Scheduling of Flexible Flow Shops," M.I.T. Lab. for Information and Decision Systems, Report No. LIDS-R-879, Jan. 1979.

Horev, Y., Cook, N.H. and Ward, J., "Complex Materials Handling and Assembly Systems," Vol. 4: Discrete Simulation of a Flexible Manufacturing System: M.I.T. Electronic Systems Lab. Report No. ESL-FR-834-4, 1978.

Hughes, J.J., "Functional F.M.S. Components - Basic Elements and their Evolution," Proc. of Multi-Station, Digitally Controlled Manufacturing Systems Workshop, University of Wisconsin, Milwaukee, Jan. 1977.

Hutchinson, G.K., "The Control of Flexible Manufacturing Systems: Required Information and Algorithm Structures," IFAC Symp. on Information - Control Problems in Manufacturing Technology, Tokyo, Japan, Oct. 1977.

Hutchinson, G.K. and Hughes, J.J., "A Generalized Model of Flexible Manufacturing Systems," Proc. of Multi-Station, Digitally Controlled Manufacturing Systems Workshop, University of Wisconsin, Milwaukee, Jan. 1977.

Jackson, J.R., "Job Shop Like Queueing Systems", Management Science, Vol. 10, No. 1, Oct. 1963.

Kanellakis, P., "Complex Materials Handling and Assembly Systems, Vol. 5: Algorithms for a Scheduling Application of the Asymmetric Travelling Salesman Problem," M.I.T. Electronic Systems Lab. Report, No. ESL-FR-834-5, 1978.

Kershenbaum, A., Hsieh, W. and Golden, B., "Constrained Routing in Large Sparse Networks," IEEE International Conference on Computer Communication, Philadelphia, 1976.

Kleinrock, L., Queueing Systems Vol. 1: Theory, John Wiley and Sons, 1975.

Kleinrock, L., Queueing Systems Vol. 2: Computer Applications, John Wiley and Sons, 1976.

Kobayashi, H. "Applications of the Diffusion Approximation to Queueing Networks I: Equilibrium Queue Distribution," Journal of the A.C.M., Vol. 21, No. 2, April 1974.

Kobayashi, H. and Reiser, M., "Accuracy of the Diffusion Approximation for Some Queueing Systems," IBM Journal of Research and Development, Vol. 18, No. 2, March 1974.

Kuhn, P., "Analysis of Complex Queueing Networks by Decomposition," 8th Annual International Teletraffic Congress, Melbourne, Australia, No. 1976.

Lasdon, L.S., "Duality and Decomposition in Mathematical Programming," IEEE Trans. Systems Science and Cybernetics, Vol. SSC-4, No. 2, July, 1968.

Lasdon, L.S., Optimization Theory for Large Scale Systems, The McMillan Co., 1970.

Lavenberg, S.S., "Stability and Maximum Departure Rate of Certain Open Queueing Networks Having Finite Capacity Constraints," IBM Research Report RJ1625, July 1975.

Lee, S.M., "A Goal Programming Approach to Multi-Period Production Line Scheduling," Computers and Operational Research, Vol. 5, pp. 205-211, 1978.

Lemoine, J., "Networks of Queues - A Survey of Equilibrium Analysis" Management Science, Vol. 24, No. 4, Dec. 1977.

Lenz, J.E. and Talavage, J.J., "General Computerized Manufacturing Systems Simulator," Proc. of Conference on The Optimal Planning of Computerized Manufacturing Systems. Purdue University, November 1977.

Luenberger, D.G., Introduction to Linear and Non-Linear Programming, Addison Wesley Publishing Co., 1973.

McRainey, J.H., "Layout, Handling and Small Lot Production," Production Engineering Vol. 24, No. 12, December 1977.

Magnanti, T. and Golden, B., "Transportation Planning: Network Models and Their Implementation," M.I.T. Operations Research Centre Working Paper. Dec. 1977.

Malek-Zavarei, M. and Frisch, I.T., "A Constrained Maximum Flow Problem," Int. J. Control, Vol. 14, No. 3, pp. 549-560, 1971.

Miele, A., Moseley, P.E., Levy, A.V. and Coggins, G.M., "On the Method of Multipliers for Mathematical Programming Problems," Journal of Optimization Theory and Application Vol. 10, No. 1, 1972.

Nguyen, S., "A Unified Approach to Equilibrium Methods for Traffic Assignment," Proc. of Symposium on Traffic Equilibrium Methods, Montreal, 1974.

Powell, M.J.D., "A Method of Non-Linear Constraints in Minimization Problems," Proc. Symposium of the Inst. of Maths and its Application, University of Keele, 1968.

Rockafellar, R.T., "Augmented Lagrange Multiplier Functions and Duality in Non-Linear Programming," SIAM J. of Control, Vol. 12, No. 2, May, 1974.

Rockafellar, R.T., "A Dual Approach to Solving Non-Linear Programming Problems by Unconstrained Optimization," Math Programming, 5 (1973) pp. 354-373.

Royce, N.J., "Linear Programming Applied to Production Planning and Operation of a Chemical Process" Operations Research Quarterly, Vol. 21, No. 1, 1970.

Salkin, H.M., Integer Programming, Addison Wesley Publishing Co. 1975.

Schick, I.C. and Gershwin, S., "Complex Material Handling and Assembly Systems, Vol. 6: Modeling and Analysis of Unreliable Transfer Lines with Finite Interstate Buffers," M.I.T. Electronic Systems Lab. Report No. ESL-FR-834-6, 1978.

Secco-Suardo, G., "Complex Materials Handling and Assembly Systems Vol. 3: Optimization of a Closed Network or Queues," M.I.T. Electronic Systems Lab. Report, No. ESL-FR-834-3, 1978.

Sivazlian, B.D. and Stanfel, L.E., Analysis of Systems in Operations Research, Prentice-Hall, 1975.

Solberg, J.J., "A Mathematical Model of Computerized Manufacturing Systems," Proc. of Conference on Optimal Planning of Computerized Manufacturing Systems, Purdue University, No. 1977.

Stecke, K.E., "Experimental Investigation of the Scheduling Problems of a Particular Computerized Manufacturing System," Proc. of Conference on Optimal Planning of Computerized Manufacturing Systems, Purdue University, Nov. 1977.

Steenbrink, P.A., Optimization of Transportation Networks, John Wiley and Sons, Ltd., 1974.

Stern, H.I., Rodriguez, E.P. and Utter, M., "Cyclical Job Sequencing on Multiple Sets of Identical Machines," Naval Research Quarterly, Logistics, Vol. 24, No. 1, March, 1977.

Ward, J., "Complex Materials Handling and Assembly Systems, Vol. 8 Numerical Experience with a Closed Network of Queues Model," M.I.T. Electronic Systems Lab. Report, No. LIDS-FR-834-8, 1980.

Wollmer, R.D., "Multi Commodity Networks with Resource Constraints: The Generalized Multi Commodity Problem," Networks 1, pp. 245-263, 1972.