[Multi-stage]

MULI-STAGE MULTI-PRODUCT LOTSIZE SEQUENCING OF OPERATIONS

by

BARTHOLOMEW OKECHUKWU NNAJI

Thesis submitted to the Graduate Facultv of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

MASTERS OF SCIENCE

in

Industrial Engineering and Operation Research

APPROVED:

R. P DAVIS, chairman

J. W. SCHMIDT

R. A. WYSK

January, 1982
Blacksburg, Virginia

ABSTRACT

MULTI-STAGE, MULTI-PRODUCT LOTSIZE SEQUENCING OF OPERATIONS

by

Bartholomew Okechukwu Nnaji

In this thesis, a multiple stage - product sequence selection problem was mathematically modelled and analyzed. A fortran program that calculates the machine requirements for each machine station and each product type, and searches for the optimal product sequence combination was developed. Four cases of the sequence selection problem were analyzed in detail. A comparison of homogenous machine stations and product lines was made in an analytical manner. Results of an example problem solved for the above types of systems were presented. Computational results for different sequence combinations (ranging from 8 to 125) are discussed.

ACKNOWLEDGEMENTS

I am deeply indepted to Dr Robert P. Davis for inspiring my interest in the area of this research. During my one year at Virginia Polytechnic and State University,Dr Davis has been a teacher, a constant source of motivation and inspiration and an invaluable friend. His tremendous patience, creative insight and relentless drive for excellence have culminated in this work. I am also indepted to Dr Richard A. Wysk and Dr J. W. Schmidt for their invaluable suggestions and guidance throughout this study.

I also wish to express my gratitude to the secretaries of the Industrial Engineering Department(Mrs Laura Moran Lopez, Mrs Joni Chambers and Ms Donna Lovern) for their help with typing equipment during the finalization of this study.

Finally, my wife Patricia deserves special thanks for the long hours she put in the typing of this thesis draft. For her patience and constant understanding, I am grateful.

CONTENTS

Appendix

page

LIST OF TABLES

LIST OF FIGURES

Figure

page

Chapter I

INTRODUCTION

1.1 SEQUENCE SELECTION PROBLEMS

The optimal planning of production systems has always posed a tremendous challenge to the production engineer. Today, with the increasing complexity of production systems, this problem is far more varied, complex and interactive than ever before. Although this research centers on production systems, similar problems exist in the planning of other systems (i.e., social, political, health care, etc.).

A major objective of most production systems is to achieve a reduction in the cost of production leading subsequently to a maximization of profit. The slogan that "time is money" is so true in production environments that the need for the elimination of unnecessary delays cannot be overemphasized. Furthermore, it would be unusual to assume that a product can take any amount of time to process through a manufacturing system. Demand for products typically dictates an upper bound on processing time. But obviously, time is not the only thing one may wish to save while manufacturing products. Reductions in production cost also constitute an important aim of production planners.

There exist many planning and design decisions which have a significant effect on both the time and cost associated with manufacturing a product. Two of the most fundamental of these decision issues are: process sequence selection and machine requirements planning. This thesis will focus on the modelling and analysis of process sequence selection decisions and their effect on machine requirements in the context of manufacturing flow systems planning.

Background

Sequencing is the order in which activities(processes) are performed in a manufacturing system. Route deals with the specific path taken to accomplish the various processes necessary for completion of a task.

The sequencing problem can be categorized into two broad classes:

(a) Static or deterministic sequencing

(b) Dynamic or stochastic sequencing

Figure 1 is a schematic diagram classifying the sequencing problem [11]. It begins by categorizing the nature of the production demands as arrivals to the manufacturing system.

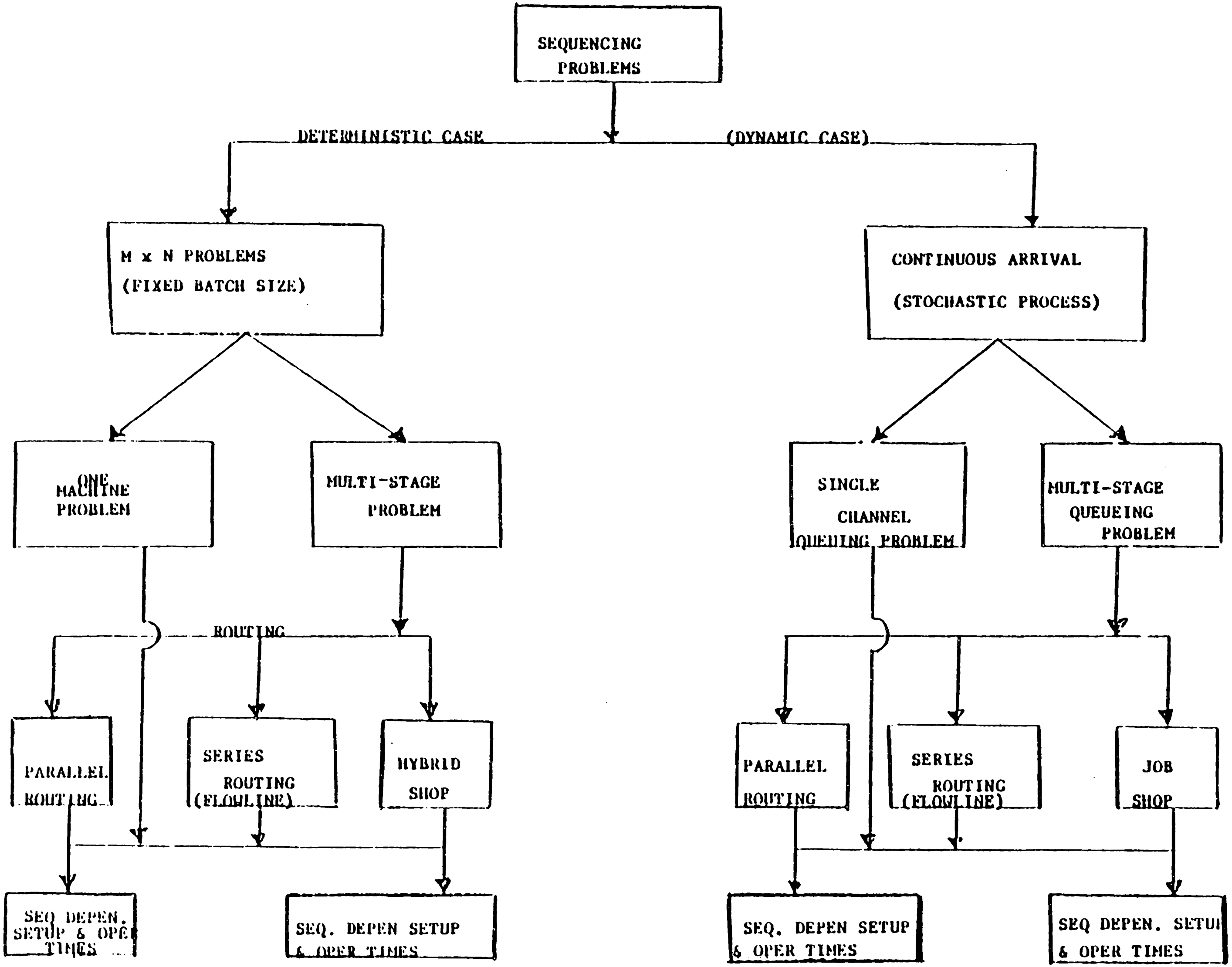


Figure 1: Categories of the Sequencing Problem

These arrivals are considered to be either deterministic or stochastic batch sizes.

Each type of arrival process is then classified by the number of machines(M) required (single stage as opposed to multi-stage process, i.e. $M = 1$ vs $M > 1$).

Next comes the characteristic of the job route and finally, the nature of the set up and operation times (i.e. dependent vs independent set up and operation times). Of course, further refinement could be added to the classification scheme: for example, it is possible to have set up time dependence and operation time independence and vice versa. However, the above serves to capture the fundamental nature of sequencing problems.

In production planning, it is usual to account for the generation of defective parts at various stages in the manufacturing system. In some processes, the percentage of these defects can be very high. Of major importance to the production planner is ensuring that defective products will not be passed on to the consumer since the repercussions inherent in such a practice can be quite devastating. At the same time, the planner has to meet demands and due dates. This problem can be accomodated by having in-process inventory. But of course, the cost of in-process inventory has to be such that it will merit its existence. Another approach to this problem could be to always start production with more items than would be required by the demand. This means that if the system is a single operation and the defective proportion is 0.1%, and the demand for a given period is M products. then it would be necessary to start with [1 + 0.1/100] x M items.

As stated earlier, the problems of the planner are quite varied and complex. The sequencing problem therefore constitutes a multi-faceted challenge to the planner in that it is directly related to both the batch size selected for production as well as the planned production capacity(i.e.,machine requirements). The specific aspects of these reseach issues that are addressed in this thesis are characterized by the problem definition given below.

1.2 PROBLEM DEFINITION

The definition of process sequencing refers to the order in which operations are performed on a product(or products). The problem of job scheduling will not be treated here. Route is not only the sequence by which operations are performed, but also the actual machine visits required to perform those operations.

A definition of the problem of interest in this research is best established from the production system characteristics given below:

1. There are N products and M processes,

2. Some of the products may go through fewer than the M processes,

3. Each process required by a product may also be required by one of the other products,

4. There is a certain percentage of defectives generated by each of the processes performed on each of the products,

5. The facility with which the sequencing problem is associated is not an existing facility,

6. There is a daily demand for each product,

7. The processing time for each operation is a function of predecessor operations,

8. Setup time is a function of (a) product predecessors, (b) process predecessors,

9. Travel time between processes is assumed to be negligible,

10. Batch production is allowed for all products.

1.3 ASSUMPTIONS

Certain basic assumptions apply to the problems addressed in this research.

1. The technological order of all the products through all the processes is known in advance [23],

2. The technological order may be different for each product,

3. The processing times and setup times for each machine for each product are deterministic and known, assuming knowledge of precedences,

4. Pre-empting is not allowed. That is, no splitting of jobs. Once a job is started on a machine, it will be completed without interruption,

5. The objective to be achieved can be either to mimimize total elapsed time from the start of the first job to the completion of the last job, or to minimize the total cost of production,

6. All jobs are of equal importance [27],

7. All jobs are processed as soon as possible,

8. All N jobs are simultaneously available at the beginning of each planning period,

9. All M machines are available at the beginning of the planning period and are ready to take on any of the N jobs,

10. A machine can process only one batch at a time,

11. In-process inventory is permitted,

and, finally,

12. Excess time available on any machine can be used for some other useful and profitable work [27].

1.4 RESEARCH OBJECTIVE

The objective of this study is to develop a mathematical model of the process sequencing decision process which will capture the problem characteristics, and also the assumptions as stated above.

To achieve this objective, certain control variables(decisions) need be stated. These control variables can be modified(or relaxed) to produce different models with possibly different constraints. They also depict the nature of the product and its movement within the system. The amount of time spent in the system would be a function of these control variables, as would the production output of the system, and consequently, the overall result of accomplishing the objective of the system's operations.

The specific control variables are identified as follows:

Sequence: The state of a product in the model will depend on the sequence of operations perfomed. This is generally true in sequencing problems; but, in this study, some other parameters are functions of the state of the product as well. These are as follows:

a. Processing time - The processing time of a given machine or processor is not constant, rather, it varies depending on the state of the product.

b. Setup time - The setup time is also a function of the state of the product as that product comes to the machine in consideration.

Batch-size : The quantity of products carried between stages will be affected by the state of the product. This means that replenishing of products between stages, to account for defectives, will not be permitted.

Machines: The number of machines required at a particular processing station may depend on the state of the product as well as the batch size chosen for each product processed.

The determination of how these control variables are related in their effect on production plannning for minimum cost(or time) constitutes the crux of the problem for investigation and analysis in this thesis. The type of algorithm(s) employed in the solution of the model developed in this research is portrayed by the nature and interrelationships of these control variables. A deterministic planning perspective is taken throughout this research.

1.5 SIGNIFICANCE OF THE PROBLEM

It is not immediately obvious how one can isolate the sequencing problem, or determine the impact of sequencing decisions on other decisions, or vise versa [11]. It is quite possible for an facility not to be aware of its sequencing problems because they are shielded, from showing a strong interdependence, by other components of the total cost of production (e.g., accounting, inventory, capital budgeting, loading, engineering).

The choice of one criterion as a measure of performance in the system is really a drastic simplification. This automatically eliminates specific consideration of some of the problems in an operational shop. Some of the factors affecting (or affected by) the sequencing problem, each of which may either be maximized or minimized to achieve an optimum solution, are as follows:

a. Relative to the facility: idle capacity, setup cost, output, material handling cost, reserve capacity, utilization, operation cost, etc..

b. Relative to products: in-process inventory, due dates, routes, technological requirements, raw material and finished product inventory, makespan, obsolescence and deterioration losses, etc..

For management these problems are very pertinent but no one has really been able to model a production system in such a way that individual and collective optimality can be achieved for every possible objective. Doing this would involve solving all facility planning, scheduling, and control problems ever investigated.

Sequencing problems arise from factors relative to both a facility and its products. As a result, it is important to model the sequencing problem in such a way as to incorporate components of these two areas. Modelling sequencing problems using cost of production and minimum elapsed time as criteria enables this.

The production planner is always interested in knowing more about the interrelationship of a production system's components. This knowledge aids in making better decisions about a system's operation, which would subsequently improve production cost, quality, etc.. Sequencing models which capture product and facility related parameters and decisions (control variables) serve as an integrating perspective from which the production planner can better view a system's operational effectiveness.

1.6 RESEARCH ORGANIZATION

Thus far, a broad view of the objective of this research has been presented. Chapter II centers on a review of earlier work on the sequencing problem. In Chapter III,a mathematical model of the sequence selection problem is presented. Methods of approach to solution of the problem are also discussed. In Chapter IV, the analysis of the model is presented,starting with calculating the frequency of moves for each product type. The calculation of the number of machines required at each work station is also presented in this chapter.

The solution of the single product-fixed sequence problem is presented in Chapter V, along with the solution of single product-varying sequence and multiple product-fixed sequence problems. The multiple product-varying sequence problem is also treated in this chapter. Chapter VI contains the summary and conclusions.

Chapter II

LITERATURE REVIEW

Sequncing problems arise quite naturally in everyday life for example at large ports (e.g., the sequencing of ships on berths), in manufacturing shops (e.g., the sequencing of different order of paints on a mixing facility),at service facilities(e.g., the sequencing of repairs at a gas station), etc. The list could be overwhelming. The classical example relates to the problem of sequencing n jobs on m machines in which each job has its own individual route, which may be distinct from the other jobs. This can be found as early as the 1920's in books on shop production control [17].

But it was not until 1954 that work on analytical models for sequencing problems was reported in the literature [17]. Before then, there was no reported attempt to treat the sequencing problem from the point of view of analytical models, with an explicit intention of optimizing some measure of performance. Nevertheless, the sequencing problem has attracted considerable research.

2.1 BACKGROUND

A major part of sequencing research seems to be concerned with the effects of both scheduling and sequencing on various measures of shop performance criteria [11]. These effects are often studied in one of the following contexts:

1. Number of component parts comprizing a job.
    a. Single-component jobs.
    b. Multi-component jobs which require assembly and/or subassembly jobs
2. Production factors which constitute the shop.
    a. Machines
    b. Labor and machines
3. Jobs available for processing.

N jobs to be scheduled/sequenced, where N is finite. This is often referred to as static sequencing/scheduling (Day et al [11]). mentioned in their review that few articles have reported work on the sequencing of multi-component jobs. Because of the inherent difference among specific sequencing problem contexts, different models may evolve due to these three factors:

1. Parameters and variables included
2. Assumptions, constraints, and relationships
3. Objective criteria

Figure 1 in Chapter I gave a classification of the sequencing problem. The following literature review will progress with that framework.

2.2 STOCHASTIC SYSTEMS

The book by Conway, Maxwell and Miller [10] gives a good exposure to the probabilistic sequencing problem. Probabilistic elements enter into the problem formulation in one of these three forms:

a. The set of n jobs dynamically varies in a stochastic fashion.

b. The requirements of each job (i.e. route, processing time, set up time etc.) vary stochastically.

c. The nature of the processors (availability, suitability, number of processors, etc.) vary [17].

A dynamic model is usually such that data could be obtained from the system at definite points in time. This is the feedback of capability necessary for dynamic sequencing. Most of the work here is done using Monte Carlo Simulation.

Single Channel Queueing Problem

Work in this area has been quite extensive. The typical assumptions associated with queueing models of sequencing problems are as follows:

(1) Poisson Arrivals, (2) exponential service times, (3) First-come, First-Served (FCFS) queue discipline, and (4) Mean service rate greater than mean arrival rate of demand (orders) [11]. Evaluation criteria typically employed are usually:

a. Expected length of waiting line or average number of jobs in the system.

b. Average job completion time.

c. Machine utilization.

d. Average waiting time for a job, which is the average completion time minus the average processing time. Work in this area of stochastic sequencing problems can be found in [6],[30], [10, Chapters 8,9,and 14].

2.3 MULTI-STAGE QUEUEING SYSTEMS

The stochastic multi-stage queueing problem can be classified by three principal arrangements of machines, as described in reference [10]: (a) Parallel channel queues having a number of identical machines working in parallel to provide a single type of service; this corresponds to a machine center of several interchangeable machines, (b) Tandem queues, or queues in series corresponding to the flowshop environment, (c) General workshop environment corresponding to a job shop in which each order is processed by a particular subset of machines in the work center.

subsection 'Parallel queues'

Certain assumptions are usually made about this kind of environment: Poisson arrival rates, exponential service times, and first come first served discipline. Using these assumptions, the following quantities are then analysed: (a) mean number of jobs in a waiting line, (b) mean number of jobs in the system, (c) mean waiting time of a job, and (d) mean time (including service time) of a job in the system [11],[50].

2.3.1 Tandem Queues

In a flowshop situation, machines (or machine groups) may be numbered so that every job routing specifies an increasing sequence of machine number identifiers. The input to a machine (machine group) consists of inputs from a dynamic job file or from machines (machine groups) with lower indices. Jobs can also exit from the shop after being processed on any machine (machine group). Not much work has been done in this area in comparison to the dynamic job shop situation [9],[11],[17].

2.4 DYNAMIC JOBSHOP (THEORY AND RESEARCH)

For more than two decades, a large number of simulation experiments have shown the advantages of considering the job shop as a network of waiting lines with fixed, short-term capacity (where the short-term could be an 8-hour shift). This realization gave much inspiration to the pursuit of research in this direction [32]. It must be mentioned that early work in this area focused on queue discipline (i.e., priority dispatching rule).

2.5 JACKSON'S DECOMPOSITION PRINCIPLE

Jackson's decompostion rule [32] is probably the most powerful technique used in the study of the dynamic job shop. The following assumptions are necessary for application of the principle:

(a) Exponential arrival distribution

(b) Exponential process time distribution

(c) Jobs are routed to a machine by a fixed probability transition matrix. This matrix defines the probablity of going from either a machine to any other machine or from a machine to the customer.

(d) First-come,First-served(FCFS) priority rule applies.

With these assumptions, the system can be divided into a network of independent, individual machine queueing systems. This is because under the FCFS assumption, the output of a single queue is exponential if the interarrival and service times are exponential. Hence, in the dynamic system, the output of one machine becomes the input to the next in a network of queues [11],[17].

2.6 GENERAL DESCRIPTION OF DYNAMIC JOB SHOP MODELS

Typical assumptions in the study of the dynamic job shop are as follows[11]:

1. Arrival and process times are generated from stated probability density functions.
2. Mean arrival rate of jobs is less than the mean service rate of the shop.
3. No preempting is allowed (i.e. no splitting of jobs).
4. Due dates are fixed.
5. Set up time for a given operation is included in the processing time of the job being serviced.
6. Flowtime between machines is negligible.
7. There are no groups of similar machines.
8. Machine breakdowns are not allowed.

9. Machines are never unable to perform a required task for lack of operator, tool or material.

10. The generation of job route does not allow for consecutive operations on the same machine.

11. Alternate routing is not considered.

12. Lot splitting is not considered.

13. No overtime or subcontracting is permitted.

14. No allowance is made for scrap or rework.

15. Statistics are usually obtained when the system is operating in steady state.

Most dynamic job shop articles usually contain at least half of these assumptions [11].

The stochastic nature of a job shop has prompted the use of Monte Carlo Simulation by researchers. Certain factors appear common to all these simulation experiments.

2.6.1 Parameters of the Shop Load

a. Mean arrival rate of jobs.

b. Mean processing rate at various machines or machine centers,

c. Number of machines or machine centers corresponding to shop size.

2.6.2 Chacteristics

a. Distribution of arrivals.

b. Distribution of processing time at machines or machine centers, and

c. Procedure for generating job routes(i.e., Queue discipline, and the policy of sequencing jobs which are waiting on line). The factors listed above are now considered in more detail.

2.6.3 Order of Arrivals

Most dynamic job shop simulations assume Poisson arrival rates[17],[28].

2.6.4 Processing Time

Process time is usually generated from a service time distribution. Although different service time distributions could be assigned to each of the machines or machine centers, this process time is usually fixed[17],[28]. Other distributions suggested include Erlang, hyperexponential, the normal distribution and the log normal distrbution [11],[17].

2.6.5 Shop Size

When simulating an actual system's operation, the number of machines in the shop will be known; but for synthetic simulation, this number is not available and so a decision must be made concerning this value.

Usually, the number of machines in a dynamic shop simulation are small. This is because researchers like Baker and Dzielinske [11] have tested shops of size 9 to 30 machines and found the size to be an insignificant variable in the problem.

2.6.6 Routing

Usually job routes are generated by means of a transition matrix - a stationary Markov chain. If there are M different machines in the shop, then the Markov chain will consist of an Mx(M + 1) matrix whose probabilities across any row of the matrix will sum to unity. The last row of the matrix is the probability of leaving the shop as a completed job.

To work with this matrix, the first machine of the job route must be specified. This can be done by random sampling from a predetermined distribution. Given the first machine assignment, the number of the next machine (if it exists) in the job route is chosen by generating a number from the distribution associated with the row of the Markov matrix which corresponds to the first machine assigned. If the number generated is in the last column of the matrix, then there is one machine assignment. If not, a second machine is assigned to the job route and this becomes the new row from which the third machine assignment can be made. The process continues until a number is selected in the last column of the matrix.

If the transition matrix consists of equal probabilities then the situation is described as a pure job shop. A pure flowshop, however, contains only 0's and 1's as entries [10],[11].

A disadvantage of synthetic, dynamic shop simulation is that many of them do not consider alternate routing; while in a realistic job shop, alternate routing could be of great importance. Alternate route simply means that if a transition matrix is used in generating job routes, then it will be possible to randomize the machine numbers with the assumption that on each segment there is no precedence constraint to be considered. This way, there is no particular order for machines on that segment [11].

Assignment of Due Dates

Management usually requires jobs to have due dates because of customer demands; but further it wants to be able to predict deliveries and also because it would like to maintain control over production. For jobs completed early, storage may be needed and for jobs completed late, some penality may accrue. Such a penalty may be monetary or nonmonetary (e.g., loss of customers). For a classification of due date considerations, the reader may refer to the sequencing review by Day and Hottenstein [11].

Priority Rules (Types):

No specific priority rule has been able to guarantee optimality: however, Day et al. pointed out that it has been shown that priority consideration can accelerate the average time of all jobs [11]. A classification of the priority rules (as they appear in [10]) is as follows:

1. Lateness rules - rules that determine priority according to some increasing function of lateness.

2. Arrival order rules - rules that assign priority according to the order in which jobs arrive at the machine under consideration.

3. Rules that determine priority according to some property of the job itself.

4. Random rule - a rule which assigns priority at random.

The effectiveness of any of these rules is measured by how well it meets a predetermined criterion (criteria). Further discussion in this area will be found in [11]. However, one of these rules is described below.

SOT [Shortest Operation Time]:

This is the minimum amount of time spent in the system (or per operation) by the product assuming set up times are not considered. Tests done by various researchers, like Conway and Maxwell, point to the fact that SOT is quite effective since it performs well under every measure [10].

2.7 STATIC SEQUENCING

The static sequencing problem has been analysed through:

1. Combinatorial Programming approaches
2. Reliable Heuristics
3. Monte Carlo Sampling and
4. General Mathematical Programming

Just as in stochastic sequencing problems, deterministic sequencing problems have single and multiprocessor components.

The single processor [56] is important because (1) certain systems are modelled as single processor systems and, (2) consideration of one processor systems can shed some light into more complex multiprocessor systems [17].

The review of research by Graham et al. [24] provides good insight into the various systems of deterministic sequencing problem. The author must mention, however, that due to the mathematical presentation of this work, it does not make for easy reading. Work on multiprocessors can be found in [17],[19],[23],[31],[45],[47],[55].

Problems exhibit various arrangements of facilities and of flow of work through these facilities . The usual assumptions in this kind of multiprocessor system are as follows[14],[43],[44]:

a. No preemptive priorities.

b. No machine may process more than one product at a time (this is usually a desirable feature).

c. Each job is homogenous, even though the job may be composed of individual parts. This is how job splitting and assembly of operations can be eliminated.

d. No job cancellation once started.

e. No job may be on two processors at a time (eliminates lap phasing).

f. Processing times and set up times are independent of sequence.

g. Due dates (if any), are fixed.

h. In-process inventory between processors is permitted.

i. Job routing is prescribed in advance.

j. There is only one of each type of processor.

One or more of these assumptions may be dropped by a researcher depending on what kind of system he is considering. For example, assumption (e) will be dropped when identical processors are considered.

Obtaining an optimal solution, or near optimal solution, depends on which of the above assumptions is relaxed and which ones are enforced. Several criteria may be used to measure the utility of strategies, algorithms or policies.For example:

a. Criteria that do not distinguish between individual jobs but are related to an aggregate measure (which is related to the sequence in an aggregate sense). An example of such a measure is the makespan, which is usually minimized [14].

b. Criteria which do distinguish among jobs are therefore related to the position of the job in the sequence. Examples of such criteria are: minimize total tardiness, minimize total cost of production, etc..

The static sequencing problem has been solved using the following major approaches.

2.7.1 Combinational Approaches

This involves problem solving procedures developed on two principal techniques.

1. The use of a controlled enumeration technique for (implicitly) considering all potential solutions [21].

2. The elimination from explicit consideration of particular potential solutions which are known from dominance, bounding and feasibility considerations to be unacceptable [58]. Alternative names for such procedures are:

(a) Branch and bound, (b) Reliable heuristics, (c) Controlled enumeration [41].

Branch and bound is the name given to the ideas of Little et al. in their algorithm for solving the Travelling Salesman Problem[42]. The branch notion comes from a tree analogy, the procedure is always concerned with choosing a branch of the tree to be evaluated. The bound signifies an emphasis on the effective use of means for bounding the value of the objective function at each node in the tree, both for eliminating a dominated path and for selecting a branch for evaluation [2],[8],[10],[11] [24],[29],[58].

2.7.2 Reliable Heuristics

This term, sometimes called combinational programming [58] or controlled enumeration, is a recent attempt to accomodate the masive combinatorial nature of sequencing problems without suffering major,detrimental consequences. Generally, these procedures work on two principal concepts:

a. The use of controlled enumeration to implicitly consider all potential solutions, and

b. The elimination from explicit consideration of particular potential solutions which are known from dominance, bounding and feasibility considerations to be unacceptable.

In general, these procedures guarantee the discovery of an acceptable solution when one exists, or the knowledge that none exist [17],[26],[38], [46],[49],[57],[59].

2.7.3 Monte Carlo Sampling

Consider the case of an N job, M machine sequencing problem. The maximum number of sequences which can be generated if one allows for the possibility of a different sequence on each machine, is $(N!)$. For large N and M, the enumeration of all such sequences to find an optimum becomes impractical, and this is where Monte Carlo sampling comes into play.

Heller applied a Monte Carlo Sampling technique to a flow shop situation and found that if the processing time on each machine is randomly assigned from a probability distribution function, uniformly distributed between zero and nine units of time, the resulting distribution of schedule times for a large number of experiments is normally distributed [11]. Elmaghraby used this result to consider the cost and general effect of continued sampling with the aim of a shorter schedule time than the generated sequences [17].

2.7.4 General Mathematical Programming

Effort in this study is concentrated on the application of mathematical programming techniques, and so, a more detailed discussion of some of these techniques will be presented [11],[21],[22],[23],[26],[37],[38],[39],[47],[64].

Many modelling and solution techniques exist, in general, in mathematical programming (a) Network flows, (b) Linear programming, (c) Dynamic programming, (d) Integer programming, (e) Quadratic programming, (f) Lagrangian methods, etc.. Each of these techniques has its beauty, elegance and advantages. But at the same time, some limitations also exist with their use. Some of them are even quite disadvantageous to use in certain problems while others prove prohibitive to apply.

2.7.5 Network Flow

A combination of a set of nodes (or vertices) represented by circles, and a set of arcs (or edges, or branches, or activities) represented with line segments or arrows connecting two neighboring nodes, is called a network. Consider a case of directed arcs [29]. When there exists an arc E which connects nodes N and M , these two nodes are connected. If there exists a sequence of nodes and arcs such as N , E , A , C ,..., B , D , M , this is called a chain or path from node N to node M . When M = N it constructs a directed cycle. An acyclic directed network is one having directed paths and containing no cycles or loops. Most production problems are of this nature.

The definition above is almost similar to one for a graph, however, a positive value d , called capacity, is defined on arcs in the network. This capacity could be the processing time, the cost of processing, or just as in the case of this study, an objective function which could be either a cost or a time function. The objective function would then have to be evaluated for the arcs.

If minimum capacity is desired, then the problem could be solved using the shortest chain (or path) method. A good description of this type of problem can be seen in references [27],[51].

2.7.6 Linear Programming

Capacity problems can be solved using linear programming procedures, which deal with the problem of minimizing or maximizing a linear function in the presence of linear inequalities. Since the development of this method in 1947 by George B. Dantzig, linear programming has been used extensively in the military, industrial, governmental and urban planning fields, among others [3], [3],[4],[39],[47],[64].

The popularity of linear programming can be attributed to many factors including its ability to accomodate large problems in a reasonable amount of time by the use of the simplex method and computers.

Certain assumptions are usually implicit in formulating a problem as a linear programming model.

1. Proportionality: Given a decision variable X_j, its contribution to cost is C_jX_j and its contribution to the i^{th} constraint is $a_{ij}X_j$. This means that if X_j is doubled, say, so is its contribution to cost and to each of the constraints.

2. Additivity: This guarantees that the total cost is the sum of individual costs, and that the total contribution to the i^{th} restriction is the sum of the individual contributions of the individual activities.

3. Divisibility: This assumption ensures that the decision variables can be divided into any fractional levels so that noninteger values for the decision variables are permitted. The L.P. formulation generally is of this form

$$\text{Min(Max) } Z = \sum_{j=1}^{n} C_j X_j$$

$$\text{Subject to } \sum_{j=1}^{n} a_{ij} x_j \geq (\leq) b_i$$

$$X_j \geq 0$$

$$\text{where} \quad i = 1,2,\ldots,m$$

$$j = 1,2,\ldots,n$$

Z is the objective function (or criterion function) C 's are the cost coefficients and X_j is the decision vector. X ≥ 0 is the nonnegativity restriction and $\sum_{j=1}^{n} a_{ij} x_j \geq b_i$ denotes the i^{th} constraint on the problem. The solution should give a set of variables, x_1, $x_2, \ldots, x_N$, which form a feasible region.

Among all feasible vectors, the one(s) which minimizes (or maximizes) the objective function is sought.

Formulations of this nature are found in numerous texts such as reference [3]. Applications can be found in [39],[64].

2.7.7 Dynamic Programming

Due to the fact that serial systems have a sequential structure, they can occasionally be transformed from N decision, one-state, initial value optimization problems into a set of N one-decision, one state problems. The basic type of models and methodologies associated with such decomposible systems was developed by Richard Bellman and is called dynamic programming.

The basic structure of a dynamic program is as follows [2],[4],[7],[16],[18],[20],[33],[35],[40],[55],[61]:

a. The problem can be divided into stages (i.e.,decomposed).

b. The state of the system at each stage is expressed by state vector, s .

c. At each stage a decision is made to change the state of the system. Only the present value of the state vector is associated with this decision. For a deterministic, discrete system, the state of the system at stage i + 1 is expressed by state i, and decision d_i at that stage as follows:

$s_{i+1} = T[s_i, d_i]$ where T is the transformation [29].

d. A measure of performance to be optimized is established for the multistage decision process.

e. A possible sequence of decisions, e.g., $[d_1, d_2, \ldots, d_N]$, where N is the number of stages, for a deterministic, discrete case, is the policy. A policy which maximizes or (minimizes) the measure of performance is the optimal policy(i.e., results in an optimal solution for the multistage decision problem).

f. Bellman's principle of optimality states that an optimal policy has the property that whatever the initial state and decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision.

g. Based upon the above principle, an optimal sequence of decisions for determining the optimal value for a measure of performance, which is to be obtained as the function of the initial state and the number of stages, is successively determined by the recursive functional equation.

The dynamic programming application to sequencing problems have been limited due to the fact that its associated solution techniques can handle only problems with a small number of states. However, the ability of this method to decompose an N-decision, one state problem into a set of N one decision, one state optimization problems is very valuable in the analysis of smaller scale problems since such analysis often provides a state dependent optimum.

2.8 OTHER TECHNIQUES:

Some other techniques employed in the modelling and solution of sequencing problems are:

a. Gantt chart[52].

b. Pairwise interchange methods [19].

c. Transition Point Method[62].

e. Quadratic programming[15].

There are cases in industry where a decision has to be made about the selection of sequence of a operations in a manufacturing environment. The paper by Wysk, Barash and Moodie[64] gives a good exposure to this kind of problem. The paper explains that the difference, or differences, in plans of manufacturing processes might be the selection of different machining parameters (speed,feed,depth of cut). In more extreme cases, the planner might have selected different machines for each plan. (For example a hole can be cored by the use of an end drill or a core drill). It is problems of this kind that initiate the sequence selection problem which is the subject of this study.

Chapter III

MATHEMATICAL MODEL DEVELOPMENT OF THE PROBLEM

The type of systems which constitute the subject of this study have been introduced and described. With that framework, the model is now mathematically developed. Since one cannot represent all aspects of sequencing problems in one model, the specific type of system studied in this thesis is described below.

The system of interest is a multistage system with different machines performing different operations. The number of machines in a work center may not equal the number of operations. Parallel processing will be permitted in order for the system to meet due dates and demands. Multiple items are produced by the system. The setup and operation times are functions of the state of the product in the system. The products could be carried in batches between stages, however, each batch contains only one product type. As products travel between stages some defectives occur and the amount defective is represented as a percentage of the number of products coming into that particular stage. The facility is not an already existing facility.

The objective will be to model this problem mathematically. Since each product can be produced by going through alternate sequences, it becomes a problem of which sequence will be chosen for each of the various product types. A combinatorial problem emerges. Hence, a search routine is employed which enumerates all possible sequences and selects the optimal sequence for each product type. Later,the relative differences in the execution time of a program that searches for this optimality is explored for various combinations of sequence and product. Relationships of the model to other sequencing problem components, such as the machine types, will be discussed.

3.1 CHARACTERISTICS

Certain characteristics describe the static sequencing problem captured in this model:

1. Order of precedence(i.e., order by which operations must follow each other),
2. Number of machines or processors,
3. Number of product types,
4. Process times with knowledge of precedence,
5. Set up times with knowledge of precedence,
6. Batch size,
7. Demand for each product,
8. Due date for each product,

9. Costs associated with the system prosess time, setup time, machines, lotsize,
10. Percentage of defectives from the system,
11. Type of facility - new or old,
12. Number of Process or Operations per Product.

3.2 OPERATIONAL RESTRICTIONS

The system is constrained by the following:-

1. Feasible sequences,
2. Amount of products carried between stages,
3. Investment cost,
4. Demand,
5. Time (in machine time),
6. Homogeneity of machines at a work station,
7. Each work station must have at least one machine.

The precedence restrictions on the processors for each product determine many of the systems other variables. Variables such as processing time and setup time depend on this sequential order for products and processors.

The number of processors in the system is important, since this will establish whether it is multistage or one stage .The case of parallel processors is a consequence of using the values of demand, sequence and due date to replicate the processors, with the aim of meeting demand and due dates.

Specifications of product types help establish product routes given the precedence restrictions on products and machines.The process and setup times are direct results of the sequence restrictions. These times are important to establish optimization criteria in the problem being considered. Lotsizes carried between stages influence other variables in the system. They can also influence in-process inventory. Demand and due dates are required to define more realistic capacity requirements.

Before proceeding further, certain nomenclature must be listed and defined.

3.3 NOMENCLATURE

i = product,

j = operation,

n_{ij} = number of machines performing operation j for product i,

$v_{i,j}$ = operation time for product i on operation j,

$k_{i,j}$ = set up time for product i, operation j,

Ui,j = quantity (unit load) moved from operation j to operation j+1 considering product i (Also run quantity/set up),

s_i = state of product in the system(i.e.,which operations have been performed on the product),

$C_{i,j}$ = cost associated with setup time for product i,operation j,

$Cv_{i,j}$ = cost associated with processing product i, operation j, for a unit of time $v_{i,j}$

$F_{i,j}$ = fixed cost incurred by using each n_{ij} machine,

$T_{i,j}$ = cost for transporting a given batch of product i after opration j,

I = Investment money available at the beginning of planning period,

Wmax = maximum weight which can be transported between work stations,

H = Amount of time avalilable for meeting demand D,

$f_{i,j} \langle s_i \; D_i \rangle$ = frequency of moves with a given demand for product type i through work center j,

$\delta_{i,j}$ = percentage of defective at work station j for product i,

C_{nj} = Cost of one machine which can perform operation type j,

k_j = the maximum setup time allowed at station j,

v_j = the maximum process time allowed for station j.

Modelling the problem with a cost objective function and with a time objective function would be advantageous since some components of the cost function cannot be captured in the time function. Certain fixed costs exist due to the system and there is no way a time function can capture such system control component. The general form of the objective functions are as follows:

3.4 COST FUNCTION:

Min:

PROCESSING COST + SETUP COST + FIXED COST

+MATERIAL HANDLING COST (3.1)

3.5 TIME FUNCTION:

Min:

TOTAL PROCESSING TIME + TOTAL SETUP TIME (3.2)

3.6 ASSUMPTIONS

Before these two objective functions are defined more specifically, the assumptions which govern the system must be stated:

1. The technological order of all the products through all the processes is known in advance.

2. The technological order may be different for each product.

3. The processing times and setup times for each product at a given machine are deterministic and known, assuming knowledge of precedences.

4. Pre-empting is not allowed.

5. All jobs are of equal importance [27].

6. All jobs are processed as soon as possible.

6a No scheduling conflict exists.

7. All N jobs are simultaneously available at the beginning of each planning period.

8. All M machines are available at the beginning of the planning period and are ready to take on any of the N jobs.

9. A machine can process only one job at a time.

10. parallel processing is allowed

11. In-process inventory is allowed.

12. Time saved on any machine can be used for some other useful and profitable work.

3.7 OBJECTIVE

The objective to be achieved is either to minimize total elapsed time from the start of first job to the completion of last job or to minimize the total cost of production per period.

3.8 MODEL COMPONENTS:

(1) Let $C_{i,j}$ be the cost of processing one item of product type i through operation j for one unit of process time $v_{i,j}$. . Then the total cost must be $C_{i,j}\ V_{i,j} <s_i, D_i>$ for this one item. The process time $v_{i,j}$ is a function of the state of the product i in the system, given demand D_i :

hence, $v_{i,j} <s_i \quad D_i>$. For demand D_i therefore, the cost will be:

$$D_i C_{i,j}\ V_{i,j} <s_i, D_i>. \tag{3.3}$$

(2) Setup for product i on machine $n_{i,j}$ is on a per batch basis. If $C_{i,j}$ is the cost associated with setting up machine $n_{i,j}$ once for product i, then, the cost of setting up for n machines will be:

$$C_{i,j}\, n_{i,j}\, f_{i,j} <s_i, D_i> \tag{3.4}$$

Notice that frequency is also a function of state of product i for a given demand D_i for product i.

(3) There is always a fixed cost due to using machine $n_{i,j}$ at the work station j. This cost is:

$$F_{i,j}\, n_{i,j}\,. \tag{3.5}$$

(4) In order to move product i between work stations, a certain cost is incurred due to material handling. This cost $T_{i,j}$ is cost per move. For $f_{i,j}$ moves then, the cost will be:

$$T_{i,j}\, f_{i,j} \langle s_i, D_i \rangle. \qquad (3.6)$$

All the above four components of the objective function are summed over all products and for all operations performed on a given product.

COST OBJECTIVE FUNCTION:

$$\Phi = \mathrm{Min} \sum \sum [K_{i,j} \langle s_i, D_i \rangle + C_{i,j}\, n_{i,j}\, f_{i,j} \langle s_i, D_i \rangle + F_{i,j}\, n_{i,j} + T_{i,j}\, f_{i,j} \langle s_i, D_i \rangle] \qquad (3.7)$$

where

$$K_{i,j} \langle s_i, D_i \rangle = D_i C_{v_{ij}} V_{i,j} \langle s_i, D_i \rangle$$

CONSTRAINTS

(1) The batch size at the last work station must be less than or equal to demand:

$$U_{i,0} \leq D_i \tag{3.8}$$

(2) For each product i, the quantity carried through operation j for all the processes at demand D_i must equal the quantity produced:

$$U_{i,j}\, f_{i,j} = U_{i,0}\, f_{i,0} \tag{3.9}$$

(3) At a work station, there must be at least one machine:

$$\sum_{i} n_{i,j} \geq 1 \tag{3.10}$$

(4) The frequency of moves through the last work station must be integer for product i:

$$f_{i,0} = Z_i \qquad \text{where } Z_i \text{ is integer} \tag{3.11}$$

(5) The total number of products for a given production period must equal total demand D_i:

$$U_{i,0}\, f_{i,0} = D_i \qquad U_{i,j} \in R \tag{3.12}$$

(6) The weight of each batch moved between work stations must be less than or equal to a given maximum weight for every operation j:

$$w_i D_i \leq Wmax\ f_{i,j} \qquad \forall j \qquad (3.13)$$

(7) The frequency of moves at a work station is dependent on the percentage of defectives at that work station for product i:

$$f_{i,j} = f_{i,0} \left(\prod_{\ell=1}^{j-1} (1- \delta_{1,\ell} <s_i>)\right)$$
(3.14)

(8) Cost of purchasing all machines for the entire system must be less than or equal to the available investment capital:

$$\sum_j C_{nj} \sum_i n_{i,j} \leq I \qquad \forall j \qquad (3.15)$$

(9) The total time utilized in setting up and processing product i at work station j must be less than or equal to the machine time specified for processing demand D_i without violating due date:

$$v_{i,j} D_i \Big/ \prod_{\ell=1}^{j-1} (1- \delta_{i\ell}) + f_{i,j} k_{i,j} n_{i,j} \leq Hn_{i,j} \qquad \forall_{i,j} \qquad (3.16)$$

(10) The frequency of moves, the machines available at work station j, and the batch moved between work stations must be greater than or equal to one:

$$f_{i,j} \quad n_{i,j} \quad U_{i,j} \geq 1 \tag{3.17}$$

Time Model: The aim here is to find the shortest time for given products to be processed through the system. Process time, as in the cost model, is in time/item. To find the process time of all products in the batch, then one obtains (Vij)(Ui,j). But parallel processing could be going on in order to meet due dates and demand. So, the number of machines performing this type of job will be involved in this objective function as follows:

$$((vij)(Ui,j))/n_{i,j} \tag{3.18}$$

The setup time is in time/batch. As a result, the setup time is $k_{i,j}$. But, many machines could be performing the same kind of operation for the product batch. The function is therefore:

$$k_{i,j}/n_{i,j} \tag{3.19}$$

However these times are only for one batch size: the general time objective function is as follows:

$$\sum_i \sum_j f_{ij} \; [\; ((v_{i,j})(Ui,j))/n_{i,j} + \; (k_{i,j}/n_{i,j})\,] \tag{3.20}$$

The following constraints apply to the time objective function:

$$0 \leq U_{i,j} \leq D_i \tag{3.21}$$

$$n_{i,j} \leq n_j \quad (3.22)$$

$$k_{i,j} \leq k_j \quad (3.23)$$

$$v_{i,j} \leq V_j \quad (3.24)$$

$$D_i = f_{i,0} U_{i,0} \quad (3.25)$$

$$w_i D_i \leq \text{Wmax } f_{i,j} \quad (3.26)$$

$$f_{i,0} = Z_i \quad \text{where } Z_i \text{ is integer.} \quad (3.27)$$

$$f_{i,j} = f_{i,0}\left(\prod_{\ell=1}^{j-1} (1-\delta_{i\ell} \quad \langle s_i \rangle)\right) \quad (3.28)$$

$$\sum_i n_{i,j} \geq 1 \quad (3.29)$$

$$f_{i,j}, \quad n_{i,j}, \quad U_{i,j} \geq 1 \quad (3.30)$$

3.9 ANALYSIS OBJECTIVE

The ultimate objective of this research is to capture the sequence selection problem in the above models; and further, to investigate solution procedures to the cost model shown. Problems of various sizes (i.e., M and N) are solved. Their execution times are compaired.

There are four cases of the model for the type of system described in this chapter.

I. Single product - fixed sequence: In this case there is only one product to be produced and there is a known sequence of operations that the product must go through. Furthermore, there is only one of this kind of sequence. A planner would not have to invoke a sequencing algorithm to set up the sequence since the system is fixed.

II. Multiple product - fixed sequence: As in the case of single product - fixed sequence, the system is also fixed and would not require a sequencing algorithm ; however, a capacity planning problem is involved (see refences[2,21,24,34,35,47]).

III. Single Product - variable sequence: The product can be processed through two or more possible sequences of operations. The fact that choice exists here brings into consideration such aspects as cost of using a given sequence (processing, investment, setup, transportation, fixed cost), the makespan for the sequence considered, the quality of the product after processing through this given sequence, the machine requirements of that sequence at various work stations, the unit load, etc..

Because of these problems, a planner would want to select a sequence which is optimal, given all the above requirements and restrictions. The variable sequence problem is therefore a complex problem with many restrictions.

IV. Multiple product - variable sequence:

As in the single product-variable sequence case, the multiple product-variable sequence case takes on far more complexity than the fixed sequence cases. The multiple product - variable sequence case has all the qualities of the single product-variable sequence plus a combinatorial problem arising from the selection of an overall optimal sequence for each product. Since each product has many possible sequence selections, the problem is one of selecting which sequence will combine with sequences from other products to yield optimal results.

An example problem which illustriates this the general case is shown in Appendix I. This problem will be used for illustriative purposes in the problem analysis of chapters IV and V, with slight modifications for cases I, II and III.

It will be necessary to now define two terms which will be used throughout the rest of this thesis. Product line and flowline mean the same thing in industrial systems. However the nature of the sequence selection problems considered in this research provides a reason for the separation of these terms. One would therefore define product line as the system whereby the products do not compete for the available machines in a work station. In other words it is completely serial in nature. This would be different to the system where the products compete for available machines in a work station. This could be viewed as a combination of serial and parallel systems.

Chapter IV

METHOD OF PROBLEM ANALYSIS AND SOLUTION OF FIXED SEQUENCE CASES

In Chapter III, the model that describes the sequence selection problem was developed and described. The four cases which depict the sequence selection problem were introduced. In this chapter, the fixed sequence cases will be examined.

It was pointed out in the last chapter that the fixed sequence problem involves only one type of ordering of the operations. As a result, the problem here is only the calculation of the objective function given that all the constraints are obeyed. The problem would not require computer coding since it can be solved by inspection. In the single product fixed sequence case, the problem is especially easy because the linear constraints in n are only one variable constraints.

One may recognize the model as follows: Let the objective function be represented by:

$$\Phi = \sum_i \sum_j [K_{i,j} < s_i, D_i > + C_{ij} n_{i,j} f_{i,j} < s_i , D_i > + F_{i,j} n_{i,j} + F_{i,j} f_{i,j} < s_i D_i >] \quad (4.1)$$

subject to:

$$U_{i,0} \le D_i \tag{4.2}$$

$$U_{i,j} f_i = U_{i,0} f_{i,0} \tag{4.3}$$

$$\sum_i n_{i,j} \ge 1 \tag{4.4}$$

$$f_{i,0} = Z_i \quad \text{where} \quad Z_i \text{ is integer} \tag{4.5}$$

$$U_{i,0} f_{i,0} = D_i \qquad U_{i,j} \in R \tag{4.6}$$

$$w_i D_i \le W_{max} f_{i,j} \quad \forall j \tag{4.7}$$

$$f_{i,j} = f_{i,0} \left(\prod_{\ell=1}^{j-1} (1 - \delta_{i,\ell} \langle s \rangle) \right) \tag{4.8}$$

$$\sum_j C_{nj} \sum_i n_{i,j} \le I \quad \forall j \tag{4.9}$$

$$V_{i,j} D_i / \prod_{\ell=1}^{j-1} (1 - \delta_{i,\ell} \langle s_i \rangle) + f_{i,j} k_{i,j} n_{i,j} \le H n_{i,j} \quad \forall j \tag{4.10}$$

$$f_{i,j},\ \sum_i n_{i,j},\ U_{i,j} \ge 1 \tag{4.11}$$

4.1 SOLUTION PROCEDURE

Before proceeding to the individual cases of the sequence selection problem, several characteristics will be presented.

1. CALCULATION OF FREQUENCY OF MOVES

The first component to be examined in the solution of the problem is constraint equation 4.8:

$$f_{i,j} = f_i \quad \left(\prod_{i=1}^{j-1} (1 - \delta_{i,\ell} <s_i>) \right)$$

The only value given in the above relation is the defectives at various work stations. An objective is to have an integer value for the frequency of moves. The frequency of moves at each work station can then be calculated in terms of the frequency of moves at the final work station.

The material handling restriction (equation 4.7):

$$w_i \; D_i \leq W_{max} \, f_{i,j} \qquad \forall \; j$$

is now used to calculate the lower bounds on the frequency of moves at the final work center. It is important to note here that this lower bound on frequency is optimal. This is because any increase in the value of the frequency will increase the objective function. Furthermore, any increase in the value of the frequency of moves will tighten the controlling constraint (4.10):

$$v_{i,j} D_i / \prod_{\ell=1}^{j-1} (1 - \delta_{i,\ell} <s_i>) + f_{i,j} \, k_{i,j} \, n_{i,j} \leq Hn_{i,j} \qquad \forall j$$

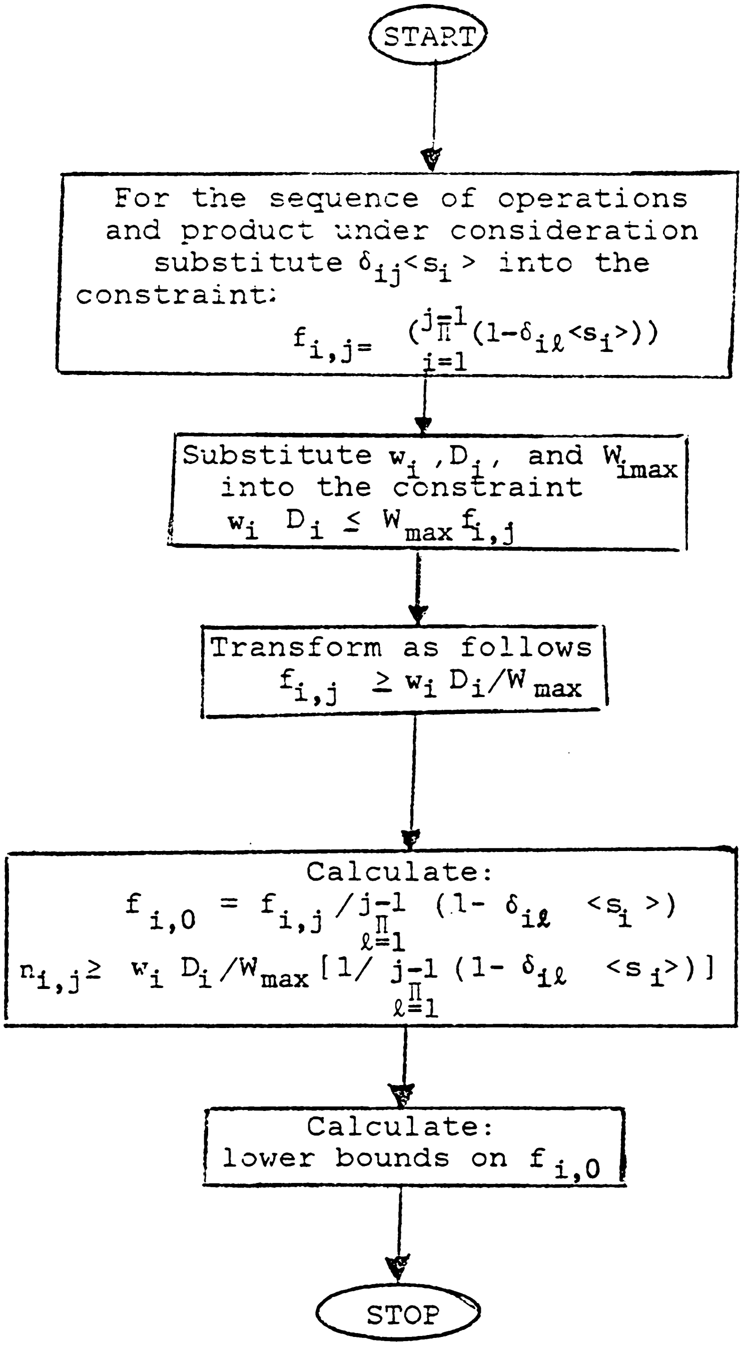


Figure 2: Procedure for Calculation of Frequency

The frequency at the final work station is obtained from the relation for $f_{i,j}$ and $f_{i,0}$ above. The frequency $f_{i,0}$ is now rounded up to the first feasible integer value.

The following flowchart(Figure 2) summarizes the procedure for obtaining lower bounded frequency of moves for each product:

2. CALCULATION OF MACHINE REQUIREMENTS

The frequency of moves at the last work station for each product has now been calculated. The next controlling constraint to be examined involves the machine time constraint(4.10):

$$v_{i,j} D_i / \prod_{\ell=1}^{j-1} (1 - \delta_{i,\ell} < s_i > + f_{i,j} k_{i,j} n_{i,j} \leq Hn_{i,j} \quad \forall j$$

The fact that the above constraint is to be examined for every work station j makes the problem easier to handle. All the given values along with $f_{i,j}$ calculated above are now substituted into this constraint. One discovers that this constraint is linear in $n_{i,j}$. Some more simplication is done to solve the problem for $n_{i,j}$. One obtains the following:

$$(H - f_{i,j} k_{i,j}) n_{i,j} \geq v_{i,j} D_i / \prod_{\ell=1}^{j-1} (1 - \delta_{i,\ell} < s_i >)$$

$$n_{i,j} \geq v_{i,j} D_i / \prod_{\ell=1}^{j-1} (1-\delta_{i,\ell} \ \langle s_i \rangle)[1/H - f_{i,j} \, k_{i,j}]$$

Lower bound on $n_{i,j}$ can now be calculated. It is also important to note here that the lower bound obtained is optimal for the problem since any increase in the value of $n_{i,j}$ will worsen the objective function. This is because the objective function is a monotone non-decreasing function. Also any increase in the value of $n_{i,j}$ will tighten the investment cost constraint:

$$\sum C_{nj} \sum n_{ij} \leq I$$

The following flowchart (Figure 3) summarizes the calculation of n

It is necessary to address here the problem of calculating $n_{i,j}$ for each work station. Since there can be more than one i (i≥ 1), one may wish to have a homogenous work station with all products being able to use any of the machines at a given work station. This means that all machines required at work station j can be summed together and the integer value taken. For example, if $n_{1,1} = 2.10$ and $n_{2,1} = 3.07$ for a two-product system, then

$$\begin{aligned}\sum_i n_{i,j} &= n_{1,1} + n_{2,1} \\ &= 2.10 + 3.07\end{aligned}$$

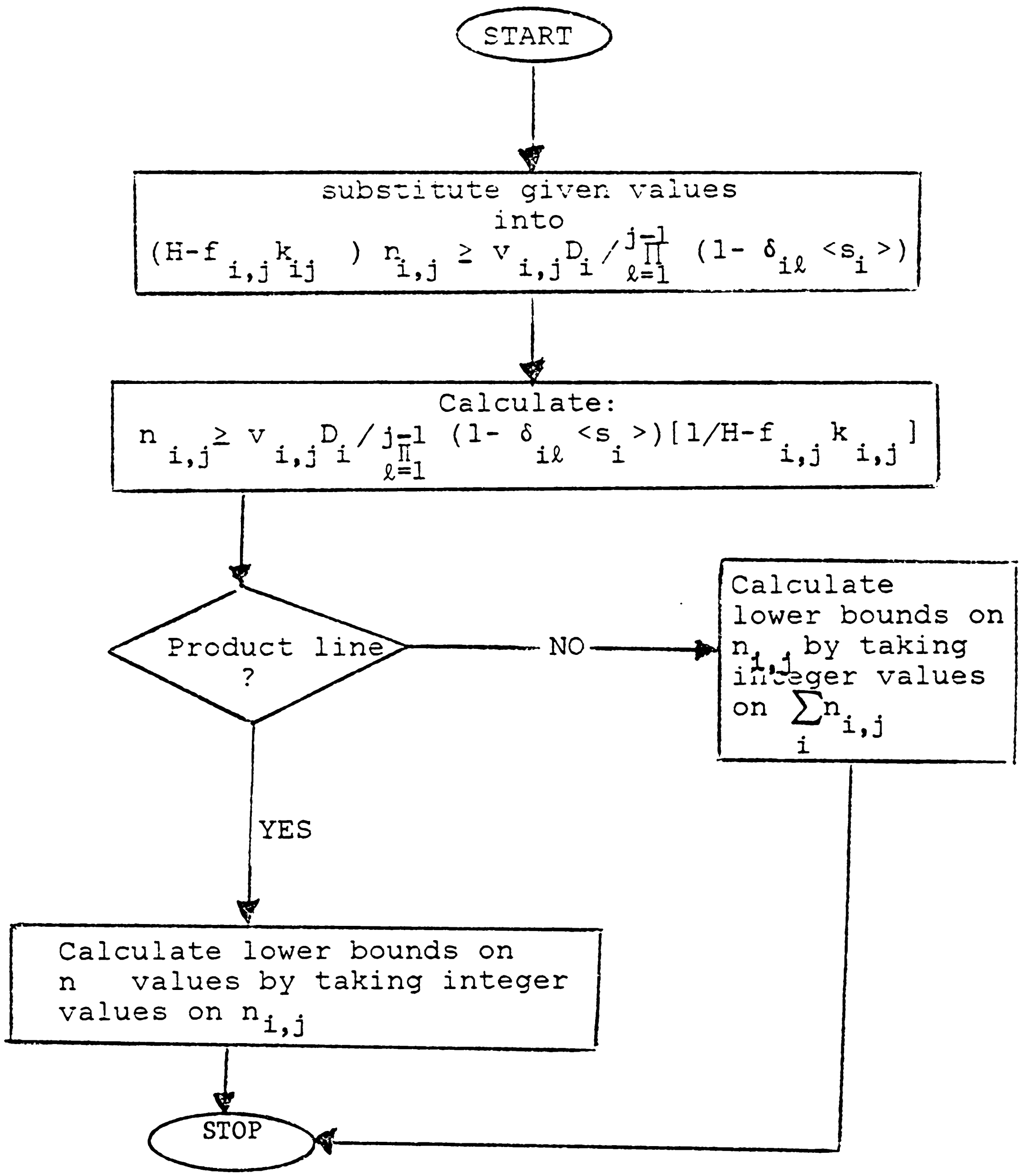


Figure 3: Calculation of Machine Requirements

$$= 5.17 \quad \text{machines}$$

The feasible integer solution is then 6 machines. However, if one wants a product line system, where each product has its own machines and would not have to be processed on the machines used by the other products, then the machine requirements for each product will have to be rounded to integer values. The two problem example above will then have $n_{1,1} = 3$ and $n_{2,1} = 4$. Then

$$\sum_i n_{i,j} = n_{1,1} + n_{2,1}$$
$$= 7$$

One can see that the last case requires an additional machine thus increasing the objective cost and tightening the capital investment constraint. It can safely be said, therefore, that structuring the work center in this fashion will not be most beneficial to the planner. Of course, introduced by having all the products share the machines at a work station, the opportunity for scheduling problems exists.

The investment constraint must be examined to determine if it will be possible to purchase all the machines required. If the machines cost more than the capital investment constraint, then the system cannot be implemented due to lack of funds. The other constraints are now checked for conformity. If they are not violated, then, the cost of production/work period is calculated and this cost (ϕ) is optimal. From the above discussions, one can see that the system is demand-driven. The investment cost and machine time are critical constraints while the frequency of moves and the machines required at each work station constitute the control variables.

4.2 COMMON FEATURES

Certain features are common to all four cases of the sequence selection problem: the procedure for calculating the lower bounded frequency of moves between work stations, and the calculation of the machine requirements at the same work station. These common features make the solution of these problems more tractable. The solution of multiple product, variable sequence problems will be combinatorial in nature; but the calculations are similar to those of the fixed-sequence problem. With the above background, one may now look at the specific cases of fixed sequence problems.

4.3 SINGLE PRODUCT - FIXED SEQUENCE

The following is a summary of the procedure used in the solution of the single product, fixed sequence problem. A flowchart is also presented in Figure 4.

The material handling restriction is used to find the lower bound on $f_{i,0}$. This bound is substituted into the machine time constraint to obtain lower bounds on $n_{i,j}$. These lower bounds are substituted into the investment constraint to check for its violation along with other constraints. The objective function is then calculated and this should be optimal. The example problem used here is a modification of the problem in Appendix 1. The procedure is illustrated here using sequence ADBE. Some of the transformation steps are trivial and will be skipped.

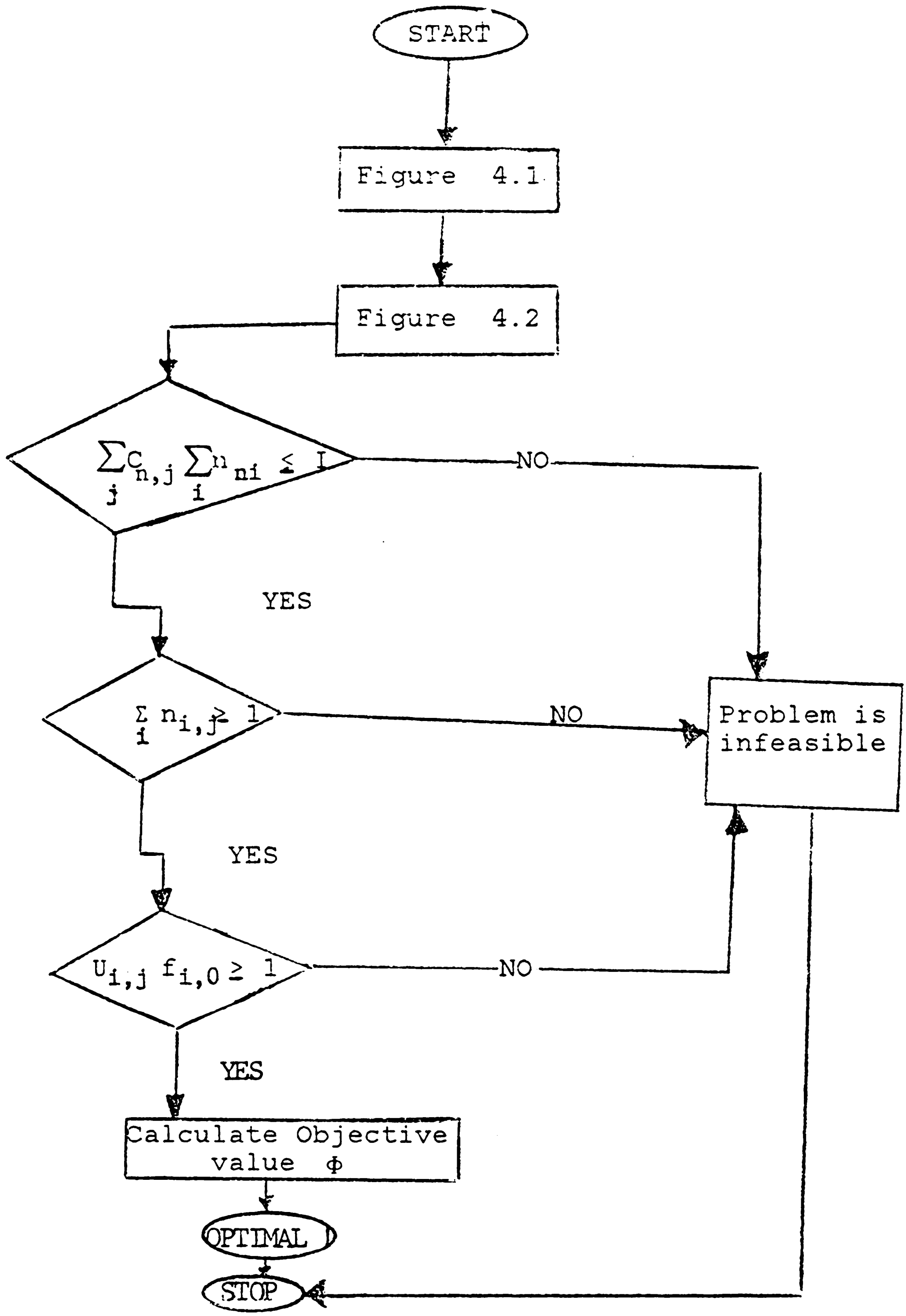


Figure 4: Single Product - Fixed Sequence

Consider the sequence ADBE for a single product. There are 4 stations and the station numbering is backwards. This means that station 4 is the first work station. The objective function is as follows:

GENERAL SUBSTITUTIONS:

$$\Phi = 1675 + (0.04905638n_{1,4} + 0.07880543n_{1,3} + 0.0987042n_{1,2} + 0.05964n_{1,4})f_{i,0}$$

$$+ 200(n_{1,4} + n_{1,3} + n_{1,2} n_{1,1})$$

$$+ 2.0732162f_{i,0}$$

$$f_{i,j} = f_{i,0}\left(\prod_{\ell=1}^{j-1}(1-\delta_{i\ell}(s_i))\right)$$

$$f_{1,1} = 0.994f_{1,0}$$

$$f_{1,2} = 0.987042f_{1,0}$$

$$f_{1,3} = 0.985068f_{1,0}$$

$$f_{1,4} = 0.981128f_{1,0}$$

$$f_{i,0} U_{i,0} = D_i$$

$$f_{i,0} U_{i,0} = 1000$$

$$f_{i,0} = Z \quad \text{where } Z \text{ integer}$$

$$f_{1,0} = Z_i$$

$$U_{i,j} f_{i,j} = U_{i,0} f_{i,0}$$

$$U_{1,1} f_{1,1} = U_{1,0} f_{1,0}$$

$$U_{1,2} f_{1,2} = U_{1,0} f_{1,0}$$

$$U_{1,3} f_{1,3} = U_{1,0} f_{1,0}$$

$$U_{1,4} f_{1,4} = U_{1,0} f_{1,0}$$

$$w_i D_i < W_{max} f_{i,j}$$

$$W_{max} f_{i,j} \geq w_i D_i \quad \forall j$$

$$100(.994) f_{1,0} \geq 2(1000)$$

$$100(.987042) f_{1,0} \geq 2(1000)$$

$$100(.985068) f_{1,0} \geq 2(1000)$$

$$100(.981128) f_{1,0} \geq 2(1000)$$

$$f_{i,0} \leq D_i$$

$$f_{1,0} \leq 1000$$

$$f_{1,0} \geq 1$$

$$\sum_j C_{nj} \left(\sum_i n_{ij} \right) \leq I$$

$$50n_{1,1} + 50n_{1,2} + 50n_{1,3} + 50n_{1,4} \leq 3000/3$$

$$\sum_i [v_{i,j} D_i / ((\prod_{\ell=1}^{j-1} (1-\delta_{i\ell} \langle s_i \rangle)) + f_{i,j} k_{i,j} n_{i,j}] \leq H\, n_{i,j}$$

$$1/\prod_{\ell=1}^{j-1} (1-\delta_{i,\ell})(2500) + 9.94 f_{1,0}\, n_{i,1} \leq 1080 n_{1,1} \quad \forall j$$

$$1/\prod_{\ell=1}^{j-1} (1-\delta_{i,\ell}) \quad (1000) + 8.83378 f_{1,0} n_{1,2} \leq 1080 n_{1,2}$$

$$1/\prod_{\ell=1}^{j-1} (1-\delta_{i,\ell}) \quad (3000) + 6.895476 f_{1,0}\, n_{1,3} \leq 1080 n_{1,3}$$

$$1/\prod_{\ell=1}^{j-1} (1-\delta_{i,\ell}) \quad (1500) + 7.849024 f_{1,0}\, n_{1,4} \leq 1080 n_{1,4}$$

$$f_{1,4} = 0.981128 f_{1,0}$$

$$\sum n_{i,j}, \quad U_{i,j} f_{i,j} \geq 1$$

For this single product, and $f^*_{1,0} = 21$. Further transformation renders the model as follows:

$$\Phi = 201.03018398n_{1,4} + 201.65491403n_{1,3} + 202.0727882n_{1,2} + 201.25244n_{1,1} + 1718.53754$$

$$f_{1,1} = 20.874$$

$$f_{1,2} = 20.727882$$

$$f_{1,3} = 20.686428$$

$$f_{1,4} = 20.603688$$

$$U_{1,0} = 47.61904764$$

$$\text{All } f_{i,j} = Z_1 = 21$$

$$U_{i,j} f_{i,j} = 1000$$

$$100f_{i,j} \geq 2000$$

$$f_{1,0} \leq 1000$$

$$50[n_{1,1} + n_{1,2} + n_{1,3} + n_{1,4}] \leq 3000$$

$$(1/0.994)(2500) + (9.94n_{1,1})f_{1,0} \leq 1080n_{1,1}$$

$$(1/0.987042)(1000) + (8.883378n_{1,2})f_{1,0} \leq 1080n_{1,2}$$

$$(1/0.985068)(3000) + (6.895476n_{1,3})f_{1,0} \le 1080n_{1,3}$$

$$(1/0.981128)(1500) + (7.849024n_{1,4})f_{1,0} \le 1080n_{1,5}$$

$$n_{i,j} \quad , \quad U_{i,j} \quad , \quad f_{i,j} \ge 1$$

$$\sum n_{i,j} f_{1,0} = Z_i$$

$$\Phi = \text{Min:} \quad 201.03018398n_{1,4} + 201.65491403n_{1,3} + 202.0727882n_{1,2} + 201.25244n_{1,1} + 1718.53754$$

$$f_{1,0} = 21$$

$$f_{1,1} = 20.874$$

$$f_{1,2} = 20.727882$$

$$f_{1,3} = 20.686428$$

$$f_{1,4} = 20.603688$$

$$U_{1,0} = 47.61904768$$

$$\text{All } f_{i,j} = Z_i = 21$$

$$U_{i,j} f_{i,j} = 1000$$

$$f_{1,0} \ge 20$$

$$f_{1,0} \leq 1000$$

$$n_{1,1} + n_{1,2} + n_{1,3} + n_{1,4} \leq 60/3$$

$$n_{1,1} \geq 2.8867279$$

$$n_{1,2} \geq 2.8348793$$

$$n_{1,3} \geq 2.7138153$$

$$n_{1,4} \geq 2.7842763$$

$$\sum_j n_{1,j} \geq 11.219699$$

$$f_{i,j} \quad U_{i,j} \quad \sum n_{i,j} \geq 1$$

The minimum value $n_{i,j}$ can assume is 3 without violating any of the constraints as well as the integer constraint for each i and j since each $n_{i,j}$ is greater than 2 but less than 3.

Hence $\Phi = \$4136.57$

The calculation of the objective value is done using the product-line argument discussed before. The objective value would be less if the products were allowed to share the same machines at a work station, i.e.,

$$\sum_j n_{i,j} = 12.$$

4.4 MULTIPLE PRODUCT - FIXED SEQUENCE(CASE II)

From the example problem, consider products 1,2, and 3 with fixed sequences ADBE, ABCE, and ACDE respectively. Then, proceeding as in single product case, all the lower bounds $f_{i,0}$ are calculated as follows:

$$100f_{1,4} \geq 2(1000)$$

$$150f_{2,4} \geq 3(2000)$$

$$125f_{3,4} \geq 2.5(1500)$$

So that:

$$(100)0.981128f_{1,0} \geq 2000 \text{ or } f_{1,0} \geq 20.39$$

$$(150)0.9762003f_{2,0} \geq 6000 \text{ or } f_{2,0} \geq 40.98$$

$$(125)0.981135f_{3,0} \geq 3750 \text{ or } f_{3,0} \geq 30.58$$

From the above result it means that:

$$f_{1,0} \geq 21$$

$$f_{2,0} \geq 41$$

$$f_{3,0} \geq 31$$

Taking the lower bound values of $f_{1,0} = 21$, $f_{2,0} = 41$, and $f_{3,0} = 31$, then the following time constraints are obtained:

$$(1/.994)(2.5)(1000) + (10)(.994)f_{1,0}n_{1,1} \leq 1080n_{1,1}$$

$$(1/.987042)(1)(1000) + (9)(.987042)f_{1,0}\,n_{1,2} \leq 1080n_{1,2}$$

$$(1/.985068)(3)(1000) + (7)(.985068)f_{1,0}\,n_{1,3} \leq 1080n_{1,3}$$

$$(1/.981128)(1.5)(1000) + (8)(.981128)f_{1,0}\,n_{1,4} \leq 1080n_{1,4}$$

$$(1/.993)(2)(2000) + (6)(.993)f_{2,0}\,n_{2,1} \leq 1080n_{2,1}$$

$$(1/.990021)(2.3)(2000) + (5)(.990021)f_{2,0}\,n_{2,2} \leq 1080n_{2,2}$$

$$(1/.980121)(2.5)(2000) + (4)(.980121)f_{2,0}\,n_{2,3} \leq 1080n_{2,3}$$

$$(1/.9762003)(1.5)(2000) + (6)(.9762003)f_{2,0}\,n_{2,4} \leq 1080n_{2,4}$$

$$(1/.995)(1.5)(1500) + (6)(.995)f_{3,0}\,n_{3,1} \leq 1080n_{3,1}$$

$$(1/.99025)(3.1)(1500) + (4)(.99025)f_{3,0}\,n_{3,2} \leq 1080n_{3,2}$$

$$(1/.985075)(2.0)(1500) + (4)(.985075)f_{3,0}\,n_{3,3} \leq 1080n_{3,3}$$

$$(1/.981135)(1.5)(1500) + (7)(.981135)f_{3,0}\,n_{3,4} \leq 1080n_{3,4}$$

$$\sum_{i} n_{i,j} \geq 1$$

After substitution of lower bound $f_{i,0}$ and solution of the resulting LP in $n_{i,j}$ the following results are obtained:

$$871.26n_{1,1} \geq 2500 \text{ or } n_{1,1} \geq 2.869$$

$$893.45n_{1,2} \geq 1000 \text{ or } n_{1,2} \geq 1.119$$

$$935.195n_{1,3} \geq 3000 \text{ or } n_{1,3} \geq 3.208$$

$$915.171n_{1,4} \geq 1500 \text{ or } n_{1,4} \geq 1.639$$

$$835.722n_{2,1} \geq 4000 \text{ or } n_{2,1} \geq 4.786$$

$$871.046n_{2,2} \geq 4600 \text{ or } n_{2,2} \geq 5.281$$

$$919.260n_{2,3} \geq 5000 \text{ or } n_{2,3} \geq 5.439$$

$$839.855n_{2,4} \geq 3000 \text{ or } n_{2,4} \geq 3.572$$

$$894.930n_{3,1} \geq 2250 \text{ or } n_{3,1} \geq 2.514$$

$$957.209n_{3,2} \geq 4650 \text{ or } n_{3,2} \geq 4.858$$

$$957.851n_{3,3} \geq 3000 \text{ or } n_{3,3} \geq 3.132$$

$$867.094n_{3,4} \geq 2250 \text{ or } n_{3,4} \geq 2.595$$

$$\sum_i n_{i,1} \geq 10.169$$

$$\sum_i n_{i,2} \geq 11.258$$

$$\sum_i n_{i,3} \geq 11.779$$

$$\sum_i n_{i,4} \geq 7.806$$

$$\sum_i \sum_j n_{i,j} \geq 41.012$$

Taking lower bound integer values for the $n_{i,j}$, the following are obtained:

$$n_{1,1} = 3$$

$$n_{1,2} = 2$$

$$n_{1,3} = 4$$

$$n_{1,4} = 2$$

$$n_{2,1} = 5$$

$$n_{2,2} = 6$$

$$n_{2,3} = 6$$

$$n_{2,4} = 4$$

$$n_{3,1} = 3$$

$$n_{3,2} = 5$$

$$n_{3,3} = 4$$

$$n_{3,4} = 3$$

$$\sum_i n_{i,1} = 11$$

$$\sum_i n_{i,2} = 13$$

$$\sum_i n_{i,3} = 14 \qquad \sum_i n_{i,4} = 9$$

$$\sum_i \sum_j n_{i,j} = 47$$

$$\Phi = 10242.5 + 16.58292 + 2200 + 43.53754$$

$$+ 59.026015 + 4620 + 84.771145$$

$$+ 29.416095 + 2250 + 52.013877$$

$$\Phi = \$\ 19{,}597.85$$

This value is the minimum cost for operating the system through the sequences enumerated for 18 hours.

4.5 SUMMARY

Thus far, the following procedures have been presented in this chapter:

1. Calculating the lower bounds on frequency of moves using the material handling constraint and substituting into the relation for $f_{i,0}$ and $f_{i,j}$.

2. Find the machine requirements by substituting given values along with the lower bound values for $f_{i,0}$ into the machine-time constraint.

3. Solve the fixed-sequence problem by checking the capital investment constraint and the other constraints for non-violation.

4. And finally, calculate the objective function.

Chapter V

VARIABLE SEQUENCE PROBLEMS

The main problem of this thesis is examined in this chapter. As mentioned in Chapter III, the variable sequence problem has many characteristics which make it complex and at the same time interesting. The combinatorial nature of this problem is first encountered at the capital investment constraint calculation:

$$\sum_j C_{n,j} \sum_i n_{i,j} \leq I \qquad \forall_j$$

It is easy to see that one must choose a sequence for each product in order to sum all the $n_{i,j}$ over all products. Each product may have more than one sequence and this introduces a combinatorial problem. Searching for the optimal sequence is necessary if one must consider feasibility requirements and the possible numerous objective values that go with these combinations.

The number of combinations build rapidly with greatest dependence on how many products one is considering. In fact, the number of possible sequence combinations is related to the product and number of sequences/product as follows: Combinations = S^N where S is the number of sequences/product and N is the number of products. One must realize that this combinatorial problem is significant because there is need to choose sequences for each product which will combine with other sequences to produce an optimal collective result. Considerations are given to the cost components, machine requirements of various combinations, the unit loads, etc.

In this chapter, an enumeration procedure is introduced for the selection of an optimal sequence combination.

5.1 ENUMERATION PROCEDURE

(1) Enumerate all possible sequence combinations for the product.

(2) For each product, calculate the frequency of moves at each work station (using Figure 4.2).

(3) For each sequence of operations for each product, calculate the machines required (using Figure4.3) to process that product at the various work stations.

(4) If a product line is under consideration, take integer values of the machine requirements for each product at each work station (i.e. $n_{i,j} \in Z_{i,j}$) where $Z_{i,j}$ is integer. Otherwise, sum all the required machines for all products at work station $j(\sum_i n_{i,j} \in Z_j)$ where Z_j is also integer.

(5) For each sequence combination, calculate for each product type and sequence of operations, the process cost, setup cost, material handling cost and fixed cost at work station j. Sum these costs as the cost of production at station j. Sum for all j. This is the total production cost for sequence combination q.

(6) Also for each sequence combination, calculate for each product type and sequence the investment cost. Sum these costs as investment cost for station j. Sum for all j. This is the investment cost of sequence combination q.

5.2 SEARCH PROCEDURE

When all these calculations are performed then the search proceeds as follows:

(1) Let CMIN be the minimum total cost.set initially at a very large value.

(2) Then for all feasible sequence combinations (q = 1,2,...S),

(a)(i) If CMIN is greater than the total cost of sequence q then check for feasibility by comparing the total investment cost associated with sequence q against the given capital investment potential.

(ii) If this sequence is feasible then the current total cost(CMIN) is set equal to the cost of sequence q.

(iii) If all sequences are considered then go to step 4; otherwise, let q=q+1 and return to (2.i).

(b) If CMIN is less than the total cost for sequence q, then go to the next sequence combination.

(3) If all sequences are infeasible, then go to step 6.

(4) Publish results showing the attributes of the optimal sequence q.

(5) Stop

(6) Problem is infeasible.

The following flowchart (Figure 5) summarizes the search for optimality after all sequence combinations are enumerated.

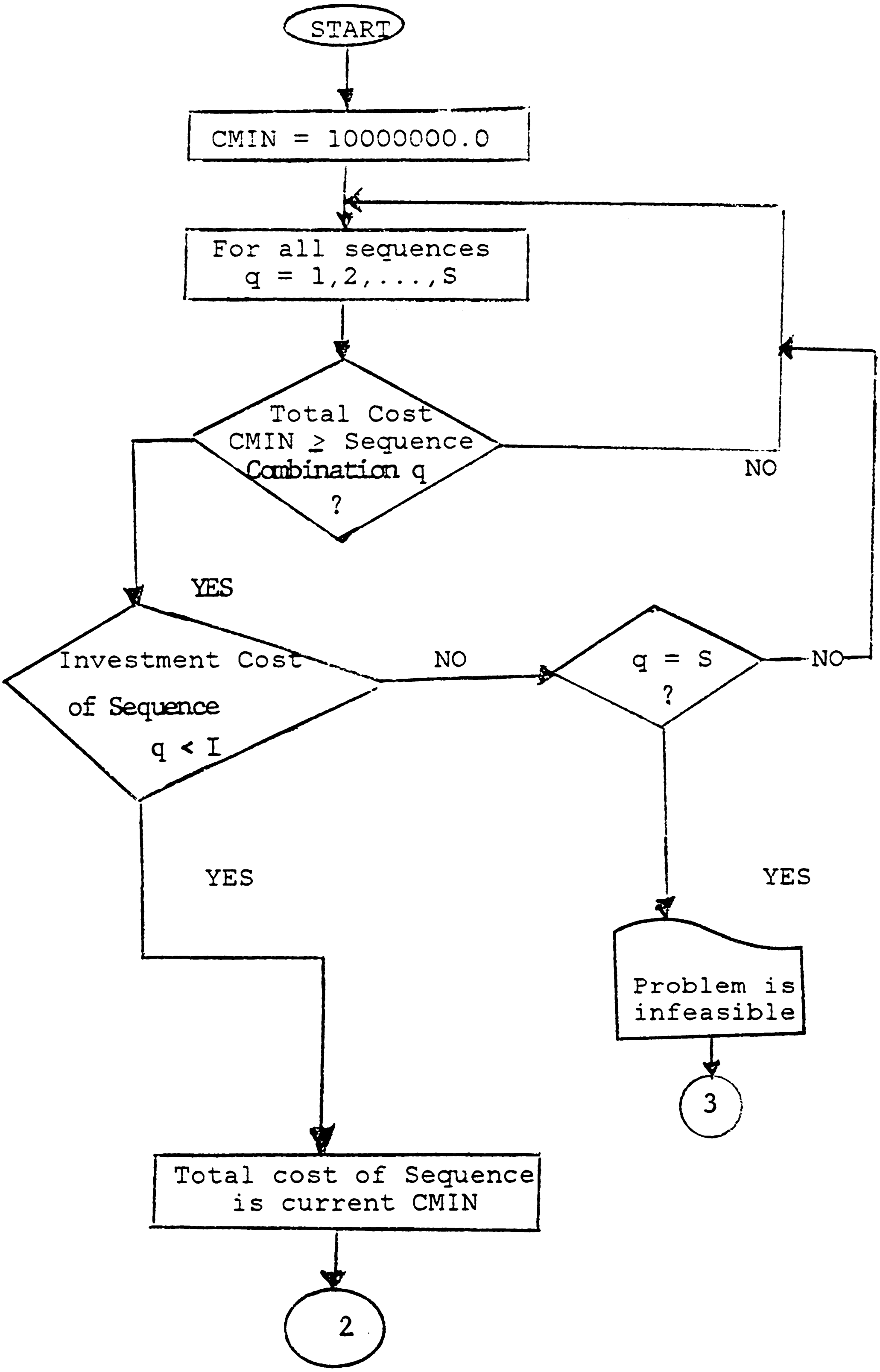


Figure 5: Search for Optimality Procedure

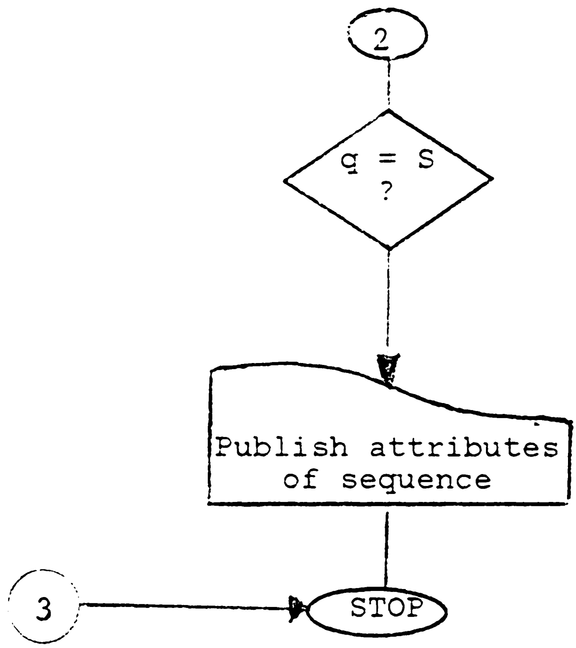


Figure 5: (continued)

5.3 DETERMINATION OF INTEGER MACHINES

The way in which integer values of machines are found from the required machines at each work station influences the objective value and more especially the investment cost. In many cases where the constraints are tight, the feasibility of the problem may depend totally on how these integer machines are obtained.

When the system is set up as a product line, then it will be necessary to take integer values on the number of machines at each work station ($n_{i,j} = Z_{i,j}$, where $Z_{i,j}$ is integer). There integer values are summed over all products and the sum is used for the investment cost calculation. However,if a homogenous work station is needed (i.e. a system where all products can share machines - a general flowline), then the machines required to process each product can be summed over all products and the resulting integer value used to calculate the cost. A comparison will show that the product-line case is more expensive and leads to infeasiblility of the problem in some cases while the flowline reduces costs including the investment cost, which determines whether a system can be implemented or not.

With this background and procedures for solving the sequence selection problem with variable sequences, it will serve to further clarify the procedures by solving cases III and IV of the sequence selection problem.

5.4 SINGLE PRODUCT - VARIABLE SEQUENCE (CASE III)

Using product 1 of the example problem, the following sequences are obtained: ADBE and ABDE

This problem is solved for a product line case. The first sequence (ADBE) has been treated earlier as a fixed sequence. Now consider sequence ABDE. The frequency $f_{1,0}$ is the same as in the previous sequence. So, $f^*_{1,0} = 21$. The machine constraint is as follows:

$$v_{i,j} D_i + k_{i,j} f_{i,j} n_{i,j} \leq H n_{i,j}$$

$$(.994)(1.5)(1000) + (8)(.994 f_{1,0}) n_{1,1} \leq 1080 n_{1,1}$$

$$(.98704)(2.5)(1000) + (10)(.98704 f_{1,0}) n_{1,2} \leq 1080 n_{1,2}$$

$$(.98507)(2.5)(1000) + (7)(.98507 f_{1,0}) n_{1,3} \leq 1080 n_{1,3}$$

$$(.981128)(1.5)(1000) + (11)(.981128 f_{1,0}) n_{1,4} \leq 1080 n_{1,4}$$

Solving for the values of $n_{i,j}$ then Obtain,

$$n_{1,1} \geq 1.643$$

$$n_{1,2} \geq 2.865$$

$$n_{1,3} \geq 2.673$$

$$n_{1,4} \geq 1.758$$

$$\sum_j n_{i,j} = 8.939$$

Taking integer bound values for $n_{i,j}$, the following is obtained:

$$n_{1,1} = 2$$

$$n_{1,2} = 3$$

$$n_{1,3} = 3$$

$$n_{1,4} = 2$$

$$\sum_j n_{i,j} = 10$$

The objective function for sequence ABDE is as follows:

$$\Phi_{ABDE} = 2175 + 0.05n_{1,4}f_{1,4} + 0.1n_{1,3}f_{1,3} + 0.08n_{1,2}f_{1,2}$$

$$+ 0.06n_{1,1}f_{1,1} + 200(n_{1,4} + n_{1,3} + n_{1,2} + n_{1,1})$$

$$+ 0.5f_{1,4} + 0.6f_{1,3} + 0.4f_{1,2} + 0.6f_{1,1}$$

$$= 2175 + 0.9811276n_{1,4} + 1.9701367n_{1,3} + 1.5792674n_{1,2}$$

$$+ 1.2n_{1,1} + 200(n_{1,4} + n_{1,3} + n_{1,2} + n_{1,1}) + 41.528433$$

$$\Phi_{ABDE} = 2175 + 401.96226 + 605.91039 + 604.73778 + 402.40$$

$$+ 41.528433$$

$= \$4321.54$

$\Phi_{ABDE} = \$4321.54 \qquad \sum_j n_{i,j} = 10$

$\Phi_{ADBE} = \$4136.57 \qquad \sum_j n_{i,j} = 12$

From the above result, it follows that sequence ADBE is optimal and will require a total of twelve machines to operate and meet demand as specified within the production time limit.

It is interesting to see that the system which produces optimal cost requires more machines than the non-optimal system. This is due to the other components of cost. For example, different process times are required for different sequences at the same work station (process time depends on the state of the product). This is why it is important to consider all the feasible sequences.

Now, consideration is given to CASE IV of the sequence selection problem. Since the procedure for solving this type of problem has been shown, the results presented here will not portray step by step calculations. The FORTRAN program for the calculations shown here can be found in Appendix II.

5.5 MULTI PRODUCT - VARIABLE SEQUENCE

The results apply to the complete example problem of Appendix I and are presented in Table 1. In this example, a general flowline is assumed.

TABLE 1

Summary of the Example Problem Solution Results

Seq. No.	No. of Mach. for station 1 2 3 4	Obj. Cost($)	Invst. Cost($)
1	8, 13, 14, 10	19631.83	2250.00
2	8, 14, 12, 11	19061.26	2250.00
3	8, 13, 13, 11	18812.98	2250.00
4	8, 14, 11, 12	18242.40	2250.00
5	8, 15, 14, 10	20567.50	2350.00
6	8, 15, 12, 11	19996.93	2300.00
7	8, 15, 13, 12	19748.65	2400.00
8	8, 15, 11, 13	19178.07	2350.00

The results obtained for the optimal sequence is presented below:

```
*******************************************************
 SEQUENCE  COMBINATION  4 RESULTS
PROC.COST FOR PROD. 1 OPER. 1=   375.000000
SETUP COST FOR PROD. 1 OPER. 1=    1.720988
TRANS.COST FOR PROD. 1 OPER. 1=   10.301830
FIXED COST FOR PROD. 1 OPER. 1=    334.113000
PROC.COST FOR PROD. 2 OPER. 1=   750.000000
SETUP COST FOR PROD. 2 OPER. 1=    5.859856
TRANS.COST FOR PROD. 2 OPER. 1=   24.038940
FIXED COST FOR PROD. 2 OPER. 1=    731.295100
PROC.COST FOR PROD. 3 OPER. 1=   562.500000
SETUP COST FOR PROD. 3 OPER. 1=    4.826466
TRANS.COST FOR PROD. 3 OPER. 1=   21.290610
FIXED COST FOR PROD. 3 OPER. 1=    528.953800
 MACHINES REQUIRED AT STATION 1=      8.000000
   COST AT STATION  1 =          3349.900000
PROC.COST FOR PROD. 1 OPER. 2=   300.000000
SETUP COST FOR PROD. 1 OPER. 2=    5.389246
TRANS.COST FOR PROD. 1 OPER. 2=    8.274569
FIXED COST FOR PROD. 1 OPER. 2=    651.302400
PROC.COST FOR PROD. 2 OPER. 2=  1599.999000
SETUP COST FOR PROD. 2 OPER. 2=   19.220120
TRANS.COST FOR PROD. 2 OPER. 2=   16.090310
```

```
FIXED COST FOR PROD. 2 OPER. 2=   1365.159000
PROC.COST FOR PROD. 3 OPER. 2=  1199.999000
SETUP COST FOR PROD. 3 OPER. 2=    7.767385
TRANS.COST FOR PROD. 3 OPER. 2=    9.161196
FIXED COST FOR PROD. 3 OPER. 2=    635.893000
 MACHINES REQUIRED AT STATION 2=     14.000000
   COST AT STATION  2 =          5818.253000
PROC.COST FOR PROD. 1 OPER. 3=   500.000000
SETUP COST FOR PROD. 1 OPER. 3=    2.350440
TRANS.COST FOR PROD. 1 OPER. 3=   12.436720
FIXED COST FOR PROD. 1 OPER. 3=    226.790200
PROC.COST FOR PROD. 2 OPER. 3=  1600.000000
SETUP COST FOR PROD. 2 OPER. 3=   13.873010
TRANS.COST FOR PROD. 2 OPER. 3=   20.213970
FIXED COST FOR PROD. 2 OPER. 3=    857.885000
PROC.COST FOR PROD. 3 OPER. 3=   465.000000
SETUP COST FOR PROD. 3 OPER. 3=    7.529474
TRANS.COST FOR PROD. 3 OPER. 3=    6.138153
FIXED COST FOR PROD. 3 OPER. 3=    981.334400
 MACHINES REQUIRED AT STATION 3=     11.000000
   COST AT STATION  3 =          4693.546000
PROC.COST FOR PROD. 1 OPER. 4=   499.999700
SETUP COST FOR PROD. 1 OPER. 4=    3.615450
TRANS.COST FOR PROD. 1 OPER. 4=   12.524390
FIXED COST FOR PROD. 1 OPER. 4=    577.345400
```

```
PROC.COST FOR PROD. 2 OPER. 4=   999.999700
SETUP COST FOR PROD. 2 OPER. 4=   20.636100
TRANS.COST FOR PROD. 2 OPER. 4=   24.452390
FIXED COST FOR PROD. 2 OPER. 4=   1265.895000
PROC.COST FOR PROD. 3 OPER. 4=   449.999700
SETUP COST FOR PROD. 3 OPER. 4=    5.455729
TRANS.COST FOR PROD. 3 OPER. 4=   15.422490
FIXED COST FOR PROD. 3 OPER. 4=    505.359100
 MACHINES REQUIRED AT STATION 4=     12.000000
   COST AT STATION  4 =          4380.703000
 TOTAL COST FOR SEQUENCE   4=        18242.400000
 TOTAL     INVESTMENT     COST     FOR     SEQUENCE        4=
2250.000000
*******************************************************
```

From these results one can see that the optimal sequence out of the 8 sequence combinations was sequence 4. The results indicate that the problem was feasible.

The example problem was modified to obtain a 4 product, 3 sequence problem. When this problem was solved,using the program in Appendix II,the problem was found to be infeasible for a product line but feasible for a general flowline. The investment cost calculated for all the possible sequence combinations was more than the capital investment cost.

This case goes to show the advantage of sequencing operations as a flowline. To gain appreciation for the combinatorial difficulties associated with obtaining solutions to this general class of problems, the following scenerios were tested.The calculations were for fixed product-variable sequence and fixed sequence-variable product cases. The execution times for these cases are shown in the Tables 2 and 3, and refer to an IBM 3032 processor.

TABLE 2

Fixed Product, Variable Sequence

Products = 3

Number of Sequences	Seq. Combinations	Time
1	1	0.10
2	8	0.30
3	27	0.80
4	64	1.79
5	125	3.26

TABLE 3

Fixed Sequence, Variable Product

Sequence = 2

Number of Products	Seq. Combinations	Time
1	2	0.08
2	4	0.12
3	8	0.31
4	16	0.61
5	32	1.32
6	64	2.93

One can see that the execution time increases almost as fast as the sequence combinations,but are most sensitive to the number of products involved. The time required to perform these calculations are also relatively small. This means that enumeration of all feasible sequences is the only cumbersome problem in the solution of the sequence selection problem. This study assumed that all sequences have been enumerated in advance as part of a manufacturer's process planning activity.

Chapter VI

SUMMARY AND CONCLUSIONS

6.1 SUMMARY

In this thesis, the static sequence selection problem has been described and modelled. Analysis were done on four cases of the sequence selection problem:

(a) Single Product - Fixed Sequence

(b) Multiple Product - Fixed Sequence

(c) Single Product - Variable Sequence

(d) Multiple Product - Variable Sequence

Homogenous machine stations (flowlines) were compared to product - line systems in their effect on optimality and feasibility. The influence of machine requirements on the total cost of production was also discussed. Solution procedures for all of these four cases above were presented.

6.2 CONCLUSION

Results obained from this study indicate that in order to establish a system of the kind described here, it will be more efficient and less costly to have homogenous machine stations. It will then be possible for all of the products to share the work stations. Scheduling conflicts may arise, but since the object of this study was to find the optimal cost of production and machine requirements, the scheduling aspect of the problem is not considered. A product line will typically require more space, more machines and more labour.

One of the significant results of this study is the analysis of the fixed sequence case of the sequence selection problem. When the sequence is fixed, the combinatorial problem typical of sequencing problems dissappears. Consequently, the problem becomes very easy to solve by transformations and inspection. It is this result that led to the consideration of varying sequence cases where combinatorial problems exist. The actual calculations in the varying sequence case involve the same process, but, due to the combinatorial nature of the problem, it becomes less tractable to handle as the sequence combinations increase. Hence, the varying sequence problem is a case of many fixed sequences.

The model here had assumed that products can have larger unit loads at the beginning of process in order to meet the requirement at the last process. However,it may be possible to have equal unit loads between work stations and this can be achieved by replenishing the in-process products between work stations. This means that the defectives from a given station must be replaced before processing continues. The computation time required for defining a system configuration may not increase significantly.

Furthermore, results indicate that the solution times go up almost as fast as the sequence combinations. It then means that the time of search for optimality does not increase exponentially as the sequence combinations increase. Rather they increase more linearly.

6.3 RECOMMENDATIONS

This research has assumed that all sequences are already enumerated. But there is need to use available information, such as precedence restrictions, to obtain feasible sequences. An efficient enumeration technique will remove the difficulties associated with this assumption, and yet may not significantly increase the execution time of the program designed for the system's design calculations. It should be stated that enumeration can be cumbersome at this time without the use of some efficient technique.

Future research opportunity exists, therefore, in the development of an efficient enumeration algorithm which will reduce the amount of work the user of this study has to do.

Another important area for research is the scheduling conflict existing when the system is treated as homogenous machine stations. It has been established that the system can either be treated as a product-line or as a flowline system. However,it was also established that it is much more efficient and less costly to implement a general flowline system. In light of this, it becomes necessary to co-odinate the scheduling aspect of the system with the sequencing aspect.

The effect of in-process inventory can also be investigated. In this research, the assumption is that one can have larger unit loads at the beginning of a process in order to end up with the product requirements at the last station. However,in-process inventory can be allowed between processes, and its effect on the overall cost of production should be investigated.

It has already been mentioned, that the enumeration of sequences constitute a major hinderance for the planner who must process many products with variable sequences of operations. However, if an appropriate search technique is devised and imbedded in the program developed in this thesis, then the amount of computational effort required of the program developed in this study should be considerably reduced. It must be mentioned, however, that this task will not be a trivial one. The interdependence of many components of cost in the system reduce the possibility of using certain types of system control characteristics as variables for optimization. For example, the single product, two sequence problem solved in Chapter V, where the number of machines obtained in the optima sequence was more than that of the non-optimal sequence, indicates that not only specific ordering but the resultant, collective effect on variable and fixed processing cost must be considered(i.e., not a straightforward relationship to differences in sequences exists).

BIBLIOGRAPHY

1. Ashour, S., Sequencing Theory, Springer-Verlag, New York, New York, 1972.

2. Baker, K. R., Introduction to Sequencing and Scheduling, John Wiley and Sons Inc., New York, 1974.

3. Bazaraa, M. S., Jarvis, J.J., Linear Programming and Network Flows, John Wiley and Sons (1977).

4. Beightler, C. S., Phillips, D. T., Wilde, D.J., Foundations of Optimization, Prentice-Hall Inc., Englewood Clifts, N.J. (1979).

5. Burkard, R. E., Stratmann, K. H., "Numerical Inverstigations on Quadratic Assignment Problems", Naval Logistics Research Quarterly, Vol. 25, No. 1, March 1978, pp. 129-148.

6. Cantellow, D. G., Pitt, R. V., Saw, R. J., Hough, J., "Machine-Shop Problems: An Operational Research Approach", Opl R., Quarterly, Vol. 24, No. 4.

7. Chandra, R., "On n/1/F Dynamic Deterministic Problems", Naval Research Logistics Quarterly, Vol. 26, No. 3, September 1979.

8. Charlton, J. M., Death, C. C., "A Method of Solution for General Machine Scheduling Problems", Ops Res., Vol. 18, No 4, pp. 689-707,
(July - August 1970).

9. Conway, R. L., Maxwell, W. L., "Network Dispatching by the Shortest- Operation Discipline", Operations Research, Vol. 10, No. 1, (Feb. 1962).

10. Conway, R. W., Maxwell, W. L., Miller, L. W., Theory of Scheduling, Addison-Welsey Publishing Company, 1967.

11. Day, J.E., Hottenstein, M. P., "Review of Sequencing Research", Naval Research Logistics Quarterly, March 1970, Vol. 17, No. 1, pp. 11-39.

12. Delporte, C. M., Thomas, L. J., "Lotsizing and Sequencing for N Products on One Facility", Management Science, Vol. 23, No. 10, June, 1977.

13. Derman, C., Lieberman, G. J., Ross, S. M., "A Sequential Stochastic Assignment Porblem", Management Science, Vol. 18, No. 7, March, 1972.

14. Dudek, R. A., Ghare, P. M., "Makespan Sequencing on M-Machine", Journal of Industrial Engineering, Vol. 18, 1967, pp. 131-134.

15. Dudek, R. A., Smith, M. L., "Panwalker, S. S., Use of a Case Study in Sequencing/Scheduling Research", Omega, The Int. Jl of Mgmt Sci., Vol. 2, No. 2, 1974.

16. Dutta, S. K., Cunningham, A. A., " Sequencing Two-Machine Flowshops with Finite Intermediate Storage", Management Science, Vol. 21, No. 9, May 1975. Extensions, Naval Logistics Quarterly, Vol. 15, No. 2, (1968).

17. Elmaghraby, S.E.,"he Machine Sequencing Problem_Review and Extenions", Naval Research Logistics Quarterly,Vol. 15, No. 2, (1968).

18. Elmaghraby, S. E., "The Sequencing of Related Jobs", Naval Research Logistics Quarterly, Vol. 15, No. 1, (1968).

19. Emmons, H., "One Machine Sequencing to Minimize Certain Functions of Job Tardiness, Opns Res., Vol. 17, pp. 701-715, (1969).

20. Erlenkotter, D., Rogers,J. S., "Sequencing Competitive Expansion Projects"
Operations Research, Vol. 25, No. 6, November-December 1977.

21. Ferreira, A. C., Hodgson, T. J., "An N-Porduct, Multi-Machine, Lotsize Scheduling Model", AIIE Transactions, September, 1973.

22. Goyal, S. K., "Job-Shop Sequencing Problem with No-Wait in Process", J. Prod. Res., 1975, Vol. 13, No. 2, pp. 197-206.

23. Goyal, S. K., "Scheduling a Single Machine System: A Multi-Product Multi-Item Case", Opl Res. Q., Pergamon Press 1975, Vol. 26, 3, ii, pp. 619-627.

24. Graham, R. L., Lawler, E. L., Lenstra, J. K., Rinnooykan, A. H. G., "Optimization and Approximation in Deterministic Sequencing and Scheduling: A Survey", Annals of Discrete Mathematics 5 (1979), pp. 287-328.

25. Greene, T. J., Sadowski, R. P., "Loading the Cellularly Divided Group Technology Manufacturing System", 1980, Fall I E Conference Proceedings.

26. Gupta, J. N. D., "A Functional Heuristic Algorithm for the Flowshop Scheduling Problem", Opl Res. Quarterly, Vol. 22, No. 1.

27. Gupta, J. N. D., "M-stage Scheduling Problem - A Critical Appraisal", Int. Journ. of Pord. Res., 1971, Vol. 9, No. 2, pp. 267-281.

28. Hershauer, J.C., Ebert, R. J., "Search and Simulation Selection of a Job-Shop Sequencing Rule", Management Science, Vol. 21, No. 7, March, 1975.

29. Hitomi, K., Manufacturing Systems Engineering, Taylor and Francis Ltd., London, 1979.

30. Hodgson, T. J., "A Note on Single Machine Sequencing with Random Processing Times" Management Science, Vol. 23, No.10, June, 1977.

31. Inoe, M. S., "Optimal Sequencing and Routing Methods for Single-Stage Processing Systems", AIIE Technical Papers, Conference and Convention, 1970.

32. Jackson, J. R., "Network of Waiting Lines", Operations Res., Vol. 5, (Aug. 1967).

33. Jensen, P. A., Khan, H. A., "Scheduling in a Multistage Production System with Set-up and Inventory Costs", AIIE Transactions, Vol. 4, No. 1, (June, 1972).

34. Johnson, L. A., "Multi Stage Economic Lot size Problems with Static, Deterministic Demand", AIIE Technical Papers, Conference and Convention, 1972, pp. 387-392.

35. Kao, E. P. C., "A Multi-Product Dynamic Lot-size Model with Individual an Joint Set-up Costs", Operations Research, Vol. 27, No. 2, March-April, 1979.

36. Kornbluth, J. S. H., "A Sequencing Problem in a Batch/Flow Environment: An alternative Approach", Int. J. Prod. Res., 1972, Vol. 10, No. 3, pp. 291-293.

37. Krone, M. J., Steiglitz, K., "Heuristic-Programming Solution of a Flowshop-Scheduling Problem", Bell Telephone,

Whippany, New Jersey and Princeton University, Princeton, New Jersey, (1971).

38. Lambrecht, M. R., Vanderveken, H., "Heuristics Procedures for the Single Operation, Multi-Item Loading Problem", AIIE Transactions, Vol. 11, No. 4 (1979).

39. Lanzenauer, C. H. V., "A Production Scheduling Model by Bivalent Linear Programming", Management Science, Vol. 17, No. 1, September, 1970.

40. Lawler, C. L., More, J. M., "A Functional Equation and its Application to Resource Allocation and Sequencing Problems", Management Science, Vol. 16, No. 1, September, 1969.

41. Lenshra, J. K., Sequencing by Enumerative Methods, Mathematical Centre Tracts 69 (1977).

42. Little, J. D. C., et al., "An algorithm for the Travelling Salesman Problem", Opns Res. JVol. 11, pp. 972-989, (1963).

43. Mitten, L. G., "A Scheduling Problem", J. Industrial Engng, 10, No. 2, (March-April 1959).

44. Mitten, L. G., (1959a), "Sequencing n Jobs on Two Machines with Arbitrary Time Lags", Mgmt Sci., Vol. 5, No. 3.

45. Moore, J. M., An in Job,"One Machine Sequencing Algorithm for Minimizing the Number of Late Jobs", Management Science, Vol. 15, No. 1, September, 1968.

46. Muhlemann, A. P., Lockett, kA. G.," A Scheduling Problem with Machine Flexiblility", Int. J. Prod. Res., 1975, Vol. 13, No. 1, pp. 57-73.

47. Murphy, F. H., Soyster, A. L.,"Multiproduct Lotsize Scheduling with Proportionality Product Demands", Naval Research Logistics Quarterly, March, 1979, Vol. 26, No. 1.

48. Panwalder, S. S., Woollm, C. R., "Flowshop Scheduling Problems with No In-Process Waiting: A Special Case", J. Opl Res. Soc., Vol. 30, 7, pp. 661-664.

49. Peterson, P. C., "Solving Sequencing Problems Through Reordering Operations", AIIE Transactions, Vol. 5, No. 1, (1973).

50. Pinedo, M. Weiss, G., "Scheduling Stochastic Tasks on Two Parallel Processors" Naval Research Logistics Quarterly, Vol. 26, No. 3, September, 1979.

51. Rathliff, H. D., "Network Models for Production Scheduling Problems with Convex Cost and Batch Processing" A.I.I.E. Transactions, March, 1978.

52. Reddi, S. S., Ramamoorthy, C. V., "On the Flow-Shop Sequencing Problem with No Wait in Process", Operational Research Quarterly, Vol. 23, No. 3, pp. 323-331.

53. Resh, M., Friedman, M., Barbosa, L. C., "On a General Solution of the Deterministic Lot-Size Problem with Time -Porportional Demand", Operations Research, Vol. 24, No. 4, July-August, 1976.

54. Salkin, H. M., Integer Programming, Addison-Wesley Publishing Company, (1974).

55. Schrage, L., Baker, K. R., "Dynamic Programming Solution of Sequencing Problems with Precedence Constraints", Operations Research, Vol. 26, No. 3, May-June, 1978.

56. Sidney, J. B., "Decomposition Algorithms for a Single-Machine Sequencing with Precedence Relations and Deferral Costs", Operations Research, Vol. 23, No. 2, March-April, 1975.

57. Smith, M. L., Panwalker, S. S., Dudek, R. A., "Flowshop Sequencing Problem with Ordered Processing Time Matrices", Management Science, Vol. 21, No. 5, January, 1971.

58. Solem, Olav, "Contributions to the Solution of Sequencing Problems in Process Industry", Int. Journ. of Prod. Res., 1974, Vol. 12, No, 1, pp. 55-75.

59. Spencer, D. L., Dudek, R. A., "The Dependent Shop Sequencing Algorithm. A General Algorithm for Solving Flowshop, Jobshop, and Dependent Shop Sequencing Problems", Int. J. Prod. Res., 1973, Vol. 11, No. 4, pp. 325-339.

60. Taha, H. A., Skeith, R. W.,"The Economic Lotsize in Multi-stage Productio Systems", AIIE Transactions, Vol. 2, No. 2, pp. 157-162, June, 1970.

61. Tamir, A., "Scheduling Jobs to Two Machines Subject to Batch Arrival Ordering", Naval Research Logistics Quarterly, Vol 26, No. 3, Sep. 1979.

62. Tudehope, B. J. S., "The Transition-Point Method to Meet Due-Dates and Minimize the Makespan in the Job-Shop Sequencing Problem", Int. J. Prod. Res., 1972, Vol. 10, No. 1, pp 3-21.

63. Wismer, D. A., "Solution of the Flow-shop Scheduling Problem with no Intermediate Queues", Operations Res. Vol. 20, pp. 689-697.

64. Wysk,R.A.,Barash,M.M.,Moodie,C.L.,"An Analysis of Some Production Planning Practices",ASME,78-wa/Prod-13,December, 1978,

65. Young, H. H., "Optimization Models for Production Lines", Journal of Industrial Engineering, January, 1967, Vol. 18, pp. 70-78.

Appendix A

EXAMPLE PROBLEM

EXAMPLE PROBLEM

The following illustrative problem will be used for computational experience in this thesis, and it represents the framework of the type of problem which is treated in this study.

A company wants to manufacture 3 sub-products. There is a total of 5 work stations at the company's facility. All the machines at each work station can perform only one type of operation. None of these sub-products needs all five processes. Setup time at each work station depends on the preceeding operation for a given product. Process time also depends on preceeding operations for a given product. The management wants to implement this new system but has $3000.00 in it's budget for the purchase of the necessary machines. The company only wants four basic types of machines for this sub-system as shown in Table A3. Tables A1-A7 contain the operational restrictions and requirements and production cost components. Two types of production layout have been proposed:

(a) Product layout, and

(b) Process layout.

These two layouts are shown in Figure 6. The sub-products could be carried in batches between work stations. The following tables show the characteristics of the system:

<u>Comment on the Use of TABLE A3</u>

Since setup time is dependent on process and product predecessor, and if operation B is prededed by say operation A, then for product 2, the setup time will be 8 min. This is the middle value for setup column, B row. For precedence by A and product 3, the value of 6 min is used and this corresponds to the third value of setup column, B row.

Material Handling Cost

Cost of material handling ($T_{i,j}$) for the products are:

Product 1 - $0.05/product/process

Product 2 - $0.06/product/process

Product 3 - $0.04/product/process

TABLE A1

Operational Restrictions

PRECEDENCE RESTRICTIONS ON OPERATIONS

Operation	Predecessor
A	NONE
B	A
C	A
D	A
E	B,C,D

TABLE A2

Daily Demand per Product

Daily demand per production period for products are as follows:

Product	Daily demand(items per prod. period)
1	1000
2	2000
3	1500

TABLE A3

Percentage of Defectives, Setup & Operation Times

Oper.	If Preceded by Sequence Oper., Prod.,1,2,3		% Def.	Setup Time(Min) Prods1,2,3	Oper. Time (Min)
A	-	1,2,3	0.4	8,6,7	1.5
B	A	1,2,3	1.0	10,8,6	2.5
	C	1,2,3	0.8	12,8,9	1.6
	D	1,2,3	0.7	9,10,6	1.0
C	A	1,2,3	0.5	5,12,4	2.0
	B	1,2,3	0.3	4,5,10	2.3
	D	1,2,3	0.6	5,8,9	2.6
D	A	1,2,3	0.2	7,12,4	3.0
	B	1,2,3	1.0	10,6,8	2.5
	C	1,2,3	0.5	12,10,4	3.1
E	B	1,2,3	0.6	10,7,5	2.5
	C	1,2,3	0.7	8,6,4	2.0
	D	1,2,3	0.5	11,9,6	1.5

Fixed Cost

A certain cost($F_{i,j}$) is incurred by using $n_{i,j}$ machines (where $n_{i,j}$ is the number of machines performing operation j for product i). This cost is $200/day for product 1, $220/day for product 2 and $150/day for product 3.

The weight of each of the products:

w(1) = 2kg/product

w(2) = 3kg/product

w(3) = 2.5kg/product

The maximum weight of each batch size:

Wmax(1) = 100kg

Wmax(2) = 150kg

Wmax(3) = 125kg

Typical Design Questions are:

1. If there are 18 production hours available each day, find the optimum (minimum production time and minimum cost of production sequence of operations).

TABLE A4

Setup & Operation Costs

Operation	Oper.Cost($/min)	SetupCost($/min)		
		Product 1	2	3
A	0.25	0.05	0.04	0.06
B	0.50	0.10	0.08	0.09
C	0.40	0.07	0.07	0.08
D	0.10	0.08	0.06	0.05
E	0.20	0.06	0.08	0.07

TABLE A5

Product/Process Table

Product	Processes
1	A B D E
2	A B C E
3	A C D E

TABLE A6

Machine Types

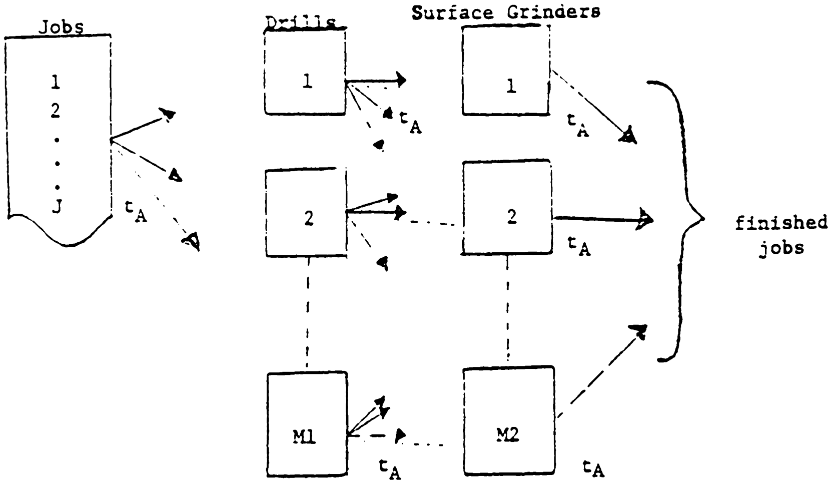
Operation	Machine
A	Lathe (for turning)
B	Mill
C	Drill
D	Drill
E	Surface Grinder

TABLE A7

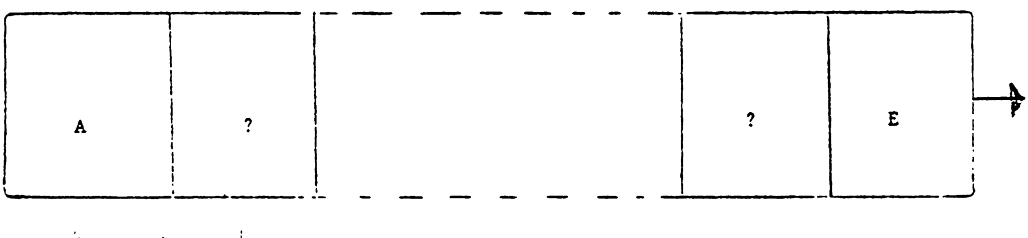
Transportation Cost(T_{ij}) Table($)

PRODUCT(i)	OPERATION A	B	C	D	E
1	.50	.60	.70	.40	.60
2	.60	.50	.40	.70	.60
3	.70	.60	.30	.20	.50

2. What are the minimum number of machines required at each work station for the optimum layout? Assume that all machines are homogenous within type.



Process layout
t_A = 3 mins.



Product Layout (Flowline)
t_B = 2.00 min

Figure 6: Product & Process Layouts

Appendix B

SAMPLE RESULTS

```
*******************************************************
                  SAMPLE RESULTS
*******************************************************
 NO.OF PROD.  OP.TIME  INVST COST  NO.SEQ  MAX.NO.OP.

       3       1080      3000         8         4

++++++++++++++++++++++++++++++++++++++++++++++++++++++

    3 PROD.  8 POSSIBLE SEQS  &  4 OPER./PROD.

++++++++++++++++++++++++++++++++++++++++++++++++++++++
 NUMBER OF OPERATIONS FOR EACH PROD.

       4         4         4

 NUMBER OF SEQUENCES FOR EACH PRODUCT

       2         2         2

 OPERATION NUMBERS FOR EACH SEQUENCE

       1         2         4         5

       1         4         2         5

       1         2         3         5

       1         3         2         5

       1         3         4         5

       1         4         3         5

 DEMAND FOR EACH PRODUCT

     1000      2000      1500

 COST FOR EACH MACHINE

    50.0000   50.0000   50.0000   50.0000

    50.0000   50.0000   50.0000   50.0000

    50.0000   50.0000   50.0000   50.0000

 DEFECTIVE FOR EACH PROD & SEQUENCE
```

0.4000	1.0000	1.0000	0.5000
0.4000	0.2000	0.7000	0.6000
0.4000	1.0000	0.3000	0.7000
0.4000	0.5000	0.8000	0.6000
0.4000	0.5000	0.5000	0.5000
0.4000	0.2000	0.6000	0.3000

PROCESS TIMES FOR EACH PROCESS & SEQUENCE

1.5000	2.5000	2.5000	1.5000
1.5000	3.0000	1.0000	2.5000
1.5000	2.5000	2.3000	2.0000
1.5000	2.0000	1.6000	2.5000
1.5000	2.0000	3.1000	1.5000
1.5000	3.0000	2.6000	2.0000

SETUP TIME FOR EACH PRODUCT & SEQUENCE

8.0000	10.0000	10.0000	11.0000
8.0000	7.0000	9.0000	10.0000
6.0000	8.0000	5.0000	6.0000
6.0000	12.0000	8.0000	7.0000
7.0000	4.0000	4.0000	6.0000
7.0000	4.0000	9.0000	4.0000

TRANSPORTATION COST FOR EACH PROD. & SEQ.

0.5000	0.6000	0.4000	0.6000
0.5000	0.4000	0.6000	0.6000
0.6000	0.5000	0.4000	0.6000
0.6000	0.4000	0.5000	0.6000

0.7000	0.3000	0.2000	0.5000
0.7000	0.2000	0.3000	0.5000

FIXED COST HAVING MACHINE

200.0000	200.0000	200.0000	200.0000
200.0000	200.0000	200.0000	200.0000
200.0000	200.0000	200.0000	200.0000

COST OF SETTING UP FOR EACH BATCH

0.0500	0.1000	0.0800	0.0600
0.0500	0.0800	0.1000	0.0600
0.0400	0.0800	0.0700	0.0800
0.0400	0.0700	0.0800	0.0800
0.0600	0.0800	0.0500	0.0700
0.0600	0.0500	0.0800	0.0700

PROCESSING COST

0.2500	0.5000	0.1000	0.2000
0.2500	0.1000	0.5000	0.2000
0.2500	0.5000	0.4000	0.2000
0.2500	0.4000	0.5000	0.2000
0.2500	0.4000	0.1000	0.2000
0.2500	0.1000	0.4000	0.2000

WEIGHT FOR EACH PRODUCT

2.0000	3.0000	2.5000

MAXIMUM WEIGHT FOR EACH PRODUCT TYPE

100.0000	150.0000	125.0000

SEQUENCE COMBINATION SETS FOR EACH PRODUCT

1	1	1
2	1	1
1	2	1
2	2	1
1	1	2
2	1	2
1	2	2
2	2	2

DEFECTIVE OF PROD. 1 AT OPERATION 4= 0.9950

DEFECTIVE OF PROD. 1 AT OPERATION 3= 0.9850

DEFECTIVE OF PROD. 1 AT OPERATION 2= 0.9752

DEFECTIVE OF PROD. 1 AT OPERATION 1= 0.9713

FREQUENCY OF PROD. 1= 21

DEFECTIVE OF PROD. 1 AT OPERATION 4= 0.9940

DEFECTIVE OF PROD. 1 AT OPERATION 3= 0.9870

DEFECTIVE OF PROD. 1 AT OPERATION 2= 0.9851

DEFECTIVE OF PROD. 1 AT OPERATION 1= 0.9811

FREQUENCY OF PROD. 1= 21

DEFECTIVE OF PROD. 2 AT OPERATION 4= 0.9930

DEFECTIVE OF PROD. 2 AT OPERATION 3= 0.9900

DEFECTIVE OF PROD. 2 AT OPERATION 2= 0.9801

DEFECTIVE OF PROD. 2 AT OPERATION 1= 0.9762

FREQUENCY OF PROD. 2= 41

DEFECTIVE OF PROD. 2 AT OPERATION 4= 0.9940

DEFECTIVE OF PROD. 2 AT OPERATION 3= 0.9860

```
DEFECTIVE OF PROD.  2 AT OPERATION  2=  0.9811
DEFECTIVE OF PROD.  2 AT OPERATION  1=  0.9772
FREQUENCY OF PROD.   2=  41
DEFECTIVE OF PROD.  3 AT OPERATION  4=  0.9950
DEFECTIVE OF PROD.  3 AT OPERATION  3=  0.9900
DEFECTIVE OF PROD.  3 AT OPERATION  2=  0.9851
DEFECTIVE OF PROD.  3 AT OPERATION  1=  0.9811
FREQUENCY OF PROD.   3=  31
DEFECTIVE OF PROD.  3 AT OPERATION  4=  0.9970
DEFECTIVE OF PROD.  3 AT OPERATION  3=  0.9910
DEFECTIVE OF PROD.  3 AT OPERATION  2=  0.9890
DEFECTIVE OF PROD.  3 AT OPERATION  1=  0.9851
FREQUENCY OF PROD.   3=  31
SEQUENCE  COMBINATION  1 RESULTS
PROC.COST FOR PROD. 1 OPER. 1=   375.000000
SETUP COST FOR PROD. 1 OPER. 1=    1.717889
TRANS.COST FOR PROD. 1 OPER. 1=   10.198630
FIXED COST FOR PROD. 1 OPER. 1=    336.886200
PROC.COST FOR PROD. 2 OPER. 1=   750.000000
SETUP COST FOR PROD. 2 OPER. 1=    5.858154
TRANS.COST FOR PROD. 2 OPER. 1=   24.014520
FIXED COST FOR PROD. 2 OPER. 1=    731.826400
PROC.COST FOR PROD. 3 OPER. 1=   562.500000
SETUP COST FOR PROD. 3 OPER. 1=    4.826466
TRANS.COST FOR PROD. 3 OPER. 1=   21.290610
```

```
FIXED COST FOR PROD. 3 OPER. 1=    528.953800
 MACHINES REQUIRED AT STATION 1=      8.000000
   COST AT STATION  1 =          3353.072000
PROC.COST FOR PROD. 1 OPER. 2=  1250.000000
SETUP COST FOR PROD. 1 OPER. 2=    5.998573
TRANS.COST FOR PROD. 1 OPER. 2=   12.287510
FIXED COST FOR PROD. 1 OPER. 2=    585.821200
PROC.COST FOR PROD. 2 OPER. 2=  2500.000000
SETUP COST FOR PROD. 2 OPER. 2=   21.621000
TRANS.COST FOR PROD. 2 OPER. 2=   20.092460
FIXED COST FOR PROD. 2 OPER. 2=   1345.094000
PROC.COST FOR PROD. 3 OPER. 2=  1199.999000
SETUP COST FOR PROD. 3 OPER. 2=    7.767385
TRANS.COST FOR PROD. 3 OPER. 2=    9.161196
FIXED COST FOR PROD. 3 OPER. 2=    635.893000
 MACHINES REQUIRED AT STATION 2=     13.000000
   COST AT STATION  2 =          7593.734000
PROC.COST FOR PROD. 1 OPER. 3=   250.000000
SETUP COST FOR PROD. 1 OPER. 3=    4.810224
TRANS.COST FOR PROD. 1 OPER. 3=    8.274419
FIXED COST FOR PROD. 1 OPER. 3=    581.336900
PROC.COST FOR PROD. 2 OPER. 3=  1839.999000
SETUP COST FOR PROD. 2 OPER. 3=   15.052790
TRANS.COST FOR PROD. 2 OPER. 3=   16.236320
FIXED COST FOR PROD. 2 OPER. 3=   1059.548000
```

```
PROC.COST FOR PROD. 3 OPER. 3=   465.000000
SETUP COST FOR PROD. 3 OPER. 3=    7.529474
TRANS.COST FOR PROD. 3 OPER. 3=    6.138153
FIXED COST FOR PROD. 3 OPER. 3=    981.334400
 MACHINES REQUIRED AT STATION 3=     14.000000
   COST AT STATION  3 =          5235.257000
PROC.COST FOR PROD. 1 OPER. 4=   299.999700
SETUP COST FOR PROD. 1 OPER. 4=    2.223121
TRANS.COST FOR PROD. 1 OPER. 4=   12.536990
FIXED COST FOR PROD. 1 OPER. 4=    354.649900
PROC.COST FOR PROD. 2 OPER. 4=   799.999700
SETUP COST FOR PROD. 2 OPER. 4=   15.698990
TRANS.COST FOR PROD. 2 OPER. 4=   24.427790
FIXED COST FOR PROD. 2 OPER. 4=    964.003900
PROC.COST FOR PROD. 3 OPER. 4=   449.999700
SETUP COST FOR PROD. 3 OPER. 4=    5.455729
TRANS.COST FOR PROD. 3 OPER. 4=   15.422490
FIXED COST FOR PROD. 3 OPER. 4=    505.359100
 MACHINES REQUIRED AT STATION 4=     10.000000
   COST AT STATION  4 =          3449.776000
 TOTAL COST FOR SEQUENCE   1=        19631.830000
 TOTAL     INVESTMENT     COST     FOR     SEQUENCE       1=
2250.000000
********************************************************
 SEQUENCE  COMBINATION  2 RESULTS
```

```
PROC.COST FOR PROD. 1 OPER. 1=   375.000000
SETUP COST FOR PROD. 1 OPER. 1=    1.720988
TRANS.COST FOR PROD. 1 OPER. 1=   10.301830
FIXED COST FOR PROD. 1 OPER. 1=    334.113000
PROC.COST FOR PROD. 2 OPER. 1=   750.000000
SETUP COST FOR PROD. 2 OPER. 1=    5.858154
TRANS.COST FOR PROD. 2 OPER. 1=   24.014520
FIXED COST FOR PROD. 2 OPER. 1=    731.826400
PROC.COST FOR PROD. 3 OPER. 1=   562.500000
SETUP COST FOR PROD. 3 OPER. 1=    4.826466
TRANS.COST FOR PROD. 3 OPER. 1=   21.290610
FIXED COST FOR PROD. 3 OPER. 1=    528.953800
 MACHINES REQUIRED AT STATION 1=       8.000000
   COST AT STATION  1 =           3350.405000
PROC.COST FOR PROD. 1 OPER. 2=   300.000000
SETUP COST FOR PROD. 1 OPER. 2=    5.389246
TRANS.COST FOR PROD. 1 OPER. 2=    8.274569
FIXED COST FOR PROD. 1 OPER. 2=    651.302400
PROC.COST FOR PROD. 2 OPER. 2=  2500.000000
SETUP COST FOR PROD. 2 OPER. 2=   21.621000
TRANS.COST FOR PROD. 2 OPER. 2=   20.092460
FIXED COST FOR PROD. 2 OPER. 2=   1345.094000
PROC.COST FOR PROD. 3 OPER. 2=  1199.999000
SETUP COST FOR PROD. 3 OPER. 2=    7.767385
TRANS.COST FOR PROD. 3 OPER. 2=    9.161196
```

```
FIXED COST FOR PROD. 3 OPER. 2=    635.893000
 MACHINES REQUIRED AT STATION 2=     14.000000
   COST AT STATION  2 =          6704.593000
PROC.COST FOR PROD. 1 OPER. 3=   500.000000
SETUP COST FOR PROD. 1 OPER. 3=    2.350440
TRANS.COST FOR PROD. 1 OPER. 3=   12.436720
FIXED COST FOR PROD. 1 OPER. 3=    226.790200
PROC.COST FOR PROD. 2 OPER. 3=  1839.999000
SETUP COST FOR PROD. 2 OPER. 3=   15.052790
TRANS.COST FOR PROD. 2 OPER. 3=   16.236320
FIXED COST FOR PROD. 2 OPER. 3=   1059.548000
PROC.COST FOR PROD. 3 OPER. 3=   465.000000
SETUP COST FOR PROD. 3 OPER. 3=    7.529474
TRANS.COST FOR PROD. 3 OPER. 3=    6.138153
FIXED COST FOR PROD. 3 OPER. 3=    981.334400
 MACHINES REQUIRED AT STATION 3=     12.000000
   COST AT STATION  3 =          5132.414000
PROC.COST FOR PROD. 1 OPER. 4=   499.999700
SETUP COST FOR PROD. 1 OPER. 4=    3.615450
TRANS.COST FOR PROD. 1 OPER. 4=   12.524390
FIXED COST FOR PROD. 1 OPER. 4=    577.345400
PROC.COST FOR PROD. 2 OPER. 4=   799.999700
SETUP COST FOR PROD. 2 OPER. 4=   15.698990
TRANS.COST FOR PROD. 2 OPER. 4=   24.427790
FIXED COST FOR PROD. 2 OPER. 4=    964.003900
```

PROC.COST FOR PROD. 3 OPER. 4= 449.999700

SETUP COST FOR PROD. 3 OPER. 4= 5.455729

TRANS.COST FOR PROD. 3 OPER. 4= 15.422490

FIXED COST FOR PROD. 3 OPER. 4= 505.359100

MACHINES REQUIRED AT STATION 4= 11.000000

COST AT STATION 4 = 3873.851000

TOTAL COST FOR SEQUENCE 2= 19061.260000

TOTAL INVESTMENT COST FOR SEQUENCE 2= 2250.000000

****

SEQUENCE COMBINATION 3 RESULTS

PROC.COST FOR PROD. 1 OPER. 1= 375.000000

SETUP COST FOR PROD. 1 OPER. 1= 1.717889

TRANS.COST FOR PROD. 1 OPER. 1= 10.198630

FIXED COST FOR PROD. 1 OPER. 1= 336.886200

PROC.COST FOR PROD. 2 OPER. 1= 750.000000

SETUP COST FOR PROD. 2 OPER. 1= 5.859856

TRANS.COST FOR PROD. 2 OPER. 1= 24.038940

FIXED COST FOR PROD. 2 OPER. 1= 731.295100

PROC.COST FOR PROD. 3 OPER. 1= 562.500000

SETUP COST FOR PROD. 3 OPER. 1= 4.826466

TRANS.COST FOR PROD. 3 OPER. 1= 21.290610

FIXED COST FOR PROD. 3 OPER. 1= 528.953800

MACHINES REQUIRED AT STATION 1= 8.000000

COST AT STATION 1 = 3352.566000

```
PROC.COST FOR PROD. 1 OPER. 2=  1250.000000
SETUP COST FOR PROD. 1 OPER. 2=    5.998573
TRANS.COST FOR PROD. 1 OPER. 2=   12.287510
FIXED COST FOR PROD. 1 OPER. 2=    585.821200
PROC.COST FOR PROD. 2 OPER. 2=  1599.999000
SETUP COST FOR PROD. 2 OPER. 2=   19.220120
TRANS.COST FOR PROD. 2 OPER. 2=   16.090310
FIXED COST FOR PROD. 2 OPER. 2=   1365.159000
PROC.COST FOR PROD. 3 OPER. 2=  1199.999000
SETUP COST FOR PROD. 3 OPER. 2=    7.767385
TRANS.COST FOR PROD. 3 OPER. 2=    9.161196
FIXED COST FOR PROD. 3 OPER. 2=    635.893000
 MACHINES REQUIRED AT STATION 2=      13.000000
   COST AT STATION  2 =          6707.394000
PROC.COST FOR PROD. 1 OPER. 3=   250.000000
SETUP COST FOR PROD. 1 OPER. 3=    4.810224
TRANS.COST FOR PROD. 1 OPER. 3=    8.274419
FIXED COST FOR PROD. 1 OPER. 3=    581.336900
PROC.COST FOR PROD. 2 OPER. 3=  1600.000000
SETUP COST FOR PROD. 2 OPER. 3=   13.873010
TRANS.COST FOR PROD. 2 OPER. 3=   20.213970
FIXED COST FOR PROD. 2 OPER. 3=    857.885000
PROC.COST FOR PROD. 3 OPER. 3=   465.000000
SETUP COST FOR PROD. 3 OPER. 3=    7.529474
TRANS.COST FOR PROD. 3 OPER. 3=    6.138153
```

FIXED COST FOR PROD. 3 OPER. 3= 981.334400

MACHINES REQUIRED AT STATION 3= 13.000000

COST AT STATION 3 = 4796.394000

PROC.COST FOR PROD. 1 OPER. 4= 299.999700

SETUP COST FOR PROD. 1 OPER. 4= 2.223121

TRANS.COST FOR PROD. 1 OPER. 4= 12.536990

FIXED COST FOR PROD. 1 OPER. 4= 354.649900

PROC.COST FOR PROD. 2 OPER. 4= 999.999700

SETUP COST FOR PROD. 2 OPER. 4= 20.636100

TRANS.COST FOR PROD. 2 OPER. 4= 24.452390

FIXED COST FOR PROD. 2 OPER. 4= 1265.895000

PROC.COST FOR PROD. 3 OPER. 4= 449.999700

SETUP COST FOR PROD. 3 OPER. 4= 5.455729

TRANS.COST FOR PROD. 3 OPER. 4= 15.422490

FIXED COST FOR PROD. 3 OPER. 4= 505.359100

MACHINES REQUIRED AT STATION 4= 11.000000

COST AT STATION 4 = 3956.630000

TOTAL COST FOR SEQUENCE 3= 18812.980000

TOTAL INVESTMENT COST FOR SEQUENCE 3= 2250.000000

****

SEQUENCE COMBINATION 4 RESULTS

PROC.COST FOR PROD. 1 OPER. 1= 375.000000

SETUP COST FOR PROD. 1 OPER. 1= 1.720988

TRANS.COST FOR PROD. 1 OPER. 1= 10.301830

```
FIXED COST FOR PROD. 1 OPER. 1=    334.113000
PROC.COST FOR PROD. 2 OPER. 1=   750.000000
SETUP COST FOR PROD. 2 OPER. 1=    5.859856
TRANS.COST FOR PROD. 2 OPER. 1=   24.038940
FIXED COST FOR PROD. 2 OPER. 1=    731.295100
PROC.COST FOR PROD. 3 OPER. 1=   562.500000
SETUP COST FOR PROD. 3 OPER. 1=    4.826466
TRANS.COST FOR PROD. 3 OPER. 1=   21.290610
FIXED COST FOR PROD. 3 OPER. 1=    528.953800
 MACHINES REQUIRED AT STATION 1=       8.000000
   COST AT STATION  1 =           3349.900000
PROC.COST FOR PROD. 1 OPER. 2=   300.000000
SETUP COST FOR PROD. 1 OPER. 2=    5.389246
TRANS.COST FOR PROD. 1 OPER. 2=    8.274569
FIXED COST FOR PROD. 1 OPER. 2=    651.302400
PROC.COST FOR PROD. 2 OPER. 2=  1599.999000
SETUP COST FOR PROD. 2 OPER. 2=   19.220120
TRANS.COST FOR PROD. 2 OPER. 2=   16.090310
FIXED COST FOR PROD. 2 OPER. 2=   1365.159000
PROC.COST FOR PROD. 3 OPER. 2=  1199.999000
SETUP COST FOR PROD. 3 OPER. 2=    7.767385
TRANS.COST FOR PROD. 3 OPER. 2=    9.161196
FIXED COST FOR PROD. 3 OPER. 2=    635.893000
 MACHINES REQUIRED AT STATION 2=      14.000000
   COST AT STATION  2 =           5818.253000
```

PROC.COST FOR PROD. 1 OPER. 3= 500.000000

SETUP COST FOR PROD. 1 OPER. 3= 2.350440

TRANS.COST FOR PROD. 1 OPER. 3= 12.436720

FIXED COST FOR PROD. 1 OPER. 3= 226.790200

PROC.COST FOR PROD. 2 OPER. 3= 1600.000000

SETUP COST FOR PROD. 2 OPER. 3= 13.873010

TRANS.COST FOR PROD. 2 OPER. 3= 20.213970

FIXED COST FOR PROD. 2 OPER. 3= 857.885000

PROC.COST FOR PROD. 3 OPER. 3= 465.000000

SETUP COST FOR PROD. 3 OPER. 3= 7.529474

TRANS.COST FOR PROD. 3 OPER. 3= 6.138153

FIXED COST FOR PROD. 3 OPER. 3= 981.334400

MACHINES REQUIRED AT STATION 3= 11.000000

COST AT STATION 3 = 4693.546000

PROC.COST FOR PROD. 1 OPER. 4= 499.999700

SETUP COST FOR PROD. 1 OPER. 4= 3.615450

TRANS.COST FOR PROD. 1 OPER. 4= 12.524390

FIXED COST FOR PROD. 1 OPER. 4= 577.345400

PROC.COST FOR PROD. 2 OPER. 4= 999.999700

SETUP COST FOR PROD. 2 OPER. 4= 20.636100

TRANS.COST FOR PROD. 2 OPER. 4= 24.452390

FIXED COST FOR PROD. 2 OPER. 4= 1265.895000

PROC.COST FOR PROD. 3 OPER. 4= 449.999700

SETUP COST FOR PROD. 3 OPER. 4= 5.455729

TRANS.COST FOR PROD. 3 OPER. 4= 15.422490

FIXED COST FOR PROD. 3 OPER. 4= 505.359100

MACHINES REQUIRED AT STATION 4= 12.000000

COST AT STATION 4 = 4380.703000

TOTAL COST FOR SEQUENCE 4= 18242.400000

TOTAL INVESTMENT COST FOR SEQUENCE 4=
2250.000000

****

SEQUENCE COMBINATION 5 RESULTS

PROC.COST FOR PROD. 1 OPER. 1= 375.000000

SETUP COST FOR PROD. 1 OPER. 1= 1.717889

TRANS.COST FOR PROD. 1 OPER. 1= 10.198630

FIXED COST FOR PROD. 1 OPER. 1= 336.886200

PROC.COST FOR PROD. 2 OPER. 1= 750.000000

SETUP COST FOR PROD. 2 OPER. 1= 5.858154

TRANS.COST FOR PROD. 2 OPER. 1= 24.014520

FIXED COST FOR PROD. 2 OPER. 1= 731.826400

PROC.COST FOR PROD. 3 OPER. 1= 562.500000

SETUP COST FOR PROD. 3 OPER. 1= 4.831235

TRANS.COST FOR PROD. 3 OPER. 1= 21.376220

FIXED COST FOR PROD. 3 OPER. 1= 527.355900

MACHINES REQUIRED AT STATION 1= 8.000000

COST AT STATION 1 = 3351.564000

PROC.COST FOR PROD. 1 OPER. 2= 1250.000000

SETUP COST FOR PROD. 1 OPER. 2= 5.998573

TRANS.COST FOR PROD. 1 OPER. 2= 12.287510

```
FIXED COST FOR PROD. 1 OPER. 2=    585.821200
PROC.COST FOR PROD. 2 OPER. 2=  2500.000000
SETUP COST FOR PROD. 2 OPER. 2=   21.621000
TRANS.COST FOR PROD. 2 OPER. 2=   20.092460
FIXED COST FOR PROD. 2 OPER. 2=   1345.094000
PROC.COST FOR PROD. 3 OPER. 2=   450.000000
SETUP COST FOR PROD. 3 OPER. 2=    7.285656
TRANS.COST FOR PROD. 3 OPER. 2=    6.132021
FIXED COST FOR PROD. 3 OPER. 2=    950.506500
 MACHINES REQUIRED AT STATION 2=      15.000000
   COST AT STATION  2 =          7154.835000
PROC.COST FOR PROD. 1 OPER. 3=   250.000000
SETUP COST FOR PROD. 1 OPER. 3=    4.810224
TRANS.COST FOR PROD. 1 OPER. 3=    8.274419
FIXED COST FOR PROD. 1 OPER. 3=    581.336900
PROC.COST FOR PROD. 2 OPER. 3=  1839.999000
SETUP COST FOR PROD. 2 OPER. 3=   15.052790
TRANS.COST FOR PROD. 2 OPER. 3=   16.236320
FIXED COST FOR PROD. 2 OPER. 3=   1059.548000
PROC.COST FOR PROD. 3 OPER. 3=  1560.000000
SETUP COST FOR PROD. 3 OPER. 3=   12.037230
TRANS.COST FOR PROD. 3 OPER. 3=    9.216467
FIXED COST FOR PROD. 3 OPER. 3=    979.543700
 MACHINES REQUIRED AT STATION 3=      14.000000
   COST AT STATION  3 =          6336.054000
```

PROC.COST FOR PROD. 1 OPER. 4= 299.999700

SETUP COST FOR PROD. 1 OPER. 4= 2.223121

TRANS.COST FOR PROD. 1 OPER. 4= 12.536990

FIXED COST FOR PROD. 1 OPER. 4= 354.649900

PROC.COST FOR PROD. 2 OPER. 4= 799.999700

SETUP COST FOR PROD. 2 OPER. 4= 15.698990

TRANS.COST FOR PROD. 2 OPER. 4= 24.427790

FIXED COST FOR PROD. 2 OPER. 4= 964.003900

PROC.COST FOR PROD. 3 OPER. 4= 599.999700

SETUP COST FOR PROD. 3 OPER. 4= 6.806971

TRANS.COST FOR PROD. 3 OPER. 4= 15.453490

FIXED COST FOR PROD. 3 OPER. 4= 629.258500

MACHINES REQUIRED AT STATION 4= 10.000000

COST AT STATION 4 = 3725.058000

TOTAL COST FOR SEQUENCE 5= 20567.500000

TOTAL INVESTMENT COST FOR SEQUENCE 5= 2350.000000

****

SEQUENCE COMBINATION 6 RESULTS

PROC.COST FOR PROD. 1 OPER. 1= 375.000000

SETUP COST FOR PROD. 1 OPER. 1= 1.720988

TRANS.COST FOR PROD. 1 OPER. 1= 10.301830

FIXED COST FOR PROD. 1 OPER. 1= 334.113000

PROC.COST FOR PROD. 2 OPER. 1= 750.000000

SETUP COST FOR PROD. 2 OPER. 1= 5.858154

```
TRANS.COST FOR PROD. 2 OPER. 1=   24.014520
FIXED COST FOR PROD. 2 OPER. 1=    731.826400
PROC.COST FOR PROD. 3 OPER. 1=   562.500000
SETUP COST FOR PROD. 3 OPER. 1=    4.831235
TRANS.COST FOR PROD. 3 OPER. 1=   21.376220
FIXED COST FOR PROD. 3 OPER. 1=    527.355900
 MACHINES REQUIRED AT STATION 1=      8.000000
   COST AT STATION  1 =          3348.897000
PROC.COST FOR PROD. 1 OPER. 2=   300.000000
SETUP COST FOR PROD. 1 OPER. 2=    5.389246
TRANS.COST FOR PROD. 1 OPER. 2=    8.274569
FIXED COST FOR PROD. 1 OPER. 2=    651.302400
PROC.COST FOR PROD. 2 OPER. 2=  2500.000000
SETUP COST FOR PROD. 2 OPER. 2=   21.621000
TRANS.COST FOR PROD. 2 OPER. 2=   20.092460
FIXED COST FOR PROD. 2 OPER. 2=   1345.094000
PROC.COST FOR PROD. 3 OPER. 2=   450.000000
SETUP COST FOR PROD. 3 OPER. 2=    7.285656
TRANS.COST FOR PROD. 3 OPER. 2=    6.132021
FIXED COST FOR PROD. 3 OPER. 2=    950.506500
 MACHINES REQUIRED AT STATION 2=     15.000000
   COST AT STATION  2 =          6265.695000
PROC.COST FOR PROD. 1 OPER. 3=   500.000000
SETUP COST FOR PROD. 1 OPER. 3=    2.350440
TRANS.COST FOR PROD. 1 OPER. 3=   12.436720
```

FIXED COST FOR PROD. 1 OPER. 3= 226.790200

PROC.COST FOR PROD. 2 OPER. 3= 1839.999000

SETUP COST FOR PROD. 2 OPER. 3= 15.052790

TRANS.COST FOR PROD. 2 OPER. 3= 16.236320

FIXED COST FOR PROD. 2 OPER. 3= 1059.548000

PROC.COST FOR PROD. 3 OPER. 3= 1560.000000

SETUP COST FOR PROD. 3 OPER. 3= 12.037230

TRANS.COST FOR PROD. 3 OPER. 3= 9.216467

FIXED COST FOR PROD. 3 OPER. 3= 979.543700

MACHINES REQUIRED AT STATION 3= 12.000000

COST AT STATION 3 = 6233.210000

PROC.COST FOR PROD. 1 OPER. 4= 499.999700

SETUP COST FOR PROD. 1 OPER. 4= 3.615450

TRANS.COST FOR PROD. 1 OPER. 4= 12.524390

FIXED COST FOR PROD. 1 OPER. 4= 577.345400

PROC.COST FOR PROD. 2 OPER. 4= 799.999700

SETUP COST FOR PROD. 2 OPER. 4= 15.698990

TRANS.COST FOR PROD. 2 OPER. 4= 24.427790

FIXED COST FOR PROD. 2 OPER. 4= 964.003900

PROC.COST FOR PROD. 3 OPER. 4= 599.999700

SETUP COST FOR PROD. 3 OPER. 4= 6.806971

TRANS.COST FOR PROD. 3 OPER. 4= 15.453490

FIXED COST FOR PROD. 3 OPER. 4= 629.258500

MACHINES REQUIRED AT STATION 4= 11.000000

COST AT STATION 4 = 4149.132000

TOTAL COST FOR SEQUENCE 6= 19996.930000

TOTAL INVESTMENT COST FOR SEQUENCE 6=
2300.000000

```
*******************************************************

 SEQUENCE  COMBINATION  7 RESULTS
PROC.COST FOR PROD. 1 OPER. 1=   375.000000
SETUP COST FOR PROD. 1 OPER. 1=     1.717889
TRANS.COST FOR PROD. 1 OPER. 1=    10.198630
FIXED COST FOR PROD. 1 OPER. 1=     336.886200
PROC.COST FOR PROD. 2 OPER. 1=   750.000000
SETUP COST FOR PROD. 2 OPER. 1=     5.859856
TRANS.COST FOR PROD. 2 OPER. 1=    24.038940
FIXED COST FOR PROD. 2 OPER. 1=     731.295100
PROC.COST FOR PROD. 3 OPER. 1=   562.500000
SETUP COST FOR PROD. 3 OPER. 1=     4.831235
TRANS.COST FOR PROD. 3 OPER. 1=    21.376220
FIXED COST FOR PROD. 3 OPER. 1=     527.355900
 MACHINES REQUIRED AT STATION 1=       8.000000
   COST AT STATION  1 =          3351.059000
PROC.COST FOR PROD. 1 OPER. 2=  1250.000000
SETUP COST FOR PROD. 1 OPER. 2=     5.998573
TRANS.COST FOR PROD. 1 OPER. 2=    12.287510
FIXED COST FOR PROD. 1 OPER. 2=     585.821200
PROC.COST FOR PROD. 2 OPER. 2=  1599.999000
SETUP COST FOR PROD. 2 OPER. 2=    19.220120
```

TRANS.COST FOR PROD. 2 OPER. 2= 16.090310

FIXED COST FOR PROD. 2 OPER. 2= 1365.159000

PROC.COST FOR PROD. 3 OPER. 2= 450.000000

SETUP COST FOR PROD. 3 OPER. 2= 7.285656

TRANS.COST FOR PROD. 3 OPER. 2= 6.132021

FIXED COST FOR PROD. 3 OPER. 2= 950.506500

MACHINES REQUIRED AT STATION 2= 15.000000

COST AT STATION 2 = 6268.496000

PROC.COST FOR PROD. 1 OPER. 3= 250.000000

SETUP COST FOR PROD. 1 OPER. 3= 4.810224

TRANS.COST FOR PROD. 1 OPER. 3= 8.274419

FIXED COST FOR PROD. 1 OPER. 3= 581.336900

PROC.COST FOR PROD. 2 OPER. 3= 1600.000000

SETUP COST FOR PROD. 2 OPER. 3= 13.873010

TRANS.COST FOR PROD. 2 OPER. 3= 20.213970

FIXED COST FOR PROD. 2 OPER. 3= 857.885000

PROC.COST FOR PROD. 3 OPER. 3= 1560.000000

SETUP COST FOR PROD. 3 OPER. 3= 12.037230

TRANS.COST FOR PROD. 3 OPER. 3= 9.216467

FIXED COST FOR PROD. 3 OPER. 3= 979.543700

MACHINES REQUIRED AT STATION 3= 13.000000

COST AT STATION 3 = 5897.187000

PROC.COST FOR PROD. 1 OPER. 4= 299.999700

SETUP COST FOR PROD. 1 OPER. 4= 2.223121

TRANS.COST FOR PROD. 1 OPER. 4= 12.536990

FIXED COST FOR PROD. 1 OPER. 4= 354.649900

PROC.COST FOR PROD. 2 OPER. 4= 999.999700

SETUP COST FOR PROD. 2 OPER. 4= 20.636100

TRANS.COST FOR PROD. 2 OPER. 4= 24.452390

FIXED COST FOR PROD. 2 OPER. 4= 1265.895000

PROC.COST FOR PROD. 3 OPER. 4= 599.999700

SETUP COST FOR PROD. 3 OPER. 4= 6.806971

TRANS.COST FOR PROD. 3 OPER. 4= 15.453490

FIXED COST FOR PROD. 3 OPER. 4= 629.258500

MACHINES REQUIRED AT STATION 4= 12.000000

COST AT STATION 4 = 4231.910000

TOTAL COST FOR SEQUENCE 7= 19748.650000

TOTAL INVESTMENT COST FOR SEQUENCE 7= 2400.000000

****

SEQUENCE COMBINATION 8 RESULTS

PROC.COST FOR PROD. 1 OPER. 1= 375.000000

SETUP COST FOR PROD. 1 OPER. 1= 1.720988

TRANS.COST FOR PROD. 1 OPER. 1= 10.301830

FIXED COST FOR PROD. 1 OPER. 1= 334.113000

PROC.COST FOR PROD. 2 OPER. 1= 750.000000

SETUP COST FOR PROD. 2 OPER. 1= 5.859856

TRANS.COST FOR PROD. 2 OPER. 1= 24.038940

FIXED COST FOR PROD. 2 OPER. 1= 731.295100

PROC.COST FOR PROD. 3 OPER. 1= 562.500000

```
SETUP COST FOR PROD. 3 OPER. 1=    4.831235
TRANS.COST FOR PROD. 3 OPER. 1=   21.376220
FIXED COST FOR PROD. 3 OPER. 1=    527.355900
 MACHINES REQUIRED AT STATION 1=      8.000000
   COST AT STATION  1 =           3348.392000
PROC.COST FOR PROD. 1 OPER. 2=   300.000000
SETUP COST FOR PROD. 1 OPER. 2=    5.389246
TRANS.COST FOR PROD. 1 OPER. 2=    8.274569
FIXED COST FOR PROD. 1 OPER. 2=    651.302400
PROC.COST FOR PROD. 2 OPER. 2=  1599.999000
SETUP COST FOR PROD. 2 OPER. 2=   19.220120
TRANS.COST FOR PROD. 2 OPER. 2=   16.090310
FIXED COST FOR PROD. 2 OPER. 2=   1365.159000
PROC.COST FOR PROD. 3 OPER. 2=   450.000000
SETUP COST FOR PROD. 3 OPER. 2=    7.285656
TRANS.COST FOR PROD. 3 OPER. 2=    6.132021
FIXED COST FOR PROD. 3 OPER. 2=    950.506500
 MACHINES REQUIRED AT STATION 2=     15.000000
   COST AT STATION  2 =           5379.355000
PROC.COST FOR PROD. 1 OPER. 3=   500.000000
SETUP COST FOR PROD. 1 OPER. 3=    2.350440
TRANS.COST FOR PROD. 1 OPER. 3=   12.436720
FIXED COST FOR PROD. 1 OPER. 3=    226.790200
PROC.COST FOR PROD. 2 OPER. 3=  1600.000000
SETUP COST FOR PROD. 2 OPER. 3=   13.873010
```

```
TRANS.COST FOR PROD. 2 OPER. 3=   20.213970
FIXED COST FOR PROD. 2 OPER. 3=    857.885000
PROC.COST FOR PROD. 3 OPER. 3=  1560.000000
SETUP COST FOR PROD. 3 OPER. 3=   12.037230
TRANS.COST FOR PROD. 3 OPER. 3=    9.216467
FIXED COST FOR PROD. 3 OPER. 3=    979.543700
 MACHINES REQUIRED AT STATION 3=     11.000000
   COST AT STATION  3 =          5794.343000
PROC.COST FOR PROD. 1 OPER. 4=   499.999700
SETUP COST FOR PROD. 1 OPER. 4=    3.615450
TRANS.COST FOR PROD. 1 OPER. 4=   12.524390
FIXED COST FOR PROD. 1 OPER. 4=    577.345400
PROC.COST FOR PROD. 2 OPER. 4=   999.999700
SETUP COST FOR PROD. 2 OPER. 4=   20.636100
TRANS.COST FOR PROD. 2 OPER. 4=   24.452390
FIXED COST FOR PROD. 2 OPER. 4=   1265.895000
PROC.COST FOR PROD. 3 OPER. 4=   599.999700
SETUP COST FOR PROD. 3 OPER. 4=    6.806971
TRANS.COST FOR PROD. 3 OPER. 4=   15.453490
FIXED COST FOR PROD. 3 OPER. 4=    629.258500
 MACHINES REQUIRED AT STATION 4=     13.000000
   COST AT STATION  4 =          4655.984000
 TOTAL COST FOR SEQUENCE   8=        19178.070000
```

TOTAL INVESTMENT COST FOR SEQUENCE 8=
2350.000000

****

```
*******************************************************
 OPTIMAL GENERAL SEQUENCE COMBINATION = SEQUENCE    4
 OPTIMAL SEQUENCE NUMBER FOR ALL PRODS= 2 2 1
 OPTIMAL TOTAL COST=        18242.400000
NUMBER OF MACHINES: TYPE   1 =             8.000000
NUMBER OF MACHINES: TYPE   2 =            14.000000
NUMBER OF MACHINES: TYPE   3 =            11.000000
NUMBER OF MACHINES: TYPE   4 =            12.000000
 INVESTMENT COST=        2250.000000
 OPTIMAL SEQUENCE FOR PROD.   1  =     2
 SEQUENCE FOR PRODUCT   1  =   1 4 2 5
 THE MACHINES REQUIRED FOR PROD. 1=1.67  3.26  1.13  2.89
*******************************************************
 OPTIMAL SEQUENCE FOR PROD.   2  =     2
 SEQUENCE FOR PRODUCT   2  =   1 3 2 5
 THE MACHINES REQUIRED FOR PROD. 2=3.66  6.83  4.29  6.33
*******************************************************
 OPTIMAL SEQUENCE FOR PROD.   3  =     1
 SEQUENCE FOR PRODUCT   3  =   1 3 4 5
 THE MACHINES REQUIRED FOR PROD. 3=2.64  3.18  4.91  2.53
*******************************************************
```

Appendix C

PROGRAM LISTING

```
C
C*****MAIN PROGRAM
C
      COMMON /NAME1/ISET(130,10),TIVCST(130),OPTCST(130)
      COMMON /NAME2/OPTIVT(130),ACOST(130,10,10),CIVST(130)
      COMMON /NAME3/BCOST(130,10),TCOST(130),M(10),IPROD(130,10,10)
      COMMON /NAME4/ID(10),DEF(10,10,10),CN(10),IIS(10),BBNN(129,10)
      COMMON /NAME5/V(10,10,10),SK(10,10,10),T(10,10,10)
      COMMON /NAME6/CV(10,10,10),FREQ(10,10,10),N(10,130,10)
      COMMON /NAME7/W(10),WMAX(10),IIP,IIF(10),INVST,IH,C(10,10,10)
      COMMON /NAME8/IPP,MPP,CVST(130,10),FIX(7,7),TTN(10,10,10)
C*****READ NUMBER OF PRODUCTS,TIME REQUIRED AND INVESTMENT COST
C*****READ THE MAXIMUM NUMBER OF SEQUENCE COMBINATIONS & NUMBER OF OPER.
C
      READ(5,10) IIP,IH,INVST,IPP,MPP
      WRITE(6,110)
  110 FORMAT(2X,'NO.OF PROD.  OP.TIME  INVST COST  NO.SEQ  MAX.NO.OP.')
      WRITE(6,7766)
      WRITE(6,100) IIP,IH,INVST,IPP,MPP
      WRITE(6,7766)
      WRITE(6,7766)
```

```
      WRITE(6,7756)
      WRITE(6,7766)
      WRITE(6,7766)
      WRITE(6,7755) IIP,IPP,MPP
      WRITE(6,7766)
      WRITE(6,7766)
      WRITE(6,7756)
      WRITE(6,7766)
      WRITE(6,7766)
 7766 FORMAT(80X)
 7755 FORMAT(2X,I4,' PROD.',I3,' POSSIBLE SEQS  &',I3,' OPER./PROD.')
 7756 FORMAT(' +++++++++++++++++++++++++++++++++++++++++++++++++++++++')
   10 FORMAT(5I10)
C
C*****READ THE NUMBER OF OPERATIONS FOR EACH PRODUCT
C
      READ(5,25) (M(I),I=1,IIP)
      WRITE(6,120)
  120 FORMAT(2X,'NUMBER OF OPERATIONS FOR EACH PROD. ')
      WRITE(6,100) (M(I),I=1,IIP)
C
```

```
C*****READ THE NUMBER OF SEQUENCES FOR EACH PRODUCT
C
      READ(5,25) (IIS(IK),IK=1,IIP)
      WRITE(6,130)
  130 FORMAT(2X,'NUMBER OF SEQUENCES FOR EACH PRODUCT ')
      WRITE(6,100) (IIS(IK),IK=1,IIP)
C
C****READ THE OPERATION NUMBERS FOR EACH SEQUENCE FOR EACH PROD.
C
      WRITE(6,140)
  140 FORMAT(2X,'OPERATION NUMBERS FOR EACH SEQUENCE ')
      DO 6 JJ=1,IIP
      MVAR=M(JJ)
      NVAR=IIS(JJ)
      DO 6 NXA=1,NVAR
      READ(5,25) (IPROD(JJ,NXA,II),II=1,MVAR)
      WRITE(6,100) (IPROD(JJ,NXA,II),II=1,MVAR)
    6 CONTINUE
C
C*****READ DEMAND FOR EACH PRODUCT
C
```

```
      WRITE(6,150)
  150 FORMAT(2X,'DEMAND FOR EACH PRODUCT ')
      READ(5,25) (ID(IKK),IKK=1,IIP)
      WRITE(6,100) (ID(IKK),IKK=1,IIP)
C
C*****READ COST FOR EACH MACHINE
C
      WRITE(6,160)
  160 FORMAT(2X,'COST FOR EACH MACHINE ')
      DO 5 IJ=1,IIP
      KA=M(IJ)
      READ(5,35) (CN(J),J=1,KA)
      WRITE(6,105) (CN(J),J=1,KA)
    5 CONTINUE
C
C*****READ DEFECTIVES FOR EACH PRODUCT,SEQUENCE&&PPOSITION
C
      WRITE(6,170)
  170 FORMAT(2X,'DEFECTIVE FOR EACH PROD & SEQUENCE ')
      DO 20 IR=1,IIP
      JOHN=M(IR)
```

```
      JACK=IIS(IR)
      DO 20 IW=1,JACK
      READ(5,35) (DEF(IR,IW,JP),JP=1,JOHN)
      WRITE(6,105) (DEF(IR,IW,JP),JP=1,JOHN)
   20 CONTINUE
C
C*****READ PROCESS TIMES FOR EACH PRODUCT,SEQUENCE & POSITION
C
      WRITE(6,180)
  180 FORMAT(2X,'PROCESS TIMES FOR EACH PROCESS & SEQUENCE ')
      DO 30 JR=1,IIP
      JAMES=M(JR)
      JAKE=IIS(JR)
      DO 30 JW=1,JAKE
      READ(5,35) (V(JR,JW,IY),IY=1,JAMES)
      WRITE(6,105) (V(JR,JW,IY),IY=1,JAMES)
   30 CONTINUE
C
C*****READ THE SETUP TIME FOR EACH PRODUCT,SEQUENCE & POSITION
C
      WRITE(6,190)
```

```
  190 FORMAT(2X,'SETUP TIME FOR EACH PRODUCT & SEQUENCE ')
      DO 33 IQ=1,IIP
      IRAND=M(IQ)
      JRAND=IIS(IQ)
      DO 33 JZ=1,JRAND
      READ(5,35) (SK(IQ,JZ,NR),NR=1,IRAND)
      WRITE(6,105) (SK(IQ,JZ,NR),NR=1,IRAND)
   33 CONTINUE
C
C*****READ TRANSPORTATION COST FROM EACH WORK STATION J TO J+1
C
      WRITE(6,200)
  200 FORMAT(2X,'TRANSPORTATION COST FOR EACH PROD. & SEQ.')
      DO 34 JB=1,IIP
      IVAR=M(JB)
      MOB=IIS(JB)
      DO 34 JQ=1,MOB
      READ(5,35) (T(JB,JQ,JC),JC=1,IVAR)
      WRITE(6,105) (T(JB,JQ,JC),JC=1,IVAR)
   34 CONTINUE
C
```

```
C*****READ FIXED COST OF HAVING MACHINE J THAT PROCESSES PRODUCT I
C
      WRITE(6,210)
  210 FORMAT(2X,'FIXED COST HAVING MACHINE ')
      DO 36 IB=1,IIP
      IVA=M(IB)
      READ(5,35) (FIX(IB,IC),IC=1,IVA)
      WRITE(6,105) (FIX(IB,IC),IC=1,IVA)
   36 CONTINUE
C
C*****READ COST OF SETTING UP FOR EACH BATCH OF PROD. I ON MACHINE J
C
      WRITE(6,220)
  220 FORMAT(2X,'COST OF SETTING UP FOR EACH BATCH ')
      DO 38 IX=1,IIP
      IPAT=M(IX)
      IBAB=IIS(IX)
      DO 38 IBA=1,IBAB
      READ(5,35) (C(IX,IBA,ILA),ILA=1,IPAT)
      WRITE(6,105) (C(IX,IBA,ILA),ILA=1,IPAT)
   38 CONTINUE
```

```
C
C*****READ PROCESSING COST
C
      WRITE(6,230)
  230 FORMAT(2X,'PROCESSING COST ')
      DO 50 MX=1,IIP
      JPAT=M(MX)
      JBAB=IIS(MX)
      DO 50 JBA=1,JBAB
      READ(5,35) (CV(MX,JBA,JLA),JLA=1,JPAT)
      WRITE(6,105) (CV(MX,JBA,JLA),JLA=1,JPAT)
   50 CONTINUE
C
C*****READ  THE WEIGHT OF EACH PRODUCT
C
      READ(5,35) (W(MM),MM=1,IIP)
      WRITE(6,240)
  240 FORMAT(2X,'WEIGHT FOR EACH PRODUCT ')
      WRITE(6,105) (W(MM),MM=1,IIP)
C
C*****READ THE MAX WEIGHT ALLOWED PER BATCH OF EACH PRODUCT
```

```
C
      READ(5,35) (WMAX(NNI),NNI=1,IIP)
      WRITE(6,250)
  250 FORMAT(2X,'MAXIMUM WEIGHT FOR EACH PRODUCT TYPE ')
      WRITE(6,105) (WMAX(NNI),NNI=1,IIP)
C
C****READ THE SEQUENCE COMBINATION SETS FOR ALL PRODUCTS
C
      WRITE(6,260)
  260 FORMAT(2X,'SEQUENCE COMBINATION SETS FOR EACH PRODUCT ')
      DO 71 IG=1,IPP
      READ(5,25) (ISET(IG,KH),KH=1,IIP)
      WRITE(6,100) (ISET(IG,KH),KH=1,IIP)
   71 CONTINUE
   25 FORMAT(8I10)
   35 FORMAT(8F10.4)
  100 FORMAT(8I10)
  105 FORMAT(2X,8F10.4)
      CALL FREQCY
      CALL MACHIN
      CALL CALCLT
```

```
      CALL SEARCH
      STOP
      END
C*****
C*****
C*****
      SUBROUTINE FREQCY
      COMMON /NAME1/ISET(130,10),TIVCST(130),OPTCST(130)
      COMMON /NAME2/OPTIVT(130),ACOST(130,10,10),CIVST(130)
      COMMON /NAME3/BCOST(130,10),TCOST(130),M(10),IPROD(130,10,10)
      COMMON /NAME4/ID(10),DEF(10,10,10),CN(10),IIS(10),BBNN(129,10)
      COMMON /NAME5/V(10,10,10),SK(10,10,10),T(10,10,10)
      COMMON /NAME6/CV(10,10,10),FREQ(10,10,10),N(10,130,10)
      COMMON /NAME7/W(10),WMAX(10),IIP,IIF(10),INVST,IH,C(10,10,10)
      COMMON /NAME8/IPP,MPP,CVST(130,10),FIX(7,7),TTN(10,10,10)
      DO 20 I=1,IIP
      JOHN=M(I)
      JAKE=IIS(I)
      DO 25 JK=1,JAKE
      JJ=JOHN
      FREQQ=1.
```

```
   10 FREQQ=FREQQ*(1.-((DEF(I,JK,JJ))/100.))
      FREQ(I,JK,JJ)=FREQQ
      WRITE(6,40) I,JJ,FREQ(I,JK,JJ)
   40 FORMAT(2X,'DEFECTIVE OF PROD. ',I2,' AT OPERATION ',I2,'=',F8.4)
      JJ=JJ-1
      IF(JJ.EQ.0) GO TO 35
      GO TO 10
   35 FX=(1./(FREQ(I,JK,1)))*(W(I)*(ID(I)))*(1./(WMAX(I)))
      IF(FX.EQ.INT(FX)) GO TO 15
      IIF(I)=INT(FX)+1
      WRITE(6,60) I,IIF(I)
   60 FORMAT(2X,'FREQUENCY OF PROD. ',I3,'=',I4)
      GO TO 25
   15 IIF(I)=FX
      WRITE(6,30) I,IIF(I)
   30 FORMAT(2X,'FREQUENCY OF PROD. ',I3,'=',I4)
   25 CONTINUE
   20 CONTINUE
      RETURN
      END
C*****
```

```
C*****
C*****
      SUBROUTINE MACHIN
      COMMON /NAME1/ISET(130,10),TIVCST(130),OPTCST(130)
      COMMON /NAME2/OPTIVT(130),ACOST(130,10,10),CIVST(130)
      COMMON /NAME3/BCOST(130,10),TCOST(130),M(10),IPROD(130,10,10)
      COMMON /NAME4/ID(10),DEF(10,10,10),CN(10),IIS(10),BBNN(129,10)
      COMMON /NAME5/V(10,10,10),SK(10,10,10),T(10,10,10)
      COMMON /NAME6/CV(10,10,10),FREQ(10,10,10),N(10,130,10)
      COMMON /NAME7/W(10),WMAX(10),IIP,IIF(10),INVST,IH,C(10,10,10)
      COMMON /NAME8/IPP,MPP,CVST(130,10),FIX(7,7),TTN(10,10,10)
      DO 20 I=1,IIP
      JJ=M(I)
      JK=IIS(I)
      DO 20 IK=1,JK
      DO 20 IL=1,JJ
      R1=(V(I,IK,IL)*ID(I))/FREQ(I,IK,IL)
      R2=IH-((SK(I,IK,IL))*(FREQ(I,IK,IL)*IIF(I)))
      TNN=R1/R2
      TTN(I,IK,IL)=TNN
      IF(TNN.EQ.INT(TNN)) GO TO 25
```

```
      N(I,IK,IL)=INT(TNN)+1
      GO TO 20
   25 N(I,IK,IL)=TNN
   20 CONTINUE
      RETURN
      END
C*****
C*****
C*****
      SUBROUTINE CALCLT
      COMMON /NAME1/ISET(130,10),TIVCST(130),OPICST(130)
      COMMON /NAME2/OPTIVT(130),ACOST(130,10,10),CIVST(130)
      COMMON /NAME3/BCOST(130,10),TCOST(130),M(10),IPROD(130,10,10)
      COMMON /NAME4/ID(10),DEF(10,10,10),CN(10),IIS(10),BBNN(129,10)
      COMMON /NAME5/V(10,10,10),SK(10,10,10),T(10,10,10)
      COMMON /NAME6/CV(10,10,10),FREQ(10,10,10),N(10,130,10)
      COMMON /NAME7/W(10),WMAX(10),IIP,IIF(10),INVST,IH,C(10,10,10)
      COMMON /NAME8/IPP,MPP,CVST(130,10),FIX(7,7),TTN(10,10,10)
      DO 35 KK=1,IPP
      QCOST=0.
      WRITE(6,960) KK
```

```
  960 FORMAT(2X,'SEQUENCE  COMBINATION ',I2,' RESULTS')
      WRITE(6,364)
  364 FORMAT(80X)
      CK=0.
      COST=0.
      DO 30 J=1,MPP
      BNN=0.
      TCOST=0.
      NN=0
      DO 20 I=1,IIP
      K=ISET(KK,I)
      BNN=BNN+TTN(I,K,J)
C1    NN=NN+N(I,K,J)
C
C*****PROCESS COST
      PCOST=V(I,K,J)*ID(I)*CV(I,K,J)
      WRITE(6,900) I,J,PCOST
  900 FORMAT(' PROC.COST FOR PROD.',I2,' OPER.',I2,'=',F13.6)
C*****SETUP COST
      STCOST=C(I,K,J)*IIF(I)*TTN(I,K,J)*FREQ(I,K,J)
C2    STCOST=C(I,K,J)*IIF(I)*N(I,K,J)*FREQ(I,K,J)
```

```
      WRITE(6,901) I,J,STCOST
  901 FORMAT(' SETUP COST FOR PROD.',I2,' OPER.',I2,'=',F12.6)
C*****TRANSPORTATION COST
      TRANS=T(I,K,J)*FREQ(I,K,J)*IIF(I)
      WRITE(6,902) I,J,TRANS
  902 FORMAT(' TRANS.COST FOR PROD.',I2,' OPER.',I2,'=',F12.6)
C*****FIXED COST ASSOCIATED WITH OWNING MACHINE J
      FIXCST=FIX(I,J)*TIN(I,K,J)
C3    FIXCST=FIX(I,J)*N(I,K,J)
      WRITE(6,903) I,J,FIXCST
  903 FORMAT(' FIXED COST FOR PROD.',I2,' OPER.',I2,'=',F14.6)
      COSTT=PCOST+STCOST+TRANS+FIXCST
      ACOST(I,K,J)=COSTT
      FCOST=FCOST+COSTT
   20 CONTINUE
      IF(BNN.EQ.INT(BNN)) GO TO 1133
      BNN=INT(BNN) +1.
 1133 BBNN(KK,J)=BNN
      WRITE(6,1155) J,BNN
C4    WRITE(6,1155) J,NN
      QCOST=QCOST+FCOST
```

```
      WRITE(6,1122)
      WRITE(6,1156) J,FCOST
      WRITE(6,1122)
 1122 FORMAT(80X)
C6
 1155 FORMAT(2X,'MACHINES REQUIRED AT STATION',I2,'=',F14.6)
C1155 FORMAT(2X,'MACHINES REQUIRED AT STATION',I2,'=',I8)
 1156 FORMAT(2X,'  COST AT STATION ',I2,' = ',F20.6)
      BCOST(KK,J)=QCOST
      COST=BCOST(KK,J)
      CVST(KK,J)=CN(J)*BNN
C5    CVST(KK,J)=CN(J)*NN
      CK=CK+CVST(KK,J)
   30 CONTINUE
      TCOST(KK)=COST
      TIVCST(KK)=CK
      WRITE(6,400) KK,TCOST(KK)
      WRITE(6,410) KK,TIVCST(KK)
  400 FORMAT(2X,'TOTAL COST FOR SEQUENCE ',I3,'=',F20.6)
  410 FORMAT(2X,'TOTAL INVESTMENT COST FOR SEQUENCE ',I3,'=',F20.6)
      WRITE(6,977)
```

```
  977 FORMAT(' ***********************************************************')
      WRITE(6,522)
      WRITE(6,522)
      WRITE(6,522)
  522 FORMAT(80X)
   35 CONTINUE
      RETURN
      END
C*****
C*****
C*****
      SUBROUTINE SEARCH
      COMMON /NAME1/ISET(130,10),TIVCST(130),OPTCST(130)
      COMMON /NAME2/OPTIVT(130),ACOST(130,10,10),CIVST(130)
      COMMON /NAME3/BCOST(130,10),TCOST(130),M(10),IPROD(130,10,10)
      COMMON /NAME4/ID(10),DEF(10,10,10),CN(10),IIS(10),BBNN(129,10)
      COMMON /NAME5/V(10,10,10),SK(10,10,10),T(10,10,10)
      COMMON /NAME6/CV(10,10,10),FREQ(10,10,10),N(10,130,10)
      COMMON /NAME7/W(10),WMAX(10),IIP,IIF(10),INVST,IH,C(10,10,10)
      COMMON /NAME8/IPP,MPP,CVST(130,10),FIX(7,7),TTN(10,10,10)
   30 CMIN=10000000.
```

```
      DO 10 I=1,IPP
      IF(TCOST(I).LE.CMIN) K=I
   10 IF(TCOST(I).LE.CMIN) CMIN=TCOST(I)
      IF(TIVCST(K).LE.INVST) GO TO 40
      BMIN=100000000.
      DO 20 J=1,IPP
      IF(TIVCST(J).LE.BMIN) MIX=J
   20 IF(TIVCST(J).LE.BMIN) BMIN=TIVCST(J)
      IF(BMIN.GT.INVST) GO TO 90
      GO TO 12
   90 WRITE(6,55)
      WRITE(6,911)
      WRITE(6,911)
  911 FORMAT(80X)
   55 FORMAT(2X,'PROBLEM IS INFEASIBLE!!!')
      RETURN
   12 TCOST(K)=100000000.
      CMIN=TCOST(MIX)
      GO TO 30
   40 WRITE(6,246)
      WRITE(6,8877)
```

```
      WRITE(6,7766) K
      WRITE(6,653)
 7766 FORMAT(2X,'OPTIMAL GENERAL SEQUENCE COMBINATION = SEQUENCE  ',I3)
      WRITE(6,70) (ISET(K,IK),IK=1,IIP)
      WRITE(6,653)
      WRITE(6,653)
      WRITE(6,653)
      WRITE(6,77) TCOST(K)
      WRITE(6,653)
      DO 37 IX=1,MPP
      WRITE(6,39) IX,BBNN(K,IX)
   39 FORMAT(' NUMBER OF MACHINES: TYPE  ',I2,' =',F20.6)
   37 CONTINUE
      WRITE(6,653)
      WRITE(6,80) TIVCST(K)
      WRITE(6,653)
      WRITE(6,653)
      DO 25 IB=1,IIP
      MBART=ISET(K,IB)
      WRITE(6,50) IB,MBART
      WRITE(6,653)
```

```
      WRITE(6,653)
      NP=M(IB)
      WRITE(6,65) IB,(IPROD(IB,MBART,JN),JN=1,NP)
      WRITE(6,653)
      WRITE(6,653)
      WRITE(6,653)
      WRITE(6,653)
      WRITE(6,653)
      WRITE(6,653)
      WRITE(6,85) IB,(TTN(IB,MBART,JM),JM=1,NP)
      WRITE(6,653)
      WRITE(6,653)
      WRITE(6,653)
      WRITE(6,8877)
 8877 FORMAT(' ********************************************************')
  653 FORMAT(80X)
   25 CONTINUE
   80 FORMAT(2X,'INVESTMENT COST=',F20.6)
   77 FORMAT(2X,'OPTIMAL TOTAL COST=',F20.6)
   70 FORMAT(2X,'OPTIMAL SEQUENCE NUMBER FOR ALL PRODS=',10I2)
   50 FORMAT(2X,'OPTIMAL SEQUENCE FOR PROD. ',I3,'  = ',I5)
```

```
   65 FORMAT(2X,'SEQUENCE FOR PRODUCT ',I3,'  =  ',10I2)
   85 FORMAT(2X,'THE MACHINES REQUIRED FOR PROD.',I2,'=',10(F4.2,2X))
      RETURN
      WRITE(6,246)
  246 FORMAT(1H1)
      END
//DATA
```

The vita has been removed from the scanned document