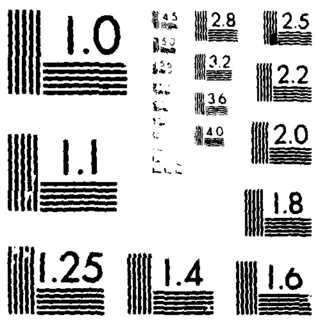
AD-A084 062 ARMY MILITARY PERSONNEL CENTER ALEXANDRIA VA F/G 13/8
A GENERALIZED MACHINE REQUIREMENTS PLANNING ALGORITHM FOR SERIA--ETC(U)
JUL 78 G M HAYES

UNCLASSIFIED NL

1 OF 2
AD
A084 062



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

ADA084062

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO. AD-A084062	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A Generalized Machine Requirements Planning Algorithm for Serial Flow Machining Systems		5. TYPE OF REPORT & PERIOD COVERED Final Report, July 1978
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Glenn M. Hayes, Jr., CPT, OrdC		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Student, HQDA, MILPERCEN (DAPC-OPP-E) 200 Stoval Street, Alexandria, VA 22332		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS HQDA, MILPERCEN, ATTN: DAPC-OPP-E, 200 Stovall Street, Alexandria, VA 22332		12. REPORT DATE July 1978
		13. NUMBER OF PAGES 128
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

LEVEL II

16. DISTRIBUTION STATEMENT (of this Report)

Approved for Public Release, distribution unlimited.

DTIC ELECTE MAY 8 1980 S C D

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

Document is a thesis submitted to the Graduate Faculty of the Virginia Polytechnic Institute and State University in partial fulfillment of the requirements for the degree of Master of Science in Industrial Engineering and Operations Research.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Machining Parameters
Manufacturing Technology
Optimization
Dynamic Programming
Machine Requirements
Parametric Analysis
Manufacturing Processes
Linear Programming
Serial Machining Systems

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The machine requirements planning problem is one which is applicable throughout the manufacturing industry. The use of automation and computer technology makes the use of machine requirements modelling particularly attractive, and the vast capital investment in machining equipment offers significant opportunity for savings through optimization. The machine requirements problem must be modelled as a mixed integer linear

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Block 20 (Continued)

program. However, a dynamic programming solution procedure with linear programming postoptimality techniques at each stage of the machining process is utilized to provide an efficient, flexible algorithm. The machine requirements model is capable of optimizing any type of machining system, whether of a discrete or continuous operating mode. Through discretizing the machine parameters problem, by limiting tool changes to between passes of a tool, any continuous mode system is approximated as discrete for optimization purposes and a direct interface with the machine requirements planning model is provided.

The dynamic programming solution procedure is compared with a mixed integer procedure. The DP formulation is not only more efficient in both time and core but provides sensitivity information and offers a broad spectrum of further application into more complex aspects of manufacturing systems. A survey of machine parameter and machine requirements literature is included in the thesis.

A Generalized Machine Requirements Planning Algorithm for Serial Flow Machining Systems

Glenn M. Hayes, Jr., CPT
HQDA, MILPERCEN (DAPC-OPP-E)
200 Stovall Street
Alexandria, VA 22332

Final Report - July 1978

A thesis submitted to Virginia Polytechnic Institute and State University in partial fulfillment of the requirements for the degree of Master of Science in Industrial Engineering and Operations Research.

A GENERALIZED MACHINE REQUIREMENTS PLANNING ALGORITHM FOR SERIAL FLOW MACHINING SYSTEMS

by

Glenn M. Hayes, Jr.

Thesis submitted to the Graduate Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

in

Industrial Engineering and Operations Research

APPROVED:

R. P. Davis, Chairman

J. W. Schmidt

L. C. Frair

R. A. Wysk

July, 1978

Blacksburg, Virginia

ACKNOWLEDGMENTS

I would like to express my appreciation and acknowledge the outstanding assistance and support offered by my thesis committee throughout the research effort. Of particular note is the guidance and timely advice of my committee chairman, Dr. Robert P. Davis, who was tireless in insuring that I stayed "on track" and attained a meaningful and challenging learning experience.

I would also like to acknowledge the constant support and understanding of my wife, Susan, and my children, Catherine and David. Their efforts were a major contribution to my completion of the research.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter I

INTRODUCTION

Manufacturing industries which utilize metal machining have incorporated an increasing level of automation in recent years. Both direct numerical control and computer numerical control systems can be found in many diverse areas of manufacturing [2,4]. It can be stated that the predominant reason for the use of these automated systems has been for control of the machining process and not as a means of integrating the total system in an optimum fashion. A major reason for this discrepancy is the lack of mathematical models which effectively optimize the manufacturing system. A consequence of this narrow use of the computer in manufacturing systems is considerable unused capacity which could be used for an aid in increasing the productivity and cost effectiveness of these systems.

An area in which automation can contribute significantly to improving machining systems' effectiveness is in the area of machine requirements planning. In general, the machine requirements (MR) planning problem can be defined as the specification of the number of each type of machine required in a production process, or group of processes, in each period during some planning horizon [18]. The significance of this problem is highly dependent on the type of production process and its complexity. Previous research on this problem has mostly been narrow in scope. The production systems considered have been relatively simple in that little consideration is given to the interrelationships of system functions and components. Attempts to realistically portray the production system have led to complex solution procedures such as those of references [6], [7], and [19]. The nature of the MR problem is such that it must be modelled as a mixed integer problem. Only in this way can the number of machines decision be meaningful. Integer linear programming procedures require a generous amount of time and core for even simple problems. Therefore, the solution procedures themselves have restricted the degree of complexity of the models.

The information required to develop a machine requirements model includes the operating rates of the individual machines within machine centers. The cost associated with a production rate and the percentage of defects related to production are also data which allow the formulation of a realistic model for minimizing cost in a production system. However, the operating rates are the key element which effect a viable model. Manufacturing machinery is either designed on an alternating current (AC) or a direct current (DC) basis. The use of alternating current usually results in discrete operating rates for the machine while direct current leads to rates from a continuum within a feasible range. The possibility of choosing from an infinite number of rates causes significant complexity in an optimization procedure, particularly one involving multistage operation. It would, therefore, be a distinct advantage if operating rate information for a machine could be discretized to further the feasibility of optimization.

When machining a particular product, the operating specifications of a machine such as the feed rate, depth of cut and speed of the cutter must be optimally chosen to manufacture a product in a cost effective manner. These operating parameters are what determine the production rate for a machine, the same rates needed for the machine requirements model. Computer technology makes it possible to more effectively determine these specifications, particularly where DC machines are concerned. This class of problems can be referred to as machine parameter (MP) optimization problems. Considerable work is documented in the literature on developing MP models but the bulk of this research has led to models which are nonlinear. Solution procedures such as geometric programming [25] or penalty function techniques [30] have proven to be either extremely difficult to apply or are slow to converge to an optimum.

While AC machinery has discrete rates and associated discrete machine parameter specifications, an MR model of broad applicability must be able to incorporate DC machines as well. A critical aspect of the MR model is that optimizing each stage of the process independently does not necessarily optimize the entire system. The MR model, therefore, needs complete operating rate information for each machining center (or stage) if the entire system is to be optimized. Consequently, an associated area of research for this thesis is to develop a mathematical model for optimizing machining parameters which provides complete discretized data. A basic assumption in this respect of the research is that a tool will only be changed between passes of a tool over a workpiece. This assumption reduces much of the nonlinearity in the problem and more importantly, leads to a discretized solution procedure using an efficient search algorithm. With discrete operating rates available, the machine requirements model can be applied to any type of machining system, whether powered by a discrete or continuous source.

The major purpose of this research is to develop a mathematical model of the machine requirements problem which achieves the following objectives:

1. Describes an integrated, serial manufacturing system.

2. Can be utilized to optimize any type of machining system, whether of a discrete or continuous operating mode.

In addition, there is the associated objective of developing a solution algorithm which is efficient in both solution time and core requirements as well as providing sensitivity information with respect to *final demand requirements*.

The research will be presented in a sequence that begins with a survey of the literature in the areas of machine parameter optimization and machine requirements planning. Following this is a description of the machine parameters model and solution procedures. The machine requirements model and its solution procedure are then presented. This formulation is tested and the results are analyzed in relation to a more complex integer programming procedure. Algorithm descriptions and information flow charts are provided to assist in the understanding of the models and associated solution procedures.

Chapter II

LITERATURE SURVEY

2.1 Introduction

This literature survey will be presented in two distinct parts. The development of machine parameter optimization techniques will be discussed initially. Then the status of machine requirements models will be reviewed.

2.2 Machine Parameter Optimization

There have been many contributions to the machine requirements optimization literature throughout the world. The most notable research has been carried out in Russia, Japan and the United States. An observation which should be made at this time is that machine parameter models are generally designed using one of three criteria: minimum production cost per part, minimum production time per part or maximum profit rate. This review will not concentrate on any particular criterion since any of the criteria can be converted to another with minor modification of the model.

All machine parameter optimization modelling requires some simplifying assumptions. One of the most basic models was developed by Wu and Ermer [31] who found the optimum machining parameters for a turning operation without considering any constraints and by varying one variable at a time. Ermer [9] solved the constrained machining problem in an economic context using geometric programming. Two years later, Petropoulos [25] developed his model for the optimal selection of machining parameters using geometric programming. Hati and Rao [12] developed mathematical programming models using SUMT techniques to determine the optimum machining parameters using all three of the criteria mentioned above. Their research concerns deterministic and probabilistic approaches and involves sensitivity analysis in relation to the cost coefficients. Hati and Rao document significant conclusions but their mathematical programming technique, although general, can be complex to apply and inefficient in operation.

Ermer [10] developed a Bayesian approach to machining economics by adaptive control. However, this approach requires precise monitoring of tool wear, a technique which has had only limited development. A significant model on adaptive searching systems for machining was developed by Lishchinski and Moshkov [17]. Their model explores the relationship between tool life, operating parameters and the power equation to develop a computerized search algorithm used to vary the feed and speed parameters. Although appearing to be a valid approach, this model is highly complex and involves searching over a wide range of values for the parameters. In addition, the algorithm requires the precise monitoring of tool wear. Davis, Wysk and Agee [8] developed a simplified approach to determining optimum machining parameters in an adaptive control context. Their procedure used an efficient iterative search algorithm to reach an optimum, but discrete operating data cannot be derived from their model.

An automated planning process for the optimization of machining processes was developed by Challa and Berra [4]. Their model used transformed functions to achieve greater linearity and is solved using a gradient projection method. The well behaved nature of the objective function leads to an efficient solution procedure but the procedure cannot be extended to other machine operations in such an efficient fashion. The model is developed only for a milling operation. Hitomi [12] discussed the optimal machining conditions for a multistage system using all three optimization criteria. However, the assumptions in his analysis were so restrictive as to render the staging inconsequential. A purely probabilistic approach was taken by Iwata, Murotsu, Iwatsubo and Fujii [13] in which chance-constrained programming was used. Although it was shown that the optimum cutting conditions are significantly affected by probabilistic considerations, the nature of parameter estimates is inherently imprecise and the assumption of normality is extremely important but substantiated by little empirical data.

Several other models have been developed which have very restrictive applications or extremely complex solution procedures. Ravignani [26] attempts to solve optimum machining conditions minimizing cost while producing to a fixed demand. This is actually a segmented approach where subsystems are isolated for individual optimization and then the effects of idle time and other disturbing events are discussed. There is no integrated systems approach for optimization. Boothroyd and Rusek [3] discussed the maximum rate of profit criteria for determining cutting speed. Their treatment results in the conclusion that the maximum rate of profit criteria is a compromise between the conditions of minimum cost and minimum production time. The conclusion is also drawn that the usefulness of their procedure is limited by inaccurate tool life data and little improvement can be expected in machining economics until greater sophistication is achieved in tool life models. Claycombe and Sullivan [5] contribute the use of response surface methodology to selecting a cutting tool in order to maximize profit. This method requires extensive testing, data collection and regression analysis and would only prove feasible to companies with extensive use of the same cutting tools and the capability for analysis.

A comparison of nonlinear programming techniques as applied to the machine parameter optimization problem is provided by Kimbler, Wysk and Davis [16]. From this analysis, SUMT and a general penalty function approach posed by the authors appear to be the most useful; but both are limited by complexities in the objective and constraint functions and the choice of penalty term values. If nonlinear programming techniques are to be more generally applicable to the machine parameter problem, much additional research is necessary to improve efficiency and broaden their feasibility. Table 2.1 is a compilation of the most significant contributions to the MP problem.

2.3 The Machine Requirements Problem

Only a limited number of procedures for analyzing the machine requirements problem have appeared in the literature. Most of the studies have been deterministic, static analyses which use a descriptive rather than a normative approach. Shubin and Madeheim [29] developed a model for a single product, single work center with one operation. Such a

Table 2.1 Machining Parameters Models

Author/Reference		Technique	Model Structure	Machine Operation
Hati and Rao	[12]	SUMT	Unit Cost, Time and Profit	Turning
Barash and Berra	[2]	Sectioning Search	Unit Time	Turning
Kimbler, Wysk and Davis	[16]	Exterior Penalty Function	Unit Time	Turning
Davis, Wysk and Agee	[8]	One-at-a-time Search	Unit Time	Turning
Lishchinski and Moshkov	[17]	Computerized Searching System	Unit Time	Drilling
Challa and Berra	[4]	Logarithmic Transformation	Unit Cost	Face Milling
Ermer	[10]	Bayesian Statistics	Unit Cost	Turning
Iwata *et al*.	[14]	Chance-Constrained Programming	Unit Cost	Turning
Petropoulos	[25]	Geometric Programming	Unit Cost	Turning

single period model can meet a prescribed production requirement but has a very limited systems scope. Similar descriptive, deterministic approaches have been reported by Apple [1], Muther [23], Johnson [15], Reed [27], and Moore [20]. Although most of the early work was concerned with the single product, single work center, single operation situation, Apple [1] and Francis and White [11] did extend the analysis to multiple work centers. A serial, single product, fixed routing operation is theorized but no production system component interaction is considered.

Morris [21] developed a probabilistic approach to the MR problem of a single product. Production requirements, operation performance time and machine effectiveness were modelled as random variables but no explicit form of the distributions was developed. Reed [28] also treated the static, descriptive problem from a probabilistic viewpoint. His multiple work center, serial flow system has the objective of finding the overall number of machines to meet a production requirement. Specific distribution functions for the basic parameters were assumed and not developed.

Morris [22] contributed the earliest normative approach reported in the literature. His decision model was based on a linear cost criterion and no constraints. The machine requirements were modelled as a probability distribution which was unspecified. All of the models to this point have resulted in continuous solutions where the actual number of machines required results from a round up or round down procedure. Davis and Miller [6] approached the MR problem using deterministic, steady-state analysis in a normative model. The formulation is based on meeting production requirements in a single planning period at minimum cost with the number of machines constrained to be an integer. An integer linear programming model was developed but a decision criteria was formulated which precluded the use of the more complex and costly ILP model. However, the decision tree type of analysis can be very tedious for multistage problems. The model, however, does include tradeoffs for investment cost and straight time and overtime operating and maintenance costs. Thus, MR models now involved more realistic considerations and became more feasible with the stagewise, integer constrained format.

Miller and Davis [19] also formulated a dynamic resource allocation model for the MR problem. Limitations on such resources as capital budget, floor space and overtime hours were explicitly considered and the interaction between machine centers was represented by consumption of these resources. The problem then becomes a deterministic resource allocation problem constrained to attain integer machine requirements. A drawback in the formulation is that the results represent changes in machine levels and the initial number required must be calculated to satisfy time phased production requirements. The model is therefore restricted to consider only a constant machine rate at each center.

A significant step forward in MR analysis was made by Davis and Miller [7]. This model formulation resolves the integer number of machines required and their discrete rates of operation for a discretely distributed demand in a minimum cost framework. The model developed is for a serial, multistage manufacturing system which has both straight time and overtime operating periods. Also provided for is the determination of in-process inventories between stages. No previous models were designed to consider the cost of variable operating rates and integration of the production and inventory systems. The model can be reoptimized if the demand distribution changes in order to update a decision policy for addition or disposal of machines. Table 2.2 presents a consolidation of major contributions to the solution of machine requirements problems.

Although considerable research has been documented on both the machine parameter and machine requirements problems, no definitive analysis appears in the literature for integration of the two problems into a single optimization model. The optimum cost of operating a machining system is directly related to the minimum cost framework of the machine parameters problem. However, to incorporate the two problems into a single model in the most widely applicable manner, requires that both discrete and continuous machining systems be modelled. As noted in the survey of the MP literature, the nonlinearity of the models greatly complicates the solution procedures and a method of discretizing all MP problems is required if a valid machine requirements model is to be formulated. To achieve such an integration of the two optimization models would be a major step forward in optimization of the total manufacturing system.

Table 2.2 Machine Requirements Models

Author/Reference		Model Type	Application
Shubin and Madeheim	[29]	Descriptive Deterministic	Single work center, single product, single period, continuous solution.
Apple	[1]	Descriptive Deterministic	Multiple work centers, serial system, single product but no interaction.
Francis and White	[11]	Descriptive Deterministic	Multiple work centers, serial system, single product but no interaction.
Morris	[21]	Descriptive Probabilistic	Single product with parameters as random variables, continuous solution.
Reed	[28]	Descriptive Probabilistic	Serial system with multiple work centers, specific distributions assumed, continuous.
Morris	[22]	Normative Probabilistic	Linear cost criterion with no constraints for multiple work center serial system, continuous solution.
Davis and Miller	[6]	Normative Deterministic	Multiple work center serial system with integer solution through a decision tree.
Miller and Davis	[19]	Normative Deterministic	Dynamic resource allocation model with changes to machine levels operating at a constant rate, integer solution.
Davis and Miller	[7]	Normative Deterministic	Serial, multistage system with discretely distributed production demand, in-process inventories, and an integer solution.

Chapter III

MACHINE PARAMETERS OPTIMIZATION MODEL

3.1 Introduction

For a machine requirements model to be generally applicable to any type of machining system, whether AC or DC, it is necessary to determine a set of operating rates, associated production costs and the percent of defects for a particular operating rate. A machine parameters model is to be developed which achieves this information in a form which can be readily applied to the machine requirements model.

The percent of defects is analogous to the probability of a defective part and little definitive work has been done in this area. The data necessary for inclusion of defectives in the model must be determined from: operational data from a similar production system, experimental data from sample production runs using statistical methods, or incorporated into the model after the actual production system is operating. In any event, the probability of defects occurring would be a highly dynamic term and would need to be updated at frequent intervals to maintain viability of the model. The cost of defectives will be initially defined in the MP model formulation but will not be pursued further in the research except as it relates to the machine requirements model. Table 3.1 presents a list of notations used in the formulation of the machining parameters optimization model.

Table 3.1 Notation for Machine Parameters Optimization Model

d_c = maximum depth of cut to be removed (mm)

d = depth of cut per tool pass (mm) where $d \varepsilon \{d_1, d_2, \ldots, d_N\}$

f = feed rate (mm/rev)

V = cutting speed (mm/min)

L = length of cut (mm)

D = diameter of cutting tool (mm)

M = torque produced on the tool (kgM)

T_h = tool thrust (kg)

C_I = idle time cost ($/min)

n = number of passes made with the tool

C = Taylor tool life constant

T_R = time required to replace a tool (min)

T_L = tool life (min)

P = power at the tool (kw)

C_p = cost per part ($)

C_c = cost of cutting ($/min)

C_T = cost of changing a tool ($/min)

n_p = number of tool passes per part

n_t = number of tool changes per pass

T_p = time per pass (min)

3.2 Problem Specification

Specifically, the machine parameter optimization problem is one of determining the operating parameters of feed, speed, and depth of cut in order to optimize a particular manufacturing objective. The objective considered in this research is one of minimum cost of production per part. The total cost to machine one part (C_p) can be represented by the following expression,

$$C_p = C_I T_I + C_C T_C(V,f,d) + C_T T_T(V,f,d) + C_d P_d(V,f,d) \qquad (3.1)$$

where,

T_I = Machine idle time due to setup, load and unload operations and idle tool motions, a constant.

T_C = Time in cut which is the expected time per pass times the required number of passes.

T_T = Tool change time which is the expected number of tool changes per part times the replacement time.

P_d = Probability of a defective part as a function of V, f and d.

C_I, C_C, C_T, C_d = Cost per minute for idle time, cutting time, tool replacement time and defectives, respectively.

The number of passes per part is restricted to being an integer and is defined by the expression,

$$np = \frac{d_c}{d}, \text{ an integer where } d \epsilon \{d_1, d_2, \ldots, d_N\}. \qquad (3.2)$$

Based on the assumption that it is infeasible to change a tool during a pass, the number of tool changes per pass can be defined as,

$$n_t = \frac{\text{time per pass}}{\text{tool life}} = \frac{T_P}{T_L} \tag{3.3}$$

where both numerator and denominator are bounded and the feasible set of n_t values is lower bounded by an integer fractional value and upper bounded by one. Thus,

$$n_{t_{min}} = \frac{T_{P_{min}}}{T_{L_{max}}} = \frac{1}{[T_{L_{max}}/T_{P_{min}}]} \tag{3.4}$$

$$n_{t_{max}} = 1, \tag{3.5}$$

since tool life is required to be at least equal to the time required for one pass. The number of tool changes per part is defined by,

$$n_T = n_p \cdot n_t \tag{3.6}$$

and $T_T = n_T \cdot T_R$, where T_R is the tool replacement time, or

$$T_T = n_p \cdot n_t \cdot T_R \tag{3.7}$$

Utilizing these relationships, the objective function (3.1) can now be written as,

$$C_P = C_I T_I + C_C \cdot T_P \cdot n_p + C_T \cdot T_R \cdot n_t \cdot n_p \tag{3.8}$$

subject to a set of constraints which will be illustrated here for turning and boring of mild steel alloys. The origin of these constraints will not be discussed here but adequate development appears in other sources, such as references [1] and [10]. The constraining relationships are:

$n_p \in \{Np\}$, where Np is the feasible set of n_p values,

$n_t \in \{Nt\}$, where Nt is the feasible set of n_t values.

$f_{min} \leq f \leq f_{max}$		(3.9)
$V_{min} \leq V \leq V_{max}$	limit on spindle speed	(3.10)
$T_{L_{min}} \leq TL \leq T_{L_{max}}$	limit on feasible T_L	(3.11)
$M = 0.084\, f^{.75} \cdot d \cdot D \leq M_{max}$	limit on spindle torque	(3.12)
$T_h = 2.02\, f^{.72} \cdot d \leq T_{h_{max}}$	limit on tool thrust	(3.13)
$P = 0.746\, M \cdot V/4500 \leq P_{max}$	limit on available power	(3.14)
$SF = g(V,f,d) \leq SF_{max}$	specific functional limit on achieving a desired surface finish	(3.15)

The optimization model will consider only initial cuts up to the final pass; and, therefore, surface finish considerations will not be included. With these considerations and the realization that idle time is a constant, the objective function can now be condensed to,

$$C_P = (n_p \cdot n_t)(C_C \cdot T_L + C_T \cdot T_R) \tag{3.16}$$

Utilizing the familiar expanded Taylor tool life equation,

$$T_L = \frac{C}{V^{\alpha} \cdot f^{\beta} \cdot d^{\gamma}} \tag{3.17}$$

and the time required for a single pass expression,

$$T_p = \frac{\pi \cdot D \cdot L}{V \cdot f} , \tag{3.18}$$

the following relationships can be developed:

$$V = \frac{C^{1/\alpha}}{T_L{}^{1/\alpha} \cdot f^{\beta/\alpha} \cdot d^{\gamma/\alpha}} , \tag{3.19}$$

and, $T_L = \frac{T_p}{T_L} = \frac{\pi \cdot D \cdot L}{n_t \cdot V \cdot f}$, or in expanded form as

$$T_L = \frac{\pi \cdot D \cdot L \cdot n_t{}^{-1} \cdot T_L{}^{1/\alpha} \cdot f^{\beta/\alpha - 1} \cdot d^{\gamma/\alpha}}{C^{1/\alpha}} . \tag{3.20}$$

Solving for feed (f), we obtain,

$$f = \left[\frac{T_L{}^{(1 - 1/\alpha)} \cdot C^{1/\alpha} \cdot n_t \cdot n_p{}^{\gamma/\alpha}}{\pi \cdot D \cdot L \cdot d_c{}^{\gamma/\alpha}}\right]^{\left[\frac{1}{(\beta/\alpha - 1)}\right]} \tag{3.21}$$

Consequently, for any given value of n_p, n_t, and T_L, the machining feed and speed can be determined from an optimization procedure developed using these relationships. The objective function (3.16) is to be minimized subject to a set of constraints given by equations (3.9) through (3.14) and the integer restrictions of (3.2) and (3.3). The model is now in a form whereby an iterative search procedure can be used to take advantage of the discrete nature of the variables n_t and n_p and the sectional convexity of the objective function.

3.3 Illustrative Example

The machine parameter optimization problem, as defined, has been solved using an example problem with the following data:

$D = 100$ mm, $L = 1000$ mm, $d_c = 5$ mm, $d_{min} = 1.2$ mm, $d_{max} = 2.75$ mm,

$f_{min} = .3$ mm/rev, $f_{max} = .75$ mm/rev, $V_{min} = 5$ m/min,

$V_{max} = 4 \times 10^5$ mm/min, $P_{max} = 2.25$ kW, $C = 6 \times 10^{26}$, $T_{L_{min}} = 25$ min,

$T_{L_{max}} = 45$ min, $\alpha = 5.0$, $\beta = 1.75$, $\gamma = 0.75$, $T_R = 1.5$ min,

$T_{h_{max}} = 60$ kg, $M_{max} = 47.5$ kgM, $C_C = .3$ \$/min, $C_T = .5$ \$/min.

The basis for this example data is a carbide tool and mild steel workpiece upon which a turning operation is performed. In observing the problem constraints, it can be seen that substitutions can be readily made using equations (3.2), (3.17), and (3.21). These substitutions result in constraint equations written in terms of n_t, n_p, and T_L. Since n_t and n_p are integer terms, the most simplifying step is to rewrite all of the constraints in terms of T_L as the bounded variable. With constraints (3.9), (3.10), (3.12), (3.13), and (3.14) altered in this way and (3.2), (3.3) and (3.11) left unaltered, the problem can now be written in a multistage decision context with an explicit optimization rationale at only one stage. This staged representation is illustrated in Figure 3.1.

In this staged context, $\underline{S}_3\langle\cdot\rangle$ represents the vector of bounding

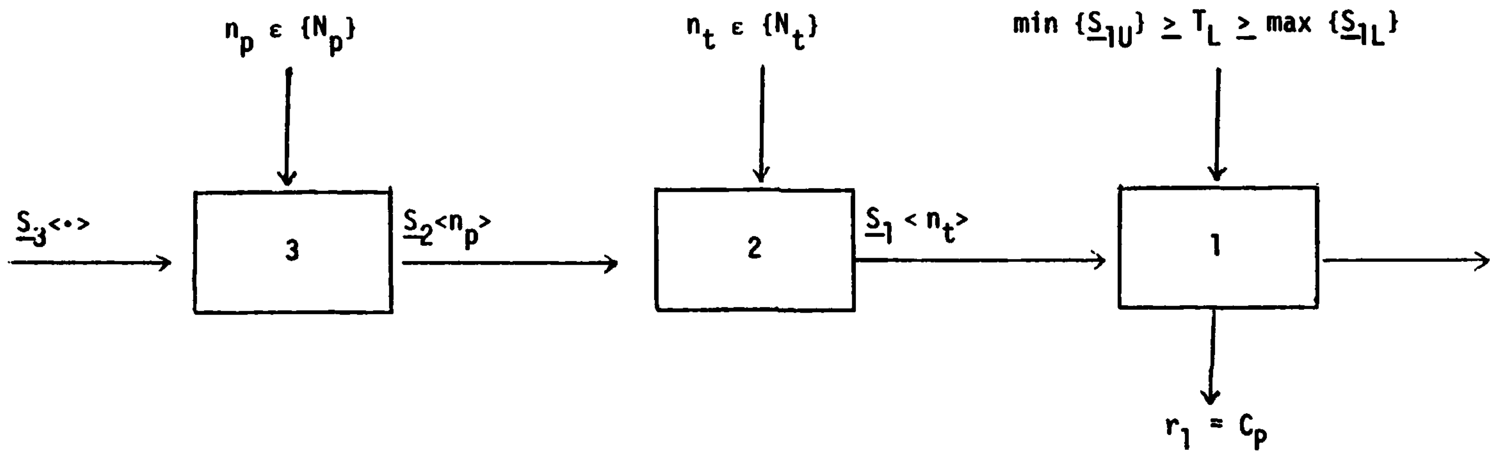


Figure 3.1: Staged representation of reduced problem

limits on tool life given by the altered constraints. $\underline{S}_2\langle n_p \rangle$ represents the same vector altered by the assignment of n_p from its feasible set and likewise for $\underline{S}_1\langle n_t \rangle$ and n_t. Consequently, $\underline{S}_1\langle n_t \rangle$ is a vector of upper and lower bound values for T_L, of which max $\{S_{1L}\}$ represents the maximum lower limit attainable by T_L and min $\{\underline{S}_{1U}\}$ represents the corresponding upper bound limiting value.

A further limiting of the problem is attained by establishing a more restrictive lower and upper bound on n_t through analysis of the tool life equation and the defining relationship for n_t. Since the tool life ratio and, therefore the n_t ratio, is most sensitive to V, the minimum attainable V and n_t can be derived in the following manner:

$$V_{min}' = \left[\frac{C}{f^{\beta}_{max} \cdot T_{L_{max}} \cdot d^{\gamma}_{max}}\right]^{1/\alpha} \tag{3.22}$$

$$n_{t_{min}} = T_p/T_{L_{max}} = \frac{\pi \cdot D \cdot L}{T_{L_{max}} \cdot V_{min}' \cdot f_{max}} = \frac{1}{\left[\frac{T_{L_{Max}} \cdot V_{min}' \cdot f_{max}}{\pi \cdot D \cdot L}\right]} \tag{3.23}$$

in a similar manner $n_{t_{max}}$ can be defined,

$$n_{t_{max}} = \frac{1}{\left[\frac{T_{L_{min}} \cdot V_{max}' \cdot f_{min}}{\pi \cdot D \cdot L}\right]} \tag{3.24}$$

Figure 3.2 is a recapitulation of the mathematical model written in terms of n_p, n_t and T_L. Figure 3.3 shows this model with the above data inserted.

Minimize $C_p = (n_p \cdot n_t)(C_c \cdot T_L + C_T \cdot T_R)$

Subject to:

$n_p \in (N_p)$

$n_t \in (N_t)$

$$\left[\frac{\pi \cdot D \cdot L \cdot d_c^{\gamma/\alpha} \cdot f_{min}^{(\beta/\alpha-1)}}{C^{1/\alpha} \cdot n_t \cdot n_p^{\gamma/\alpha}}\right]^{\left[\frac{1}{1-(1/\alpha)}\right]} \geq T_L \geq \left[\frac{f_{max}^{(\beta/\alpha-1)} \cdot \pi \cdot D \cdot L \cdot d_c^{\gamma/\alpha}}{C^{1/\alpha} \cdot n_t \cdot n_p^{\gamma/\alpha}}\right]^{\left[\frac{1}{1-(1/\alpha)}\right]}$$

$$\left[\frac{C^{1/\alpha} \cdot n_p^{\gamma/\alpha}}{d_c^{\gamma/\alpha} \cdot V_{max}\left[\frac{C^{1/\alpha} \cdot n_t \cdot n_p^{\gamma/\alpha}}{\pi \cdot D \cdot L \cdot d_c^{\gamma/\alpha}}\right]^{\left[\frac{1}{(\beta/\alpha)-1}\right]}}\right]^{\left[\frac{1}{(1/\alpha) + \left[\frac{1-(1/\alpha)}{(\beta/\alpha)-1}\right]}\right]} \geq T_L \geq \left[\frac{C^{1/\alpha} \cdot n_p^{\gamma/\alpha}}{d_c^{\gamma/\alpha} \cdot V_{min}\left[\frac{C^{1/\alpha} \cdot n_t \cdot n_p^{\gamma/\alpha}}{\pi \cdot D \cdot L \cdot d_c^{\gamma/\alpha}}\right]^{\left[\frac{1}{(\beta/\alpha)-1}\right]}}\right]^{\left[\frac{1}{(1/\alpha) + \frac{1-(1/\alpha)}{(\beta/\alpha)-1}}\right]}$$

$$T_{L_{min}} \leq T_L \leq T_{L_{max}}$$

$$T_L \geq \left[\left[\frac{T_{h_{max}} \cdot n_p}{2.02\, d_c}\right]^{\left[\frac{(\beta/\alpha)-1}{.72}\right]} \left[\frac{\pi \cdot D \cdot L \cdot d_c^{\gamma/\alpha}}{C^{1/\alpha} \cdot n_t \cdot n_p^{\gamma/\alpha}}\right]\right]^{\left[\frac{1}{1-(1/\alpha)}\right]}$$

$$T_L \geq \left[\left[\frac{M_{max} \cdot n_p}{0.084\, d_c \cdot D}\right]^{\left[\frac{(\beta/\alpha)-1}{.75}\right]} \left[\frac{\pi \cdot D \cdot L \cdot d_c^{\gamma/\alpha}}{C^{1/\alpha} \cdot n_t \cdot n_p^{\gamma/\alpha}}\right]\right]^{\left[\frac{1}{1-(1/\alpha)}\right]}$$

$$T_L \leq \left[\left[\frac{C^{1/\alpha} \cdot D \cdot d_c^{(1-\gamma/\alpha)}}{71811.548\, P_{max} n_p^{(1-\gamma/\alpha)}}\right]\left[\frac{C^{1/\alpha} \cdot n_t \cdot n_p^{\gamma/\alpha}}{\pi \cdot D \cdot L \cdot d_c^{\gamma/\alpha}}\right]^{(.75-\beta/\alpha)}\right]^{\left[\frac{1}{(1/\alpha) - (1-(1/\alpha))(.75 - (\beta/\alpha))}\right]}$$

Figure 3.2: Resultant model structure.

Minimize $\quad C_p = (n_p \cdot n_t)(0.3\ T_L + 0.75)$

Subject to:

$$n_p \in \{2,3,4\}$$

$$n_t \in \{\frac{1}{10}, \frac{1}{9}, \frac{1}{8}, \ldots, \frac{1}{4}\}$$

$$\left[\frac{3.8574}{n_t \cdot n_p^{.15}}\right]^{1.25} \geq T_L \geq \left[\frac{2.1264}{n_t \cdot n_p^{.15}}\right]^{1.25}$$

$$\left[\frac{0.4454 \cdot n_p^{.15}}{(0.567 \cdot n_t \cdot n_p^{.15})^{-1.54}}\right]^{-0.97} \geq T_L \geq \left[\frac{35.6323 \cdot n_p^{.15}}{(0.567 \cdot n_t \cdot n_p^{.15})^{-1.54}}\right]^{-0.97}$$

$$25 \leq T_L \leq 45$$

$$T_L \geq \left[(5.9406 \cdot n_p)^{-.90} \left(\frac{1.7637}{n_t \cdot n_p^{.15}}\right)\right]^{1.25}$$

$$T_L \geq \left[(1.131 \cdot n_p)^{-.867} \left(\frac{1.7637}{n_t \cdot n_p^{.15}}\right)\right]^{1.25}$$

$$T_L \leq \left[(551.29 \cdot n_p^{-.85})(0.567 \cdot n_t \cdot n_p^{.15})^{.4}\right]^{-8.33}$$

Figure 3.3: Mathematical model for example problem.

An iterative search algorithm was developed to take advantage of these discrete aspects of the problem formulation. Figure 3.4 is an information flow chart of the algorithm and Table 3.2 contains solution information for the example problem. The optimum solution is at iteration 4 and is as follows:

$$d^* = 2.5, \quad f^* = 0.75, \quad V^* = 114.232, \quad T_{L_p} = 25.0000.$$

The execution time required to produce all feasible solutions was 0.27 sec using a FORTRAN algorithm implemented on an IBM 370/Mod 165 computer. However, taking advantage of the sectional convexity of the objective function, termination of the solution procedure upon reaching a minimum reduces the execution time to 0.11 seconds.

Table 3.3 presents a relative comparison of the solution results obtained for the above example problem when solved using several existing algorithms and the staged approach given here. Not only does the approach given here produce an optimum result much faster; but, more importantly, it is the only algorithm which generates a solution that reflects tool changes only between passes (an important practical consideration).

3.4 Conclusions

The realistic assumptions utilized in developing the machine parameter optimization model presented here offer the basis for a much more efficient algorithm than previously available. The method is very amenable to planning optimum machine settings and can be used to initialize an adaptive control system which will then reoptimize feed

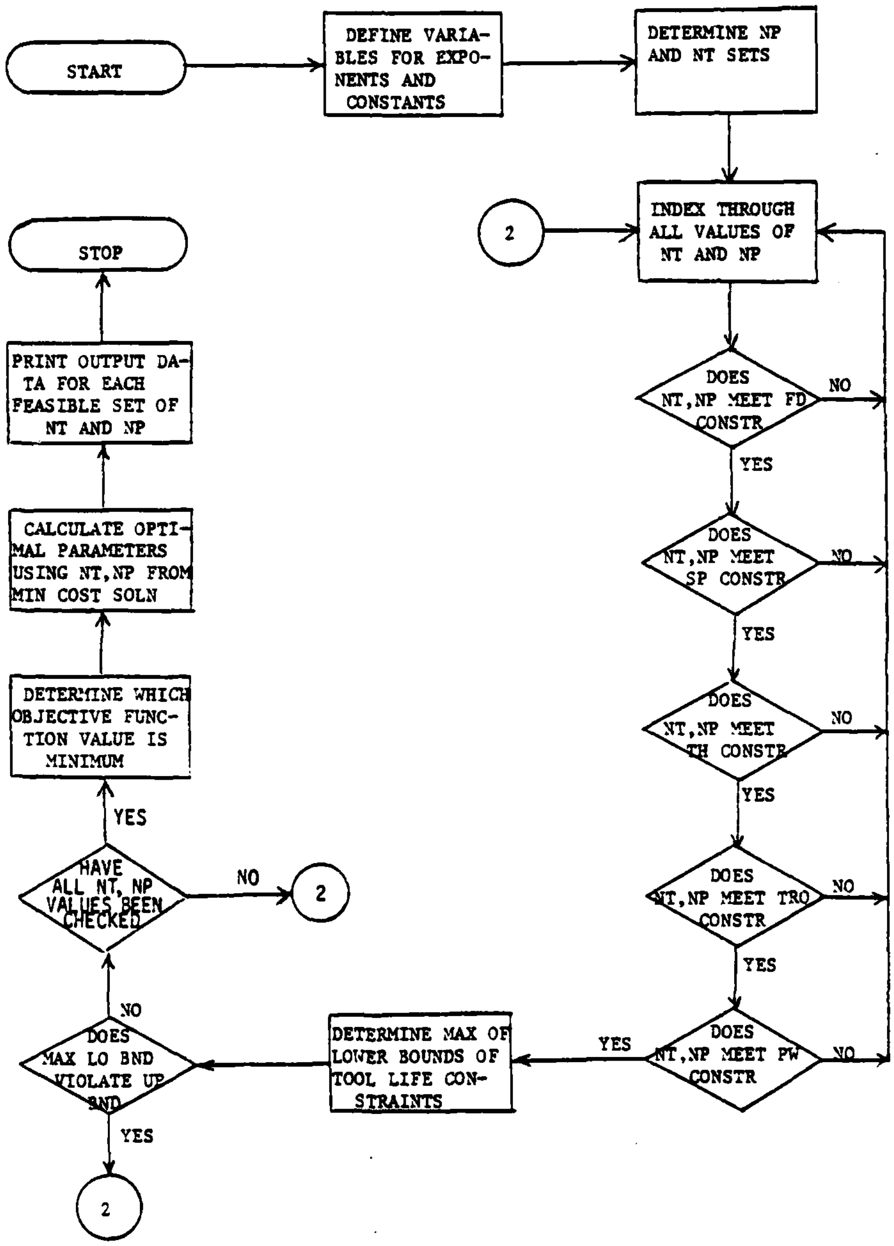


Figure 3.4: Information flow chart of the machine parameters optimization algorithm.

Table 3.2 Feasible Results for Example Problem

Iteration Number	n_p	n_t	Tool Life (min)	Feed (mm/rev)	Speed (m/min)	Depth of Cut (mm)	Cost Per Part ($)
1	2	0.100	40.089	0.75	104.487	2.50	2.555
2	2	0.111	35.142	0.75	107.276	2.50	2.509
3	2	0.125	30.331	0.75	110.482	2.50	2.462
4	2	0.142	25.668	0.75	114.232	2.50	2.414
5	2	0.167	25.000	0.61	123.365	2.50	2.750
6	2	0.200	25.000	0.46	136.091	2.50	3.300
7	2	0.250	25.000	0.33	153.465	2.50	4.125
8	3	0.100	37.154	0.75	112.740	1.67	3.569
9	3	0.111	32.569	0.75	119.208	1.67	3.507
10	3	0.125	28.111	0.75	119.208	1.67	3.444
11	3	0.143	25.000	0.71	124.675	1.67	3.536
12	3	0.167	25.000	0.56	135.466	1.67	4.125
13	3	0.200	25.000	0.42	149.439	1.67	4.950
14	4	0.100	35.203	0.75	118.988	1.25	4.524
15	4	0.111	30.859	0.75	122.164	1.25	4.448
16	4	0.125	26.634	0.75	125.815	1.25	4.370
17	4	0.143	25.000	0.66	133.233	1.25	4.714
18	4	0.167	25.000	0.52	144.764	1.25	5.500
19	4	0.200	25.000	0.39	159.697	1.25	6.600

Table 3.3 Comparison of Solution Times for Example Problem

Algorithm/Reference		Execution Time (sec.)
SUMT	[16]	4.88
Kimbler, Wysk, Davis	[16]	2.85
Davis, Wysk, Agee	[8]	0.46
Hayes, Davis (Thesis Algorithm)		0.11

and speed during cut, as described in [8]. However, the principal advantages of the model are: the discrete nature of the resultant solution process, consequent reduction in solution time, and improved solution accuracy. The discrete parameter information of Table 3.2 can now be incorporated into a generally applicable machine requirements model. The production rate data needed for the MR model can be derived from the machine parameters model by calculating the production time per unit which is imbedded in the formulation. The operating rate for the machine is then the reciprocal of this production time.

Chapter IV

DEFINITION OF THE MACHINE REQUIREMENTS MODEL

4 .1 Introduction

The problem addressed in this section is one of defining the optimum number of machines, their operating rates and the time of operation at each rate to accomplish the production of a specific product. The manufacturing system in which production occurs conforms to a serial, multistage configuration with a different processing operation occurring at each machining center (or stage). It is assumed that product demand and production are given on a per day basis. Within this general framework, a model is developed which can be employed to answer the following fundamental questions related to the design of such a manufacturing system, in the context of cost minimization.

1. How many machines are needed at each machining center of the system?

2. What are the optimum operating rates for the machines in each production stage?

3. How long should the machines be run in each stage for a specific final production level?

Approaches to the first of these questions were addressed in Chapter II but the fundamental framework of the model will closely follow that of Davis and Miller [7]. Such a methodology treats the manufacturing process as an integrated system where production costs vary with processing rates. The rates at each stage exist as either a continuum over a range (for DC type machinery) or as a discrete set of feasible rates (for AC machines). However, the methodology discussed in Chapter III allows a discrete set of operating rates for use in the machine requirements model whether AC or DC machinery is employed. Variation in production cost as the rate varies is due to such factors as tool wear, power consumption, maintenance cost and direct labor, factors which were integral elements of the machine parameters model. It is assumed that the percent defectives resulting from a manufacturing process is a function of the processing parameters as discussed in Section 3.2. Therefore, a defectives rate can be associated with each discrete processing rate.

4.2 Definition of the Mathematical Model

Three basic assumptions underlie the machine requirements model:

1. The machines in each work center are homogeneous.

2. Each production day is broken down into two components, an 8 hour first shift and an 8 hour second shift period.

3. The output from a production stage dictates the amount to be processed at the next stage.

A list of notations used in the development of the machine requirements model is contained in Table 4.1.

For the above system description, the objective for a minimum cost configuration is one which minimizes the following daily operating cost function:

Table 4.1 Notation for the Machine Requirements Model

N	=	the number of machine centers in the system
r_i	=	a feasible operating rate from a set R at stage i
$C^s_{r_i}$	=	manufacturing cost for one hour of production at rate r_i during the first shift ($/hour)
$C^o_{r_i}$	=	manufacturing cost for one hour production at rate r_i during the second shift ($/hour)
b_{r_i}	=	percent of defective units incurred at stage i by processing at rate r_i
C_{F_i}	=	fixed cost of a machine per day in stage i ($/day)
n_i	=	number of machines utilized in stage i
$t^s_{r_i}$	=	number of hours of operation at rate r_i during the first shift
$t^o_{r_i}$	=	number of hours of operation at rate r_i during the second shift time period.

$$\text{Total Cost} = \sum_{i=1}^{N} \sum_{r \varepsilon R_i} (C^s_{r_i} t^s_{r_i} + C^o_{r_i} t^o_{r_i} + C_{F_i} n_i) \tag{4.1}$$

processing cost equipment cost

subject to the following restrictions, for i = 1, . . . , N.

First, the total number of units processed at stage i must equal the quantity available for processing at that stage,

$$\sum_{r \varepsilon R_i} r_i (t^s_{r_i} + t^o_{r_i}) = S_i \tag{4.2}$$

Second, the quantity of output product at a stage, quantity processed less the fraction defective, must equal the quantity processed by the next stage,

$$\sum_{r \varepsilon R_i} r_i (1 - b_{r_i})(t^s_{r_i} + t^o_{r_i}) = \tilde{S}_i = S_{i+1} \tag{4.3}$$

For the final stage this restriction takes the following form,

$$\sum_{r \varepsilon R_i} r_n (1 - b_{r_n})(t^s_{r_n} + t^o_{r_n}) = \tilde{S}_n = \text{final demand} \tag{4.4}$$

Finally, the units being processed at a stage cannot employ more processing time than is available on the number of machines allocated to that stage. For the first shift operation this is:

$$\sum_{r \varepsilon R_i} t^s_{r_i} \leq 8n_i \tag{4.5}$$

and for the second shift operation,

$$\sum_{r \varepsilon R_i} t^o_{r_i} \leq 8n_i, \text{ and} \tag{4.6}$$

$$t^s_{r_i}, t^o_{r_i}, n_i \geq 0 \text{ and } n_i \text{ an integer} \tag{4.7}$$

The above model is a fundamental linear program requiring an integer domain for the subset of variables associated with the number of machines and allowing the other decision variables to have a continuous solution. A multistage system employing this model can be solved using a mixed integer linear programming algorithm with a branch and bound solution technique. However, the solution time and core requirements for such a procedure can become prohibitive for a problem involving a significant number of stages and operating rates. An alternative approach, using dynamic programming, may offer distinct computational advantages and result in a greater variety of decision information for the analyst. Such an approach will be formulated in the next chapter.

Chapter V

DYNAMIC PROGRAMMING SOLUTION PROCEDURE

5.1 Rationale of a Dynamic Programming Formulation

The staged character of the manufacturing system being modelled and the nature of the transition between stages as described in equations (4.2) and (4.3) lead directly to investigating a solution procedure using dynamic programming. Figure 5.1 depicts a multistage serial production system with decisions at each stage represented by the number of machines and operating hours decision variables. The return Z_i, represents the cost incurred at each stage in producing to a specified final demand $\tilde{S}_N$. The transition between stages is represented by the state variable S_i. Solving for the necessary S value at each stage dictates the number required as output from the previous stage. Through working backwards through the problem, a set of decisions can be determined at each stage to attain the required final production level of S.

Handling the problem in the above fashion restricts the decisions to those which are tied directly to the final demand value. However, a much more broadly based solution procedure can be achieved by using state inversion techniques [24] thereby reversing the state transition arrows in the serial system as shown in Figure 5.2. The problem can then be treated as a fundamental dynamic programming problem using a recursive procedure to work against the flow of the arrows. The decision resulting from such an inversion procedure will reflect a broad

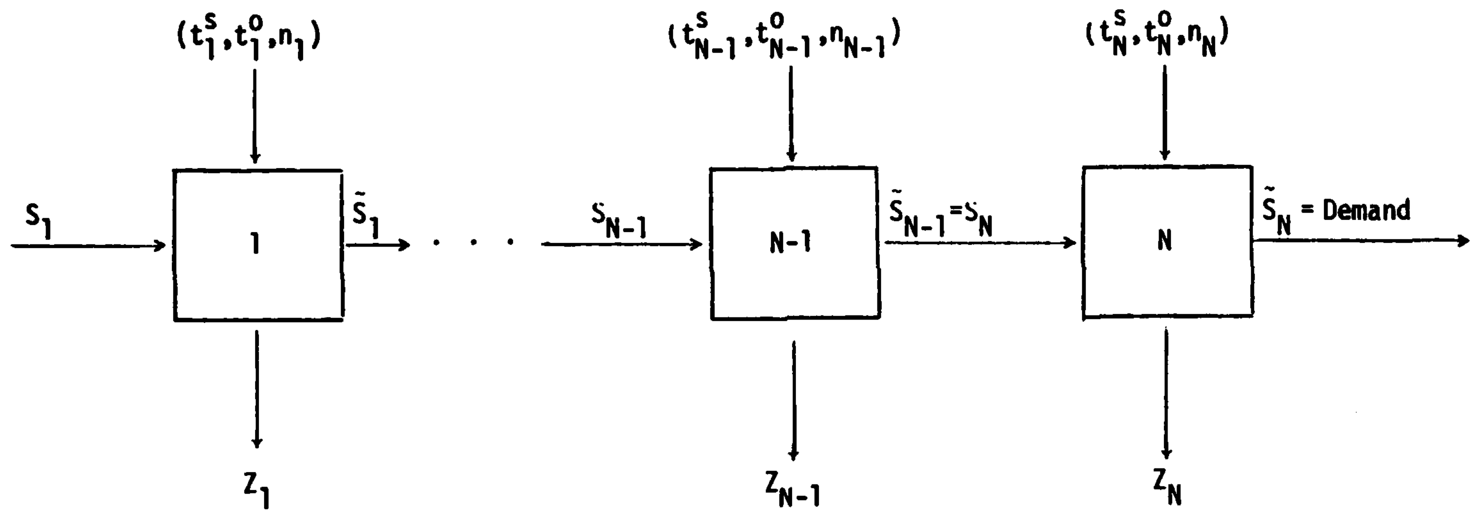


Figure 5.1: Multistage serial production system.

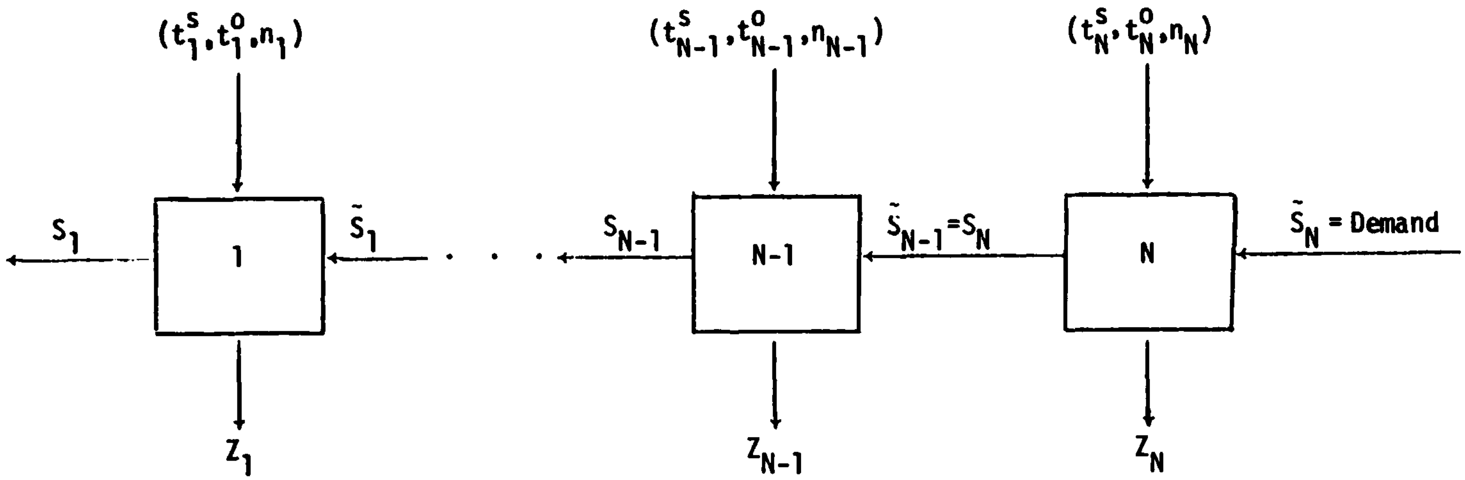


Figure 5.2: Multistage serial production system depicting state inversion.

range of system states in relation to both initial and final values in the production process.

As the model is basically a mixed integer linear program which is decomposable into stages, it is necessary to use linear programming solution procedures at each stage of the dynamic program to determine the optimum decisions which hold for that stage and, therefore, all prior stages. Insuring that both optimality and feasibility are attained in relation to the decisions at each stage will assure that decisions for the entire system are optimized in accordance with the principle of dynamic programming.

5.2 Maintaining Optimality and Feasibility Through Parametric Analysis

The linear programming approach used in each stage of the manufacturing model is actually one in which the initial decision values are predefined to maintain optimality and further iterations maintain optimality of the solution while determining the range of production values for which the current solution remains feasible. Given a set of production rates, unit cost information and percent of defectives for the initial stage in the system, it is possible to determine the lowest cost production rate for the first 8 hour shift. This relationship is of the form:

$$\text{Minimum cost rate} = \min_{r_1 \varepsilon R_1} \left(C_{r_1}/r_1(1-b_{r_1})\right) \tag{5.1}$$

Figure 5.3a illustrates the formation of a linear programming tableau with the three rows of the constraint matrix representing the

	$t^s_{1_1}$	$t^s_{2_1}$	---	---	$t^s_{R_1}$	$t^o_{1_1}$	---	---	$t^o_{R_1}$	A	R_1	R_2	RHS
	$-C^s_{1_1}$	$-C^s_{2_1}$	---	---	$-C^s_{R_1}$	$-C^o_{1_1}$	---	---	$-C^o_{R_1}$	0	0	0	0
A	$r_{1_1}(1-b_{1_1})$	$r_{2_1}(1-b_{2_1})$	---	---	$r_{R_1}(1-b_{R_1})$	$r_{1_1}(1-b_{1_1})$	---	---	$r_{R_1}(1-b_{R_1})$	1	0	0	$8xr_1(1-b_{r_1})_{mc}$
R_1	1	1	1	1	1	0	0	0	0	0	1	0	8
R_2	0	0	0	0	0	1	1	1	1	0	0	1	8

mc = Minimum cost

Figure 5.3a: Initial LP tableau for first stage.

	$t^s_{1_1}$	$t^s_{2_1}$	--- ---	$t^s_{R_1}$	$t^o_{1_1}$	--- ---	$t^o_{R_1}$	A	R_1	R_2	RHS
	$Z^s_{1_1}-C^s_{1_1}$	$Z^s_{2_1}-C^s_{2_1}$	--- ---	$Z^s_{R_1}-C^s_{R_1}$	$Z^o_{1_1}-C^o_{1_1}$	--- ---	$Z_{R_1}-C_{R_1}$	Z_{A_1}	$Z_{R_{1_1}}$	$Z_{R_{2_1}}$	$[C_B][B^{-1}][b]$
$(t^s_{r_1})_{mc}$, R_1, R_2	$[B^{-1}][A]$							B^{-1}			$[B^{-1}][b]$

Figure 5.3b: Initial LP tableau of first stage after first pivot.

restrictions of equations (4.3), (4.5) and (4.6) of the mathematical model. The right hand side of the first row of the matrix is the production level for 8 hours using one machine at the lowest cost production rate. The second and third row right hand side values are the first and second shift slack time hours.

The tableau of Figure 5.3b illustrates the solution after an initial pivoting operation where the variable associated with the minimum cost rate enters the basis replacing the artificial variable A. The production range for which this solution is feasible is zero up to the level determined by 8 hours at $r_{1\,min}$. The key element of linear programming which has repeated application with this model is that of post-optimality analysis which can be done to the right hand side in relation to changes in production level represented by the first row. The change in production level for which a solution is feasible is determined by the effect of the first column of the basis inverse on the right hand side. The feasible range of production is then calculated by moving this basis inverse column to the right hand side and solving for the number of units which maintain feasibility. Each subsequent iteration through the simplex procedure, in which a new variable enters the basis, will maintain optimality while the artificial variable column, or first column of the basis inverse, establishes the production range for which the solution holds. For example, with the first column of the basis inverse and the right hand side vector having the following values,

$$\begin{bmatrix} .0 \\ .034 \\ -.034 \end{bmatrix} \text{ and } \begin{bmatrix} 8.0 \\ .0 \\ 8.0 \end{bmatrix} \text{ leads to } \begin{bmatrix} 8.0 \\ .034 \ \ b \\ 8-.034 \ \ b \end{bmatrix}$$

$$\text{A} \qquad\qquad \text{RHS} \qquad\qquad \text{RHS} + \Delta b$$

the lower range for this solution is the 8 hour production rate determined by the time variable associated with the first row and the upper bound is determined by the point at which the solution becomes infeasible. Solving the equation,

$$8-0.034\ \Delta b = 0 \tag{5.2}$$

determines the Δb value or the additional production above the initial right hand side first row value for which the current solution holds. Continuing to iterate while bringing in the next most cost effective production rate variables will maintain optimality; while determining the effective range of production through postoptimality analysis of the right hand side. These iterations are accomplished through dual simplexing on the right hand side value which will first cause the solution to become infeasible when parametric analysis is used. Progressing in this manner will insure that there are no discontinuities in the ranges.

The iterations at each stage continue until either no variable can be brought into the basis or the upper limit of production is reached. If no additional variable can enter the basis, it is necessary to add a machine to accomplish higher production levels. This is done through increasing the initial hours available for first and second shift production by 8 for each shift and continuing to iterate after insuring that the solution is returned to feasibility. When the upper limit of planned production is attained at a stage, it is necessary to continue to the next stage.

5.3 Transition Between Stages of the Manufacturing Process

Following the initial stage of production, each stage of the production process, and therefore the model, must absorb the returns (or cumulative costs) of stages previous to it. It must also reflect the state variable production quantity transition value between stages. The addition of a new row to the constraint matrix representing equation (4.2) of the model enables both of these requirements to be met. Figure 5.4a is the initial simplex tableau for all stages other than the first. S_i is the transition value and represents the input to stage i to obtain the level of production indicated for the current solution. It is also the output needed from stage i-1. The cost of production from previous stages must be absorbed into the current stage. The cost of production at a stage is represented by the objective function value associated with a feasible tableau and contains two components. One is the fixed component consisting of the basic variable cost values multiplied by the right hand side. The fixed component also includes the fixed cost of a machine times the number of machines. The variable component is the basic variable cost vector times the first column of the basis inverse to account for the cost of production above the lower bound of the range for which the solution holds. In summary, this relationship can be stated as,

	$t^s_{1_1}$	$t^s_{2_1}$	---	---	$t^s_{R_1}$	$t^o_{1_1}$	---	---	$t^o_{R_1}$	A	R_1	R_2	S_1	RHS
	$-C^s_{1_1}$	$-C^s_{2_1}$	---	---	$-C^s_{R_1}$	$-C^o_{1_1}$	---	---	$-C^o_{R_1}$	0	0	0	0	0
A	$r_{1_1}(1-b_{1_1})$	$r_{2_1}(1-b_{2_1})$	---	---	$r_{R_1}(1-b_{R_1})$	$r_{1_1}(1-b_{1_1})$	---	---	$r_{R_1}(1-b_{R_1})$	1	0	0	0	$Fxr_1(1-b_{r_1})_{mc}$
R_1	1	1	1	1	1	0	0	0	0	0	1	0	0	8
R_2	0	0	0	0	0	1	1	1	1	0	0	1	0	8
S_1	$-r_{1_1}$	$-r_{2_1}$	---	---	$-r_{R_1}$	$-r_{1_1}$	---	---	$-r_{R_1}$	0	0	0	1	0

mc = Minimum cost

Figure 5.4a: Initial LP tableau for stages other than first.

	$t^s_{1_1}$	$t^s_{2_1}$	---	---	$t^s_{R_1}$	$t^o_{1_1}$	---	---	$t^o_{R_1}$	A	R_1	R_2	S_1	RHS
	$Z^s_{1_1}-C^s_{1_1}$	$Z^s_{2_1}-C^s_{2_1}$	---	---	$Z^s_{R_1}-C^s_{R_1}$	$Z^o_{1_1}-C^o_{1_1}$	---	---	$Z^o_{R_1}-C^o_{R_1}$	Z_{A_1}	$Z_{R_{1_1}}$	$Z_{R_{2_1}}$	0	$[C_B][B^{-1}][b]$
$(t^s_{r_1})_{mc}$ R_1 R_2 S_1	$[B^{-1}][A]$									B^{-1}				$[B^{-1}][b]$

Figure 5.4b: Initial LP tableau for stages other than first after first pivot operation.

$$\text{Total Cost}_i = C_{F_i} \cdot n_i + [C_B][b] + [C_B][B_i^{-1}] \qquad (5.3)$$

When a stage absorbs the cost from a previous stage, it must absorb both the fixed and variable components. The fixed portion is added directly to the objective function value of the next stage while the variable component becomes the cost coefficient associated with the transition function value S_i. Since S_i is always in the basis, the cost coefficient, C_{S_i}, is always included as a component of C_B, the basis cost vector. Figure 5.4b is the initial tableau of the ith stage (for i greater than 1) following the first pivot operation, or iteration update. Thus the cost (or objective function value) includes the cost associated with the previous stage and all stages prior to it. The S_i variable in the basis gives the input level needed at the ith stage and the necessary output from stage i-1.

Since the costs and transition values are optimized at each stage, and therefore independent of all previous stages after absorption, the number of stages has no effect on the procedure except in total solution time.

The most critical aspect of absorbing previous stage costs is that of maintaining continuity of production ranges. Cost functions which hold for particular ranges in stage i must be absorbed into stage i + 1 with full cognizance of the ranges involved. A change in cost function in stage i directly causes a change in the fixed and variable portions of the cost components absorbed into stage i + 1. A range split will then result in stage i + 1. The solution vector for stage i + 1 will not change, but the cost associated with the solution changes, necessitating the split in ranges to maintain correct accountability of absorbed costs. An element which must be considered in handling the absorption of returns caused by a range split is that the range from a previous stage affects the current stage through the transition function; and is not directly analogous to the current production of good units but to the input units to the current stage. For instance, if a change in returns occurred at a certain production level in the previous stage, that change is reflected in the current stage at a production level which is reduced to account for defectives in the production process.

5.4 Incorporating Economic Tradeoffs as Decision Criteria

A key decision point in the analysis of the manufacturing process is whether it is more economically advantageous to bring in higher cost production rates or to increase the number of machines in order to increase the level of production in a stage of the machining system. Information included in the linear programming solution provides a readily available means of economic analysis to make this decision. Associated with the slack values in the constraint matrix of the LP procedure are reduced cost, or "shadow price," values represented as the Zj-Cj values in the cost row pertaining to the slack variables. These values indicate the amount of savings which could be incurred for each additional hour of time available in the first or second shift respectively. It must also be considered that adding hours of production can only be done in 16 hour increments per machine. Therefore, what appears as a simple check to determine if the savings available exceeds the cost of adding another machine, becomes more complex with the realization that a full

16 hours of production may not be needed from another machine. Consequently, the decision must be based on production levels as well as costs and savings, a task for which the LP procedure is particularly well suited.

Through the use of the objective function cost values, it is possible to determine whether it is economically feasible to add a machine as well as the precise production level at which this occurs. Within a particular stage, the reduced costs associated with the slack variables are used as an indicator of whether the economic situation bears further analysis. If,

$$8 \cdot (Z_{R_1} - C_{R_1}) + (Z_{R_2} - C_{R_2}) > C_{F_i} , \qquad (5.4)$$

it is necessary to undertake further analysis. The reduced costs are multiplied by 8 hours since adding a machine would provide two additional shifts at 8 hours each and the relationship of equation (5.4) is therefore balanced.

Upon determination that it may be advantageous to add another machine, a formal analysis is conducted to determine if the pertinent production level for a breakeven point is within the range associated with the current solution vector. This analysis proceeds as follows. Let RVCUR be the current variable portion of the cost function and RFCUR be the fixed portion as they pertain to a basic LP solution with reduced costs satisfying equation (5.4). In order to test whether adding a machine is valid, it is now necessary to hypothetically do so by adding 8 hours to the first and second shifts.

The lower bound of the range for which the basic solution holds is used as the new initial right hand side production value. Using the dual simplex procedure, the feasibility of the solution can be restored and the lower bounds of the base solution and the updated solution will correspond, although the upper bounds of the ranges may differ. The objective functions of the two solutions will differ since they involve different bases and this serves as the means of comparison for purposes of a decision.

Through setting the current objective function of the base solution equal to the objective function of the feasible solution with a machine added, represented by fixed and variable components, RFP and RVP, as

$$RFCUR + RVCUR\ (\Delta b) = RFP + RVP\ (\Delta b) + C_{F_i}\,, \qquad (5.5)$$

a Δb value can be determined for which the cost functions are equal. This indicates the production level at which adding a machine is economically feasible. If this production level is outside the range of production for which the base and revised solution hold, then no machine is added and further iterations proceed from the base solution. If adding a machine is indicated, iterations proceed from the feasible revised solution with a machine added. The production level at which the change occurs becomes the upper bound of the production range for the base solution and the lower bound for the feasible range of the revised solution. Adding any number of machines proceeds in the same fashion since adding 8 hours to each shift has the same effect on both sides of equations (5.4) and (5.5), regardless of the current level of machines.

All aspects of the decision criteria necessary to utilize the dynamic programming formulation have now been discussed. It remains to develop a computer algorithm which incorporates all elements of the dynamic and linear programming solution procedures. This development will be detailed in the next chapter.

Chapter VI

DEVELOPMENT OF THE COMPUTER ALGORITHM FOR THE MR MODEL

6.1 Introduction

A computer algorithm to achieve a solution to the dynamic programming formulation of the machine requirements model must include all of the decision analysis discussed in the previous chapter. The most effective means of designing such an algorithm is to utilize subroutines to perform those aspects of the procedure which recur repetitively, particularly the linear programming iterations. This chapter will utilize a discussion and information flow charts to explain the operation of the solution procedure, focusing on the main program and then those subroutines which require more detailed explanation. Further detail is available in Appendix B where the complete code listing is included along with a description of notation used in the FORTRAN program.

6.2 Main Program

The main program of the computer code is the vehicle which coordinates all elements of the solution procedure. Those decision points which are reached within a subprogram will be discussed in relation to that subroutine. Figure 6.1 is an information flow chart of the main program. Given a set of input data in the form of operating rates, percent defectives, production costs and fixed equipment costs, the first step in the solution algorithm is to set up the initial simplex tableau for each stage of the production process. This initial tableau will be

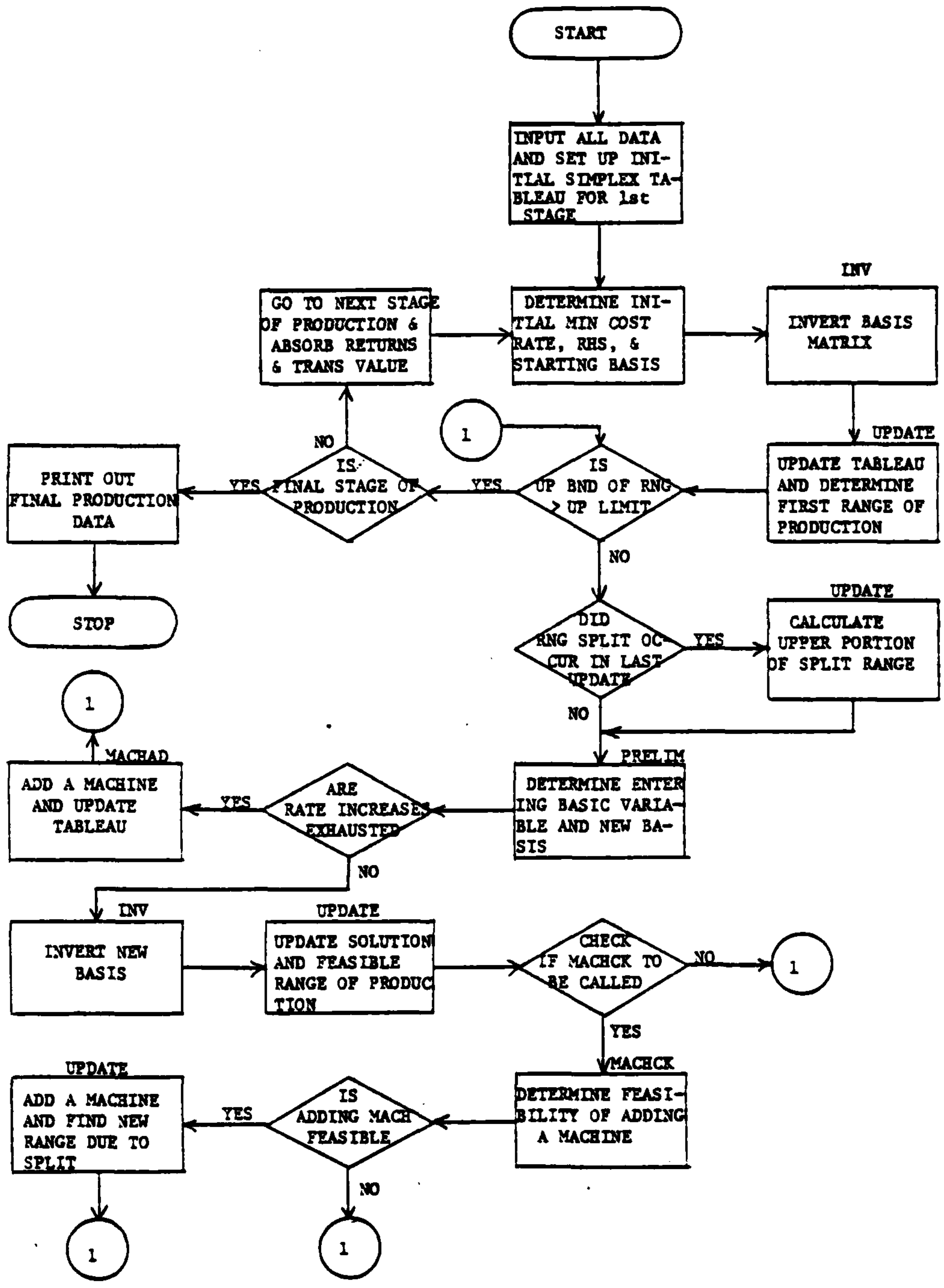


Figure 6.1: Information flow chart of main program.

utilized within each stage as it is reached in the dynamic programming formulation. Following this initial step, the most cost effective entering variable is determined and the simplex iterations within each stage begin. Each iteration corresponds to a distinct production range at a stage.

Unless there is a range split due to a change in cost function from the previous stage or a machine being added, each change in production range is marked by a change in basis requiring the formation of a new tableau. The new tableau representing revised solution information is derived in the algorithm through a series of checks and matrix manipulations. Upon determination of the need to change the basis to attain a higher production level, the PRELIM subroutine is called. This routine determines the row in which to dual simplex, and therefore the leaving variable, and finds the entering variable using the minimum ratio rule. Upon establishing the new basis, the INV subroutine is called to invert the basis matrix for use in updating the tableau. If no variable is eligible to enter the basis, indicating all of the feasible rate increases have been exhausted, the subroutine to add a machine, MACHAD, is called and the resultant tableau is then utilized for further iterations.

The revised basis inverse can now be used in conjunction with the initial tableau for the current stage to obtain the updated tableau. This procedure is accomplished in the UPDATE subroutine where the initial tableau constraint matrix is premultiplied by the basis inverse to obtain the current constraint matrix values. The right hand side, objective function and Zj-Cj values are also updated using the basis inverse or basic variable cost values where appropriate. The updated tableau now contains all of the information needed to determine the production range, transition function values and applicable cost function. These calculations are also performed within the UPDATE routine. If the updating of a tableau is necessary within a dual simplexing procedure designed to return a solution to feasibility, a RETURN statement is used with the UPDATE routine to bypass the calculations not associated directly with the update of a tableau. Such a dual simplex procedure occurs within the subroutines which check for adding a machine (MACHCK) and which add a machine when rates are exhausted (MACHAD).

Each time a new solution is determined, it is necessary to check if adding a machine may be economically feasible. If this check is positive, the MACHCK subroutine is called to make this analysis as explained in Chapter V. Upon this determination, the necessary updating is accomplished and the normal procedure continues with return to 1 as indicated in the flow chart.

When the upper limit of production is reached within a stage, the solution information is compiled and printed in a form which can be used to develop the operational data which optimizes the entire production system. Given any final demand value for production within the upper limit of the last stage, the total cost of production, operating rates at each stage, number of machines and hours of operation at each rate can be determined. Two of the subroutines will now be explained in greater detail.

6.3 Machine Check Subroutine

The flow of decisions within the subroutine MACHCK for determining the economic feasibility of adding a machine is depicted in Figure 6.2 Since the procedure of the analysis is to hypothetically add a machine and use the resultant solution for further analysis, it is first necessary to retain all pertinent values of the iteration which leads to the check. This information can then be recovered if no machine is added. When a machine is added to perform the analysis, the immediate result is that the solution becomes infeasible for the range comparison needed. Therefore, the dual simplex procedure must be used to drive the solution to feasibility at which time the ranges will be compatible for analysis. This dual simplex procedure is handled by the RHSCK subroutine which is used solely as a vehicle to determine the new basis while iterating towards feasibility. Once this is accomplished, the decision criteria discussed in Section 5.4 is utilized to decide upon the addition of a machine and the resultant range information. Deciding to not add a machine results in recovering the solution tableau which initiated the procedure and iterations continue from there.

6.4 Machine Add Subroutine

The MACHAD subroutine is invoked by the PRELIM subroutine when it is no longer feasible to bring another rate into the basis. Adding a machine makes it possible to achieve higher production levels. This subroutine alters the right hand side of the initial tableau for the current stage to reflect the increase in production hours due to an additional machine. The UPDATE, RHSCK and INV subroutines are then used to drive

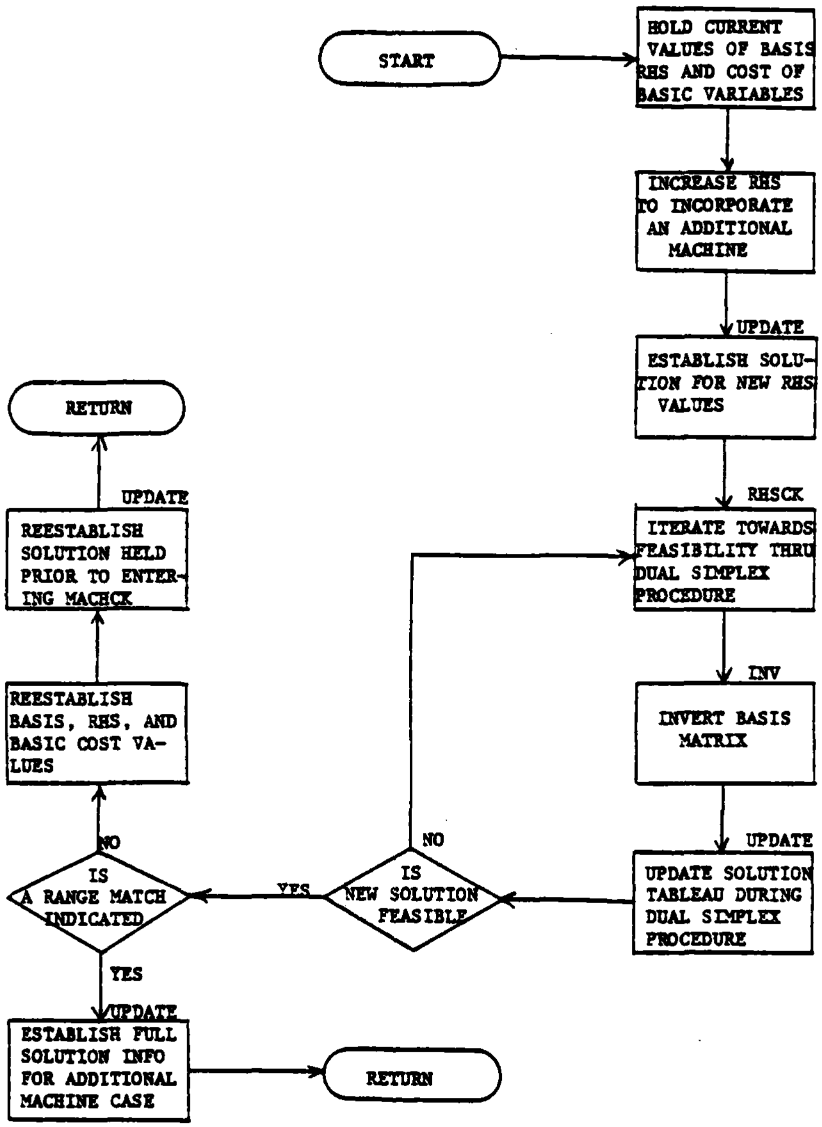


Figure 6.2: Information flow chart of the MACHCK subroutine.

the solution to feasibility at a range lower bound corresponding to the upper bound of the range of the last feasible solution without the additional machine. Continuity of production ranges is maintained and iterations can continue from this point.

Chapter VII

COMPUTATIONAL RESULTS AND MODEL VALIDATION

7.1 Validation of the Dynamic Programming Formulation

In order to be assured that the dynamic programming formulation is performing all of the functions for which it was designed, it is necessary to test the model against a valid reference. Although hand calculations can be used to verify calculations in the model, the complexity and detail of the solution procedure makes such a validation prohibitive for all but the most simple example. Hand calculated simplex tableaus were used to test some elements of the model but the most meaningful validation results from using a commercial integer linear programming system as the means of comparison of results. For this purpose, the Mathematical Programming System (MPS) mixed integer linear programming package (MISTIC) was utilized. This system is available through the Virginia Tech Computing Center.

A four stage manufacturing process with input data as shown in Table 7.1 was used as the validation example. The procedure of using the results of the dynamic programming solution algorithm is shown in Table 7.2. This table depicts the production range of each stage and the resulting operational information for optimization of producing to a final demand of 500 units. The procedure starts with the fourth stage since the final production is known and the decisions can be determined through recursive movement through the stages. The complete

Table 7.1 Input Data for 4 Stage Production Process

Stage		Production Data					
1	Rates (Units/hour)	25	30	35	40	45	50
	% Defectives	.041	.042	.045	.050	.051	.055
	1st Shift Cost ($/hour)	10.0	12.0	14.0	17.2	19.8	25.0
	2nd Shift Cost ($/hour)	13.0	13.8	16.1	19.2	22.9	30.0
2	Rates	10	15	20	25	30	
	% Defectives	.094	.095	.101	.105	.110	
	1st Shift Cost ($/hour)	7.0	7.95	9.00	10.75	13.2	
	2nd Shift Cost ($/hour)	10.00	10.95	13.00	14.00	17.40	
3	Rates	40	45	50	55	60	
	% Defectives	.045	.047	.050	.051	.053	
	1st Shift Cost ($/hour)	5.2	5.85	8.0	9.35	10.8	
	2nd Shift Cost ($/hour)	7.2	7.65	10.0	11.0	13.8	
4	Rates	5	10	15	20	25	
	% Defectives	.057	.060	.063	.067	.072	
	1st Shift Cost ($/hour)	10.0	15.0	19.95	30.0	40.0	
	2nd Shift Cost ($/hour)	15.0	20.0	30.0	35.0	50.0	

Fixed Cost of Equipment Per Day:

Stage	Cost ($/day)
1	80.0
2	60.0
3	40.0
4	70.0

Table 7.2 Analysis of Solution Results for 4 Stage Example

Stage 4:	Enter at 500 units output in range 483 - 512	
	Total Cost = (3.6220)(500) + 81.56 = $1892.56	
	Operating Rates and Production Time: 2 machines	
	1st Shift: 16 hours at 15 units/hour	= 240 units
	2nd Shift ((0.0536)(500) - 12.05) hours at 20	= 295 units
	Total Units Processed	= 535 units
Stage 3:	Enter at 535 units output in range 517 - 548	
	Operating Rates and Production Time: 1 machine	
	1st Shift: 8 hours at 45 units/hour	= 360 units
	2nd Shift: ((0.0233)(535) - 8) hours at 45	= 201 units
	Total Units Processed	= 561 units
Stage 2:	Enter at 561 units output in range 542 -575	
	Operating Rates and Production Time: 2 machines	
	1st Shift: 16 hours at 30 units/hour	= 480 units
	2nd Shift: ((0.0447)(561) - 19.09) hours at 25	= 150 units
	Total Units Processed	= 630 units
Stage 1:	Enter at 630 units output in range 609 - 645	
	Operating Rates and Production Time: 1 machine	
	1st Shift: 8 hours at 45 units/hour	= 360 units
	2nd Shift: ((0.2186)(630) - 133.12) hours at 40	= 184 units
	((-.2186)(630) + 141.12) hours at 35	= 119 units
	Total Units Processed	= 663 units

solution results for the four stage example are shown in Appendix C and these can be used to find the optimal operational data for any final demand up to 753 units.

To test the results obtained from the dynamic programming model and to assure that all routines of the model are exercised, two validation computer runs were made using the MISTIC procedure. One test was for a final demand of 500 units and the other for a final demand of 650 units. Table 7.3 depicts the results of the comparison of the two models for each of the demand values. These results verify that both procedures reach virtually identical results for a specific final demand when maintaining an integer number of machines. Of particular note is the difference in execution time to attain the same solution using the different procedures. Not only is the dynamic programming formulation much faster, using an IBM 370/158 computer, but it only has to be executed once to obtain solution information for any projected final demand up to the upper limit planned for. The integer linear programming procedure must be executed for each distinct final demand value.

7.2 Computational Efficiency of the Model

Several tests of the dynamic programming model formulation were conducted to determine the computational effect as related to the MISTIC procedure. In addition to the four stage process already examined, the number of operating rates for the four stage example were doubled to test the effect of increased number of variables on the decision process. To analyze the effect of an increased number of stages in the

Table 7.3 Comparison of Validation Results

A. Final Demand of 500 Units

		DP Formulation				ILP MISTIC Formulation			
Stage	Shift	Rate	Hours	Machines	Units	Rate	Hours	Machines	Units
4	1	15	16.0	2	240	15	16.0	2	240
	2	20	14.75	2	295	20	14.75	2	295
3	1	45	8.0	1	360	45	8.0	1	360
	2	45	4.47	1	201	45	4.47	1	201
2	1	30	16.0	2	480	30	16.0	2	480
	2	25	6.0	1	150	25	6.0	1	150
1	1	45	8.0	1	360	45	8.0	1	360
	2	40	4.6	1	184	40	4.5	1	181
		35	3.4	1	119	35	3.5	1	122
Total Production Cost:		$1892.56				$1894.01			
Execution Time:		13.1 seconds				1 min. 36 sec.			

Table 7.3--(Continued)

B. Final Demand of 650 Units

		DP Formulation				ILP MISTIC Formulation			
Stage	Shift	Rate	Hours	Machines	Units	Rate	Hours	Machines	Units
4	1	15	24.0	3	360	15	24.0	3	360
	2	20	16.76	3	335	20	16.76	3	335
3	1	45	7.34	1	330	45	7.36	1	331
	1	60	.66	1	40	60	.64	1	38
	2	45	8.0	1	360	45	8.0	1	360
2	1	30	16.0	2	480	30	16.0	1	480
	2	25	13.5	2	338	25	13.5	1	338
1	1	35	16.0	2	560	35	16.0	2	560
	2	30	9.86	2	296	30	9.86	2	296
Total Production Cost:		$2466.91				$2468.30			
Execution Time:		13.1 seconds				1 min. 36 sec.			

manufacturing process, a six stage and a five stage example were formulated and tested. All of the examples were solved using both the DP procedure and MISTIC. The results of these test problems are shown in Table 7.4 and the production costs are identified for each method to further support the credibility of the DP model.

The most significant aspect of the comparative results is the major decrease in computation time for the DP algorithm as compared to the integer programming method. As the number of stages increases, the time to attain a solution with the MISTIC procedure rapidly increases while the DP model leads to a more linear increase. The modelling of even higher numbers of stages should realize an even greater discrepancy in execution times. Such a result indicates that the MISTIC code or, in fact, any branch and bound mixed integer procedure, would be prohibitive for a complex manufacturing process.

Two other areas for consideration in comparing the two solution procedures are input data requirements and the computer core storage needed. The data requirements for a programming package such as MPS are considerably more complex than that of the DP model as demonstrated in Table 7.1. MPS is limited to two data items per card and the task of preparing the data cards for a large linear programming constraint matrix is monumental. The DP formulation contains routines to form the linear programming tableaus from simple production data and much preparation time and sources for error are eliminated. Table 7.4 also indicates the core requirements for the computer runs of the example problems. Much less core is required in the DP algorithm resulting in

Table 7.4 Comparative Results for MR Test Problems

Type Test	DP Formulation			ILP Formulation			Computation Cost Ratio (ILP/DP)
	Execution Time (sec)	Core (KB-sec)	Prod Cost ($)	Execution Time (sec)	Core (KB-sec)	Prod Cost ($)	
4 Stage:							
Demand=500	13.11	5280 (195K)*	1892.56	96	50232 (42K)	1894.01	3.39
Demand=650	13.11	5280 (195K)*	2466.91	86	41832 (35K)	2468.30	2.90
4 Stage--Double # of Rates							
Demand=650	16.44	6336 (195K)*	2430.60	119	51232 (90.5K)	2431.83	2.24
5 Stage:							
Demand=400	14.01	5632 (195K)*	2009.84	547	1874432 (303K)	2018.63	22.40
6 Stage:							
Demand=400	16.42	6336 (195K)*	2838.80	3554	19119432 (537K)	2838.86	203.55

*Denotes constant dimensioning for all DP solutions.

application to computer systems with smaller memory capacity and less cost of computer operation.

The significant results obtained through solving the machine requirements problem with dynamic programming offers a base upon which to build further research in manufacturing systems optimization. Extension of the model into additional areas will be discussed in the next chapter.

Chapter VIII

EXTENSIONS

8.1 Extensions Related to Dynamic Programming Formulation

The dynamic programming formulation of the machine requirements model lends itself to several variations on the model depicted in Chapter IV. Any manufacturing process which can be handled in distinct stages and which has similar production and cost data can be modelled. This feature allows extension of the model to non-serial systems such as converging and diverging branch and feedforward and feedback systems. In essence, non-serial systems can be decomposed into a series of serial systems for solution purposes. The extension of the model to these types of systems would add complexity to the absorption of previous stage production cost as well as to the transition function. The basic solution procedure is capable of handling such complexities within the linear programming structure at each stage.

The feasibility of extending the DP formulation to non-serial systems allows the procedure to be applied to complex manufacturing systems having a variety of operating modes. Such production situations as manufacturing two or more products simultaneously for all or a portion of a production line, rework of defective items, and the converging or diverging of machining processes in a system can all be treated within the framework of non-serial systems. More detail on the structure and solution of such systems is available in Nemhauser [24].

An example of a non-serial system which is particularly amenable to a dynamic programming solution procedure is the converging branch system as diagrammed in Figure 8.1. This system consists of three serial flow machining systems which converge into a final assembly point. Since quantity input to the last stage must be the output from the previous stage, or stages, in the assembly situation as shown, the output from the three converging branches must be equal. This simplifies the problem considerably as each converging branch can be solved individually with the output of the branch required to be the input needed by the last stage for a specified final demand value. The total cost of the system through the assembly process would be composed of the sum of all of branches and the cost of final assembly.

The development of the dynamic programming model has focused on a manufacturing system with the stages limited to machining centers. Such a concept proceeds directly from the historical perspective of the problem and the practical consideration of limiting the scope of the problem to the most relevant aspects. This approach is also consistent with the incorporation of the machine parameters problem into the machine requirements problem structure where all machining processes are modelled as having discrete operating rates of operation and associated costs can be built into the optimization procedure.

A particular example of modelling a non-machining process would be material handling between machining stages. In the design of a manufacturing process, the type and number of material handling equipment may be critical. Given a set of alternative equipment with discrete

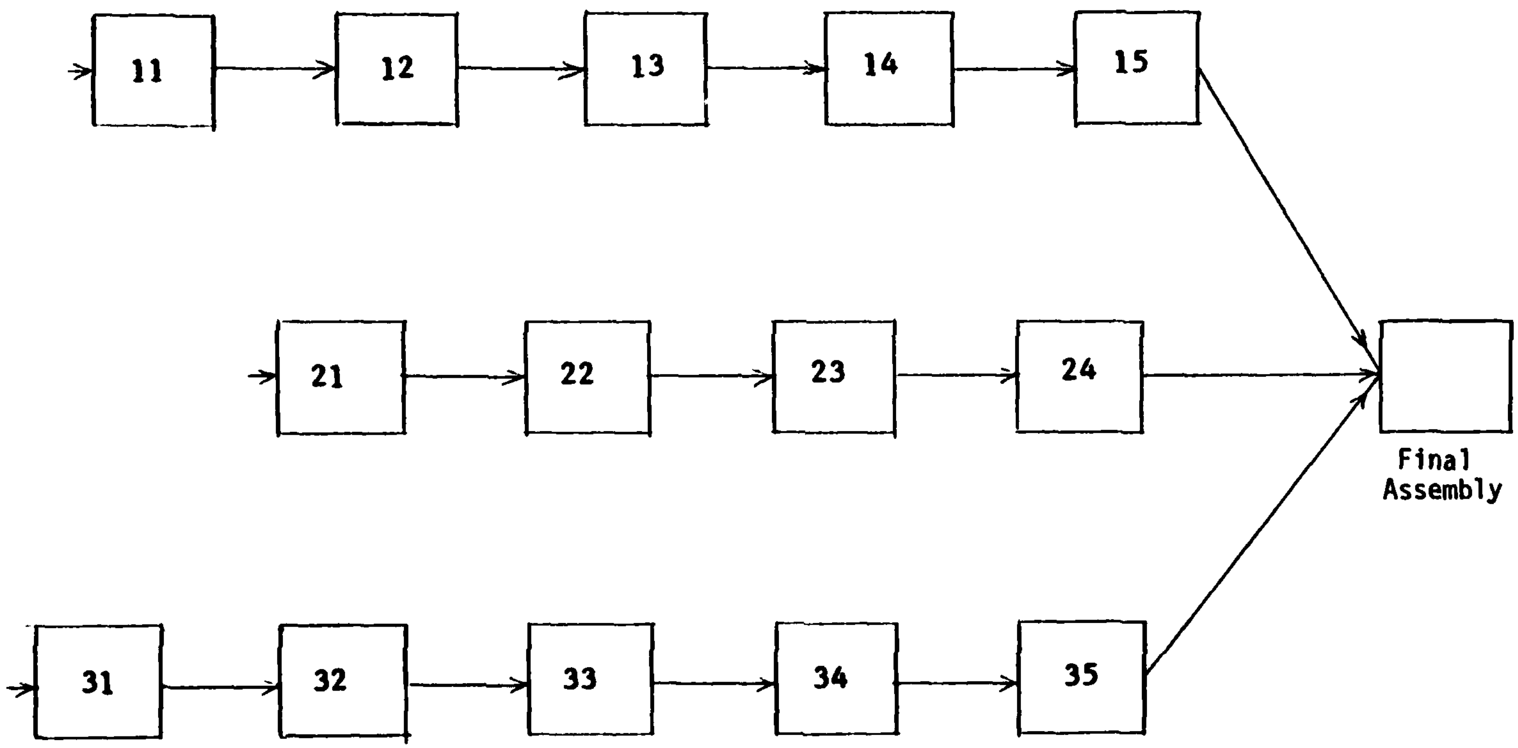


Figure 8.1: Converging branch manufacturing system.

operating rates, these rates could be used to develop the decision variables for a material handling stage between the machining stages. The operating cost per unit time must be known as well as the fixed equipment cost. The development of such an integrated concept would take the optimization of the manufacturing process a step further and limit the variability associated with actual implementation of the model due to restrictions imposed by materials handling equipment selection.

8.2 Extensions Related to Model Structure

In addition to extensions related to the DP solution procedure, the structure of the machine requirements model is such that a variety of situations can be incorporated. The model developed in Chapter VI utilizes a two shift production system but the time system employed in the model is highly flexible. The use of such modes as 3 shifts, 2 shifts with overtime and limitations to overtime can be readily achieved through variations in the constraints or the addition of new constraints. These constraint changes then lead to revising the set of decision variables in much the same way as increasing the number of rates or stages affected the test problem discussed in Chapter VII.

Two elements of the machine requirements model developed by Davis and Miller [7] can be included in the dynamic programming formulation. The Davis and Miller model is identical to the model developed in this research except for their inclusion of in-process inventories. Also, their model uses units processed as the decision variables in each stage rather than the hours used for production, but there is a direct relationship between these variables. Through analysis of the Davis and Miller model it can be seen that the effect of in-process inventories ultimately appears only in the objective function where storage cost can directly affect the cost incurred from a certain operating rate. Therefore, the DP formulation can be altered to include in-process inventories through modification of the cost data. In this way, the model can be designed to accommodate the cost of storage between stages as it affects the cost of the entire manufacturing system.

The Davis and Miller model is also designed to handle discretely distributed production demand through the use of probability data for distinct demand values and cost data for shortage and overage production. Such a feature can be readily added to the DP model if the probability information is available, since the calculations occur outside of the main optimization process. The DP formulation actually makes this feature more readily applicable since all of the optimization results are available for an entire range of production. The optimum cost for producing to each specific demand value can be easily calculated as shown in Table 7.2 and then the probability and cost data can be applied to determine the actual expected cost. Recourse to an integer formulation such as MISTIC requires a complete solution procedure for each distinct demand value for which probability information is available.

Although the main approach in this research has been to consider the optimization of a machining system from the design standpoint, the availability of solution data for an entire range of production readily allows extension to a production scheduling procedure. The fixed cost of a machine in the model formulation includes investment cost since the number of machines is being planned. For an optimization procedure of a production line already installed, the investment portion of the fixed daily cost is removed. An upper bound must also be placed on the number of machines allowed to be allocated to a machining center. The production time decision variables can readily be transformed to production quantity if desired by production personnel.

Chapter IX

SUMMARY AND CONCLUSIONS

In summary, it has been shown that it is possible to discretize the machining parameter problem to enable any type of machining equipment to be incorporated into the machine requirements planning problem. The machining parameter algorithm is more efficient than any other developed in the literature and is the only one which results in discrete operational data as shown in Table 3.2. The dynamic programming formulation of the machine requirements model has been shown to lead to a highly efficient and flexible solution procedure. The computational efficiency as depicted in Table 7.4, results in the ability to optimize complex multistage systems without incurring prohibitive solution costs and core storage requirements.

In many complex systems, branch and bound integer programming algorithms may prove to be incapable of reaching a solution in a feasible amount of time. Thus a DP approach is the only recourse if an optimal solution is to be obtained. The fact that solution information for an entire range of production demand is available also contributes greatly to reducing computational time when optimizing for a set of final demand values. The feasibility of the dynamic programming approach as demonstrated by the research can lead to a much greater use of optimization techniques in manufacturing processes.

The development of the discrete approach to the machining parameter problem is the key aspect of the research which allows the

DP model to be applied to either AC or DC machining systems. In addition, machining systems with any continous power source, such as hydraulic or AC systems with a continuously variable drive, can be discretized with the MP algorithm. Through the use of a basic, practical assumption, discrete operating rate and cost information is attainable for any machine center with continuous drive equipment. This data can then be incorporated into the stages of the MR model resulting in a direct interface between the two optimization procedures. The importance of the discrete nature of the MP formulation becomes clear with the realization that a high percentage of machinery is of a continuous nature. Although the machining parameters model can optimize the parameters for a particular process, this aspect of the procedure is unnecessary for the interface with the MR model. All feasible operating rate data is needed to optimize the total system. Attempting to limit the data to only rates which can be expected to be utilized would not significantly reduce the complexity of the solution procedure. This is supported by the results of the test to analyze the effect of increasing the number of rates. Only a slight increase in solution time was exhibited for the DP solution.

A distinct advantage of the dynamic programming approach to the MR problem is the variety of machining systems which can be effectively optimized. The extension of the model to non-serial systems without greatly increasing the complexity of the solution procedure allows optimization techniques to be applied to virtually all types of machining systems. The flexibility of the model is also demonstrated by the ease with which different aspects such as changes in time periods, demand probability data, and cost of storage between stages can be incorporated.

An integrated manufacturing system optimization model as developed in this research contributes significantly to machining technology state of the art. The systems approach using dynamic programming should lead to modeling additional aspects of the manufacturing process. With the development of efficient models which expand into all realms of manufacturing industry, the use of computers and automation can reach its full potential for improving industrial productivity.

REFERENCES

1. Apple, James M. Plant Layout and Material Handling, The Ronald Press Company, New York, 1950.

2. Barash, M. M. and Berra, P. B., "Automated Planning of Optimal Metal Cutting Operations and Its Effect on Machine Tool Design". Journal of Engineering for Industry, May 1971, pp. 545-551.

3. Boothroyd, G., and Rusek, P., "Maximum Rate of Profit Criteria in Machining", Transactions of the ASME, Paper No. 75-WA/Prod-22, 1975.

4. Challa, K., and Berra, P. B., "Automated Planning and Optimization of Machining Processes: A Systems Approach", Computers in Industrial Engineering, Vol. 1, 1976, pp. 35-45.

5. Claycombe, W. W., and Sullivan, W. G., "Use of Response Surface Methodology to Select a Cutting Tool to Maximize Profit", Journal of Engineering for Industry, Paper No. 75-Prod-H, 1975.

6. Davis, R. P., and Miller, D. M., "Determining the Optimum Number of Machines and Their Discrete Operating Rates for a Serial, Multistage Processing Center", Technical Report No. 78-01, Department of Industrial Engineering and Operations Research, Virginia Polytechnic Institute and State University, Blacksburg, VA., 1977.

7. Davis, R. P., and Miller, D. M., "A Model for Determining Machine Requirements in a Serial, Multistage Manufacturing System with Discretely Distributed Production Demand", (Journal of Applied Mathematical Modelling - to appear).

8. Davis, R. P., Wysk, R. A., and Agee, M. H., "Optimizing Machining Parameters in a Framework for Adaptive Control", Technical Report No. 78-02, Department of IEOR, VPI and SU, Blacksburg, Virginia, 1977.

9. Ermer, D. S., "Optimization of Constrained Machining Economic Problems by Geometric Programming", Journal of Engineering for Industry, Vol. 93, 1971, pp. 67-72.

10. Ermer, D. S., "A Bayesian Model of Machining Economics for Optimization by Adaptive Control", Journal of Engineering for Industry, August 1970, pp. 628-632.

11. Francis, R. L., and White, J. A., Facility Layout and Location: An Analytical Approach, Prentice-Hall, Inc., New Jersey, 1974.

12. Hati, S. K., and Rao, S. S., "Determination of Optimum Machining Conditions - Deterministic and Probabilistic Considerations", Journal of Engineering for Industry, Paper No. 75-Prod-k.

13. Hitomi, K., "Optimization of Multistage Machining Systems, Analysis of Optimal Machining Conditions for the Flow-Type Machining System", Journal of Engineering for Industry, May 1971, pp. 498-506.

14. Iwata, K., Murotsu, Y., Iwatsubo, T., and Fujii, S., "A Probabilistic Approach to the Determination of the Optimum Cutting Conditions", Journal of Engineering for Industry, November 1972, pp. 1099-1105.

15. Johnson, A. J. C., "The Calculation of Equipment Capacities", Time and Motion Study, Vol. 10, No. 2, 1958.

16. Kimbler, D. L., Wysk, R. A., and Davis, R. P., "Alternative Approaches to the Machining Parameter Optimization Problem", (Computers and Industrial Engineering - to Appear).

17. Lishchinski, L. Y., and Moshkov, E. A., "Principles of Designing Searching Systems for Machining Optimization", Machines and Tooling, Vol. 46, No. 11, 1975.

18. Miller, D. M., and Davis, R. P., "The Machine Requirements Problem", International Journal of Production Research, Vol. 15, No. 2, 1977, pp. 219-231.

19. Miller, D. M., and Davis, R. P., "A Dynamic Resource Allocation Model for the Machine Requirements Problem", (AIIE Transactions - to appear).

20. Moore, James M., Plant Layout and Design, The Macmillan Company, New York, 1962.

21. Morris, W. T., Engineering-Economic Analysis, R. D. Irwin, Inc., Homewood, Illinois, 1976.

22. Morris, W. T., "Facilities Planning", Journal of Industrial Engineering, Vol. 9, No. 5, Sept. - Oct. 1958.

23. Muther, R., Practical Plant Layout, McGraw-Hill Book Company, New York, 1955.

24. Nemhauser, G. L., Introduction to Dynamic Programming, Wiley, New York, 1966.

25. Petropoulos, P. G., Optimal Selection of Machining Rate Variables by Geometric Programming", International Journal of Production Research, Vol. 2, No. 4, 1973.

26. Ravignani, G. L., "Machining for a Fixed Demand - an Approach to Optimization", International Journal of Production Research, Vol. 12, No. 3, 1974, pp. 361-376.

27. Reed, R. Jr., Plant Layout: Factors, Principles, and Techniques, R. D. Irwin, Inc., Homewood, Illinois, 1961.

28. Reed, R. Jr., "A Probabilistic Model for Machine Determination", Proceedings of the Seventeenth Annual Insitute Conference and Convention, AIIE 1966, AIIE, New York, New York, pp. 153-164.

29. Shubin, J. A., and Madeheim, H., Plant Layout, Prentice Hall, Inc., New York, 1951.

30. Wilde, D. J., and Beightler, C. S., Foundations of Optimization, Prentice-Hall Inc., New Jersey, 1967.

31. Wu, S. M., and Ermer, D. S., "Maximum Profit as the Criteria in the Determination of the Optimum Machining Conditions", Journal of Engineering for Industry, Vol. 88, 1966, pp. 435-442.

32. Wysk, R. A., Davis, R. P., and Agee, M. H., "Optimizing Machining Parameters in a Realistic Manufacturing Environment", Proceedings of the North American Metalworking Research Conference VI, Gainesville, Florida, April 1978.

APPENDIX A

COMPUTER CODE FOR MACHINING PARAMETER OPTIMIZATION PROCEDURE

```
C
C  DETERMINATION OF OPTIMAL MACHINING PARAMETERS USING DISCRETIZED
C  SOLUTION PROCEDURE
C
C
C  NOTATION FOR MACHINE PARAMETER OPTIMIZATION ALGORITHM
C
C     NT     NUMBER OF PASSES PER TOOL LIFE
C     NP     NUMBER OF PASSES PER PART
C     CPT    MACHINING COST PER PART ($)
C     DIAM   DIAMETER OF THE WORKPIECE (MM)
C     L      LENGTH OF THE CUT TO BE MADE (MM)
C     F      FEED RATE OF THE WORKPIECE (MM/REV)
C     TL     TOOL LIFE (MIN)
C     C      TAYLOR TOOL LIFE CONSTANT
C     ALPHA, BETA, GAMMA       TOOL LIFE EXPONENTS
C     DC     DEPTH OF MATERIAL TO BE REMOVED (MM)
C     D      DEPTH OF CUT PER PASS (MM)
C     CCUT   COST OF CUTTING ($/MIN)
C     CCH    COST OF TOOL CHANGING ($/MIN)
C     TR     TIME TO REPLACE A TOOL (MIN)
C     TH     TOOL THRUST (KG)
C     M      TORQUE ON THE TOOL (KGM)
C     P      POWER AT THE TOOL (KW)
C     V      CUTTING SPEED (MM/MIN)
C     VMINAT MINIMUM ATTAINABLE SPEED (MM/MIN)
C     VMAXAT MAXIMUM ATTAINABLE SPEED (MM/MIN)
C
      REAL MMAX, NTMIN, NT, MT, NTFEAS, MP, NTOPT, NTMAX
      DIMENSION NT(100),NP(99),CPT(1000),TLFSBL(1000),NTFEAS(1000),NPFEA
     1S(1000),F(1000),D(1000),V(1000)
      READ, PI,DIAM,L,VMAX,VMIN,TLMIN,TLMAX,FMIN,FMAX,C,ALPHA,BETA,GAMMA
```

```
     1,DC,DMAX,DMIN,CCUT,CCH,TR,THMAX,MMAX,PMAX
C
C  DEFINITION OF VARIABLES FOR EXPONENTS AND ALGEBRAIC CONSTANTS
C
      E1=GAMMA/ALPHA
      E2=BETA/ALPHA-1
      E3=1/(1-1/ALPHA)
      E4=1/ALPHA
      E5=1/E2
      E6=1/(E4+(1-E4)/E2)
      E7=E2/0.72
      E8=E2/0.75
      E9=1-E1
      E10=0.75-BETA/ALPHA
      E11=1/(E4-(1-E4)*E10)
      C1=(PI*DIAM*L*DC**E1)/C**E4
      C2=C**E4/(PI*DIAM*L*DC**E1)
      C3=C**E4/DC**E1
      C4=THMAX/(2.02*DC)
      C5=MMAX/(0.084*DC*DIAM)
      C6=(C**E4*DIAM*DC**E9)/(71811.548*PMAX)
C
C DETERMINATION OF NP AND NT SETS
C
      NPMIN=INT((DC/DMAX)+0.99999)
      NPMAX=INT(DC/DMIN)
      NPSTOP=NPMAX-NPMIN+1
      DO 5 K=1,NPSTOP
      NP(K)=NPMIN+K-1
    5 CONTINUE
      VMINAT=(C/(TLMAX*(DC/NPMIN)**GAMMA*FMAX**BETA))**E4
      NTMIN=(PI*DIAM*L)/(TLMAX*VMINAT*FMAX)
```

```
      INTMAX=INT(1/NTMIN)
      VMAXAT=(C/(TLMIN*(DC/NPMAX)**GAMMA*FMIN**BETA))**E4
      NTMAX=(PI*DIAM*L)/(TLMIN*VMAXAT*FMIN)
      INTMIN=INT(1/NTMAX)
      N=INTMAX-INTMIN+1
      DO 10 I=1,N
      NT(I)=1./(INTMAX-(I-1))
   10 CONTINUE
C
C DETERMINATION OF FEASIBILITY OF NT AND NP
C
      L=1
      DO 20 K=1,NPSTOP
      MP=NP(K)
      I=1
      MT=NT(I)
      TLFDUP=((C1*FMIN**E2)/(MT*MP**E1))**E3
      TLFDLO=((C1*FMAX**E2)/(MT*MP**E1))**E3
      IF ((TLFDUP.LT.TLMIN).OR.(TLFDLO.GT.TLMAX)) GO TO 30
      TLSPUP=((C3*MP**E1)/(VMAX*(C2*MT*MP**E1)**E5))**E6
      TLSPLO=((C3*MP**E1)/(VMIN*(C2*MT*MP**E1)**E5))**E6
      IF ((TLSPUP.LT.TLMIN).OR.(TLSPLO.GT.TLMAX)) GO TO 30
      TLTHST=(((C4*MP)**E7)*(C1/(MT*MP**E1)))**E3
      TLSTQ=(((C5*MP)**E8)*(C1/(MT*MP**E1)))**E3
      IF (TLSTQ.GT.TLMAX) GO TO 30
      TLPWR=((C6/MP**E9)*(C2*MT*MP**E1)**E10)**E11
      IF (TLPWR.LT.TLMIN) GO TO 30
C
C CALCULATION OF BINDING LOWER CONSTRAINT ON TOOL LIFE (MAX OF LO BNDS)
C
      TLFEAS=TLMIN
      IF (TLFEAS.LT.TLFDLO) TLFEAS=TLFDLO
```

```
      IF (TLFEAS.LT.TLSPLO) TLFEAS=TLSPLO
      IF (TLFEAS.LT.TLTHST) TLFEAS=TLTHST
      IF (TLFEAS.LT.TLSTQ) TLFEAS=TLSTQ
C
C DETERMINATION IF MINIMUM FEASIBLE TOOL LIFE VIOLATES ANY UPPER BOUND
C
      IF (TLFEAS.GT.TLFDUP) GO TO 30
      IF (TLFEAS.GT.TLSPUP) GO TO 30
      IF (TLFEAS.GT.TLPWR) GO TO 30
C
C CALCULATION OF OBJECTIVE FUNCTION VALUE
C  UTILIZING ADAPTIVE CONTROL PAPER FORMULATION
C
      NPFEAS(L)=MP
      NTFEAS(L)=MT
      TLFSBL(L)=TLFEAS
      D(L)=DC/NPFEAS(L)
      F(L)=(TLFSBL(L)**(1-E4)*C2*NTFEAS(L)*NPFEAS(L)**E1)**E5
      V(L)=(C**E4)/(TLFSBL(L)**E4*F(L)**(BETA/ALPHA)*D(L)**E1)
      CPT(L)=(NTFEAS(L)*NPFEAS(L))*(CCUT*TLFSBL(L)+CCH*TR)
      L=L+1
   30 CONTINUE
   20 CONTINUE
C
C DETERMINATION OF MINIMUM OBJECTIVE FUNCTION VALUE
C
      M=1
      L=L-1
      CPTOPT=CPT(1)
      KEEP=1
      DO 40 M=2,L
      IF (CPTOPT.LT.CPT(M)) GO TO 40
```

```
      CPTOPT=CPT(M)
      KEEP=M
   40 CONTINUE
      NPOPT=NPFEAS(KEEP)
      NTOPT=NTFEAS(KEEP)
      TLOPT=TLFSBL(KEEP)
C
C CALCULATION OF OPTIMAL VALUES OF MACHINING PARAMETERS
C
      FOPT=(TLOPT**(1-E4)*C2*NTOPT*NPOPT**E1)**E5
      DOPT=DC/NPOPT
      VOPT=(C**E4)/(TLOPT**E4*FOPT**(BETA/ALPHA)*DOPT**E1)
      WRITE(6,100)
  100 FORMAT('1',T14,'ITERATION NO.',5X,'NP',10X,'NT',10X,'TOOL LIFE',10
     1X,'FEED',8X,'SPEED',6X,'DEPTH OF CUT',6X,'COST PER PART')
      I=1
      DO 60 I=1,L
      WRITE(6,200) I,NPFEAS(I),NTFEAS(I),TLFSBL(I),F(I),V(I),D(I),CPT(I)
  200 FORMAT('0',T20,I3,10X,I2,8X,F8.5,8X,F8.5,8X,F8.5,5X,F12.4,5X,F8.5,
     18X,F10.5)
   60 CONTINUE
      WRITE (6,300) KEEP
  300 FORMAT('-','THE OPTIMAL MACHINING PARAMETERS ARE AT ITERATION NUMB
     1ER:',2X,I3)
      WRITE (6,400) FOPT, VOPT, DOPT
  400 FORMAT ('0','THE OPTIMAL FEED RATE IS:  ',F8.5,//,'THE OPTIMAL SPI
     1NDLE SPEED IS:  ',F12.5,//,'THE OPTIMAL DEPTH OF CUT IS:  ',F8.5)
      STOP
      END
```

APPENDIX B

COMPUTER CODE FOR MACHINE REQUIREMENTS PLANNING MODEL SOLUTION PROCEDURE

```
C
C  MACHINE REQUIREMENTS PLANNING MODEL SOLUTION ALGORITHM
C
C    THE MR SOLUTION PROCEDURE USES A DYNAMIC PROGRAMMING FORMULATION
C    WITH LINEAR PROGRAMMING STEPS AT EACH STAGE OF A MULTISTAGE MACHINING
C    PROCESS TO OPTIMIZE THE NUMBER OF MACHINES FOR THE MACHINING SYSTEM
C
C
C  NOTATION FOR MACHINE REQUIREMENTS COMPUTER ALGORITHM
C
C     JJ        -  NUMBER OF RATES (VARIABLES) AT A STAGE
C     PERDEF    -  PERCENT DEFECTIVES AT A STAGE
C     FC        -  FIXED COST OF EQUIPMENT PER DAY ($)
C     RNGLIM    -  UPPER LIMIT OF PLANNED PRODUCTION (UNITS)
C     RNGRED    -  REDUCTION FACTOR TO REDUCE RNGLIM AT EACH STAGE
C     RRATE     -  RATE REDUCED FOR DEFECTIVES (UNITS/HOUR)
C     A1        -  INITIAL 'A' MATRIX FOR SIMPLEX TABLEAU
C     RHS1      -  INITIAL RIGHT HAND SIDE OF TABLEAU FOR EACH STAGE
C     B         -  BASIS MATRIX
C     COSTBS    -  BASIC VARIABLE COST VALUE
C     BAS       -  INDEX OF VARIABLE IN BASIS
C     RVAR      -  VARIABLE COST PORTION OF OBJECTIVE FUNCTION ($)
C     RFIX      -  FIXED COST PORTION OF OBJECTIVE FUNCTION ($)
C     RNGLO     -  LOWER BOUND OF A PRODUCTION RANGE
C     RNGUP     -  UPPER BOUND OF A PRODUCTION RANGE
C     M         -  NUMBER OF MACHINES IN OPERATION AT A STAGE
C     STRC      -  FIRST SHIFT REDUCED COST FROM SIMPLEX TABLEAU ($)
C     OTRC      -  SECOND SHIFT REDUCED COST ($)
C     TOTRC     -  TOTAL REDUCED COST ($)
C     LCOUNT    -  COUNTER TO DETERMINE THE NUMBER OF RANGES PER STAGE
C     TRANSF    -  FIXED PORTION OF TRANSITION FUNCTION FOR STAGE INPUT
C     TRANSV    -  VARIABLE PORTION OF TRANSITION FUNCTION
```

```
C      TIMEF      -  FIXED PORTION OF PRODUCTION TIME (HOURS)
C      TIMEV      -  VARIABLE PORTION OF PRODUCTION TIME (HOURS)
C      RNGADJ     -  ADJUSTED RANGE VALUE DUE TO RETURNS CHANGE IN I-1
C      BALPT      -  PRODUCTION POINT AT WHICH ADDING A MACHINE IS
C                       ECONOMICALLY FEASIBLE
C
C
      REAL MTHETA, MINRTO
      INTEGER X, Q, H, ROWS, Z, ENTER, BAS
C
      DIMENSION A1(10,4,30), BAS(10,90,4), RHS(10,90,4), COST(10,30),
     1B(10,4,8), BINV(10,4,8),RHS1(10,1,4), RFIX(10,90), RVAR(10,90), RN
     2GUP(10,90), RNGLO(10,90), M(10,90), FC(10), A(10,4,30), ZJCJ(30),
     3RRATE(10,30), RATIO(30), THETA(30), RATE(10,30), TRANSF(10,90),
     4COSTBS(4), TRANSV(10,90), JJ(10), RNGRED(10), PERDEF(10,30),
     5OPRATE(10,90,3), TIMEF(10,90,3), TIMEV(10,90,3), LCOUNT(10)
      COMMON K, L, Q, A1, A, BAS, B, BINV, RHS1, RHS, COST, RFIX, RVAR,
     1RNGUP, RNGLO, M, FC, ZJCJ, RATE, RRATE, TRANSF, TRANSV, COSTBS,
     2JJK, MARKQ, OPRATE, TIMEF, TIMEV
      DATA A/1200*0./,BAS/3600*0/,RHS/3600*0./,COST/300*0./,B/320*0./,
     1BINV/320*0./,RHS1/40*0./,RFIX/900*0./,RVAR/900*0./,RNGUP/900*0./,
     2RNGLO/900*0./,M/900*0/,FC/10*0./,A1/1200*0./,ZJCJ/30*0./,RRATE/300
     3*0./,RATIO/30*0./,THETA/30*0./,RATE/300*0./,TRANSF/900*0./,COSTBS
     4/4*0./,TRANSV/900*0./,JJ/10*0/,RNGRED/10*1.1/,PERDEF/300*0./,OPRAT
     5E/2700*0./,TIMEF/2700*0./,TIMEV/2700*0./,LCOUNT/10*0/
C
C   INPUT PRODUCTION DATA FOR ALL STAGES
C
      READ, N, RNGLIM
      READ, (JJ(K), K=1,N)
      DO 3 K=1,N
      IF (K.EQ.1) ROWS=3
```

```
      IF (K.GT.1) ROWS=4
      JJH=JJ(K)/2
      READ, (RATE(K,J), J=1,JJH)
      READ, (PERDEF(K,J), J=1,JJH)
    3 CONTINUE
      DO 11 K=1,N
      JJK=JJ(K)
      READ, (COST(K,J), J=1,JJK)
   11 CONTINUE
      READ, (FC(I), I=1,N)
      DO 10 K=1,N
      RNGLIM=RNGLIM/RNGRED(K)
      M(K,1)=1
      IF (K.EQ.1) II=3
      IF (K.GT.1) II=4
      L=1
      Q=1
      JJK=JJ(K)
C
C   FORMATION OF INITIAL CONSTRAINT MATRIX FROM INPUT DATA
C
      JJH=JJK/2
      DO 12 J=1,JJH
   12 RPATE(K,J)=RATE(K,J)*(1-PERDEF(K,J))
      DO 14 J=1,JJH
      A1(K,1,J)=RRATE(K,J)
      A1(K,2,J)=1.0
      JJOT=J+JJH
      RATE(K,JJOT)=RATE(K,J)
      A1(K,1,JJOT)=RRATE(K,J)
      A1(K,3,JJOT)=1.0
      IF (K.GT.1) A1(K,4,J)=-RATE(K,J)
```

```
      IF (K.GT.1) A1(K,4,JJOT)=-RATE(K,J)
   14 CONTINUE
C
C  FORMATION OF IDENTITY MATRIX TO AUGMENT INITIAL CONSTRAINT MATRIX
C
      JA=JJK+1
      J2=JJK+2
      J3=JJK+3
      J4=JJK+4
      A1(K,1,JA)=1.0
      A1(K,2,J2)=1.0
      A1(K,3,J3)=1.0
      IF (K.GT.1) A1(K,4,J4)=1.0
C
C  DETERMINF LOWEST COST PRODUCTION RATE FOR ENTERING VARIABLE
C
      DO 15 J=1,JJK
      RATIO(J)=ABS(COST(K,J))/A1(K,1,J)
      PRINT, RATIO(J)
      IF (J.GT.1) GO TO 16
      MINRTO=RATIO(1)
      KEEP=1
   16 IF (RATIO(J).LT.MINRTO) GO TO 17
      GO TO 15
   17 MINRTO=RATIC(J)
      KEEP=J
   15 CONTINUE
      PRINT, KEEP
      L=1
C
C  ESTABLISH INITIAL RHS VALUES FOR LP TABLEAU
C
```

AD-A084 062 ARMY MILITARY PERSONNEL CENTER ALEXANDRIA VA F/G 13/8

A GENERALIZED MACHINE REQUIREMENTS PLANNING ALGORITHM FOR SERIA--ETC(U)

JUL 78 G M HAYES

UNCLASSIFIED NL

2 OF 2

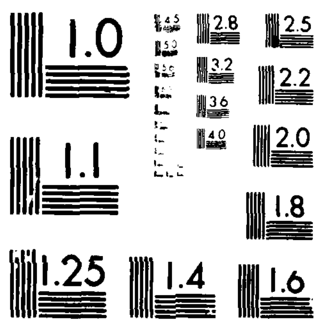
AD A084 062

END

DATE FILMED

6-80

DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

```
      RHS1(K,1,1)=8.0*RRATE(K,KEEP)
      RHS1(K,1,2)=RHS1(K,1,3)=8.0
      RHS1(K,1,4)=0.0
C
C  SET UP BASIS MATRIX
C
      JA=JJK+1
      J2=JJK+2
      J3=JJK+3
      IF (K.GT.1) J4=JJK+4
      DO 21 I=1,II
      B(K,I,1)=A1(K,I,KEEP)
      IF (K.EQ.1) JF=J3
      IF (K.GT.1) JF=J4
      DO 22 J=J2,JF
      KK=J-JJK
      B(K,I,KK)=A1(K,I,J)
   22 CONTINUE
   21 CONTINUE
      COSTBS(1)=ABS(COST(K,KEEP))
      COSTBS(2)=ABS(COST(K,J2))
      COSTBS(3)=ABS(COST(K,J3))
      CALL INV (II,&13)
C
C  UPDATING THE CONSTRAINT MATRIX, COST ROW AND RHS
C
C
C  PARAMETER 'IPARAM' INDICATES THE POINT IN SUBROUTINE UPDATE FROM WHICH
C  TO RETURN.  IF IPARAM = 1, ONLY TABLEAU UPDATING IS DONE;  IF IPARAM = 2,
C  UPDATING AND RANGE CALCULATING IS DONE.
C
      KK=K-1
```

```
      IPARAM=2
      IF (K.EQ.1) GO TO 23
      COST(K,J4)=RVAR(KK,Q)
      COSTBS(4)=ABS(COST(K,J4))
   23 BAS(K,L,1)=KEEP
      BAS(K,L,2)=J2
      BAS(K,L,3)=J3
      IF (K.GT.1) BAS(K,L,4)=J4
      CALL UPDATE (NEG, ENTER, KEEP, IPARAM, &99)
C
C  INITIAL RANGES WHICH SUPERCEDE THOSE  DETERMINED IN UPDATE
C
      RVAR(K,1)=COST(K,KEEP)/RRATE(K,KEEP)
      RFIX(K,L)=M(K,L)*FC(K)
      IF (K.EQ.1) RNGUP(K,1)=RHS1(K,1,1)
      RNGLO(K,1)=0
C
C  DETERMINE IF UPPER LIMIT OF PLANNED PRODUCTION HAS BEEN REACHED.  IF
C  NOT, CONTINUE LP ITERATIONS TO ESTABLISH PRODUCTION RANGES.
C
   24 IF (RNGUP(K,L).GE.RNGLIM) GO TO 8
      L=L+1
      LML=L-1
      DO 40 I=1,II
      BAS(K,L,I)=BAS(K,LML,I)
   40 CONTINUE
      M(K,L)=M(K,LML)
C
C  ROUTINE TO ESTABLISH THE UPPER PORTION OF A RANGE SPLIT
C
C  PARAMETER 'MARKQ' INDICATES A RANGE SPLIT OCCURRED IN THE LAST UPDATE
C  ROUTINE IF MARKQ = 1.
```

```
C
      IF (MARKQ.EQ.1) GO TO 42
      GU TO 44
   42 IPARAM=2
      CALL UPDATE (NEG, ENTER, KEEP, IPARAM, &99)
      GO TO 30
   44 CONTINUE
      CALL PRELIM (NEG, ENTER, II, JA, KEEP, IPARAM, &24)
      CALL INV (II,&13)
      IPARAM=2
      CALL UPDATE (NEG, ENTER, KEEP, IPARAM, &99)
C
C  CHECK IF ADDING ANOTHER MACHINE SHOULD BE INVESTIGATED
C
C  IF ADDING A MACHINE MAY BE FEASIBLE, CALL MACHCK SUBROUTINE TO MAKE
C  AN ECONOMIC COMPARISON
C
   30 STRC=ABS(8.*ZJCJ(J2))
      OTRC=ABS(8*ZJCJ(J3))
      TOTRC=STRC+OTRC
      IF (TOTRC.GT.FC(K)) GO TO 32
      GO TO 37
   32 CALL MACHCK (NEG, ENTER, KEEP, II, IPARAM)
   37 GU TO 24
    8 LCOUNT(K)=L
   10 CONTINUE
C
C  OUTPUT OF PRODUCTION DATA FOR PROBLEM SOLUTION
C
      WRITE (6,120)
  120 FORMAT ('1',T50,'RECAP OF CALCULATED PRODUCTION INFORMATION',///)
      DO 50 K=1,N
```

```
      WRITE (6,130)
  130 FORMAT ('1',T10,'STAGE',4X,'RNGLO',4X,'RNGUP',4X,'MACH',4X,'TRANSF
     2',4X,'TRANSV',4X,'FIXRTN',4X,'VARRTN',4X,'BASVRBL',4X,'OPRRATE',4X
     3,'FIXTIME',4X,'VARTIME',/)
      LSTOP=LCOUNT(K)
      DO 52 L=1,LSTOP
      IF (L.GE.7) LPAGE=MOD(L,7)
      IF (L.LT.7) GO TO 82
      IF (LPAGE.EQ.0) GO TO 80
      GO TO 82
   80 WRITE (6,130)
   82 CONTINUE
      WRITE (6,135) K,RNGLO(K,L),RNGUP(K,L),M(K,L),TRANSF(K,L),TRANSV(K,
     1L), RFIX(K,L), RVAR(K,L)
  135 FORMAT ('0',T12,I2,4X,F7.2,2X,F7.2,4X,I2,4X,F7.2,3X,F7.4,2X,F8.2,3
     1X,F7.4)
      DO 54 I=1,3
      WRITE (6,140) BAS(K,L,I), OPRATE(K,L,I), TIMEF(K,L,I),TIMEV(K,L,I)
  140 FORMAT ('0',T87,I2,7X,F6.2,3X,F7.2,6X,F7.4)
   54 CONTINUE
   52 CONTINUE
   50 CONTINUE
   99 CONTINUE
   13 STOP
      END
```

```
C
C   SUBROUTINE TO DETERMINE MINIMUM THETA RATIO AND THE ENTERING AND
C   LEAVING VARIABLE USING THE DUAL SIMPLEX PROCEDURE.
C
      SUBROUTINE PRELIM (NEG, ENTER, II, JA, KEEP, IPARAM, *)
      DIMENSION A1(10,4,30), BAS(10,90,4), RHS(10,90,4), COST(10,30),
     1B(10,4,8), BINV(10,4,8),RHS1(10,1,4), RFIX(10,90), RVAR(10,90), RN
     2GUP(10,90), RNGLO(10,90), M(10,90), FC(10), A(10,4,30), ZJCJ(30),
     3RRATE(10,30), RATIO(30), THETA(30), RATE(10,30), TRANSF(10,90),
     4COSTBS(4), TRANSV(10,90), JJ(10), RNGRED(10), PERDEF(10,30),
     5OPRATE(10,90,3), TIMEF(10,90,3), TIMEV(10,90,3), LCOUNT(10)
      COMMON K, L, Q, A1, A, BAS, B, BINV, RHS1, RHS, COST, RFIX, RVAR,
     1RNGUP, RNGLO, M, FC, ZJCJ, RATE, RRATE, TRANSF, TRANSV, COSTBS,
     2JJK, MARKQ, OPRATE, TIMEF, TIMEV
C
C   DETERMINE WHICH ROW HAS THE NEGATIVE  VALUE IN THE ARTIFICIAL COLUMN
C
      INTEGER ENTER, BAS, Q
      REAL MTHETA
      JA=JJK+1
   22 DO 15 I=1,II
      IF (A(K,I,JA).LT.-1.E-04) GO TO 16
      GO TO 15
   16 NEG=I
      GO TO 35
   15 CONTINUE
   35 CONTINUE
      MARK=0
      MTHETA=1000000.0
C
C   USE MINIMUM RATIO RULE TO DETERMINE ENTERING BASIC VARIABLE
C
```

```
      DO 28 J=1,JJK
      IF (A(K,NEG,J).GE.-1.E-04) GO TO 27
      THETA(J)=ZJCJ(J)/A(K,NEG,J)
   29 IF (THETA(J).LT.MTHETA) GO TO 30
      GO TO 28
   30 MTHETA=THETA(J)
      ENTER=J
      GO TO 28
   27 MARK=MARK+1
   28 CONTINUE
      IF (MARK.EQ.JJK) GO TO 36
C
C   REVISE BASIS MATRIX FOR ENTERING AND LEAVING VARIABLE
C
      DO 31 I=1,II
      B(K,I,NEG)=A1(K,I,ENTER)
   31 CONTINUE
      COSTBS(NEG)=ABS(COST(K,ENTER))
      BAS(K,L,NEG)=ENTER
      GO TO 40
C
C   INCREASE IN NUMBER OF MACHINES DUE TO EXHAUSTING ALL RATES BEFORE RNGLIM
C
   36 M(K,L)=M(K,L)+1
      CALL MACHAD (NEG, ENTER, II, KEEP, IPARAM)
      RETURN 1
   40 RETURN
      END
```

```
C
C   SUBROUTINE TO DETERMINE THE NEGATIVE RHS VALUE AND BASIS CHANGE
C   WHEN DRIVING THE SOLUTION BACK TO FEASIBILITY AFTER ADDING A MACHINE
C
      SUBROUTINE RHSCK (NEG, ENTER, II, KEEP)
      DIMENSION A1(10,4,30), BAS(10,90,4), RHS(10,90,4), COST(10,30),
     1B(10,4,8), BINV(10,4,8),RHS1(10,1,4), RFIX(10,90), RVAR(10,90), RN
     2GUP(10,90), RNGLO(10,90), M(10,90), FC(10), A(10,4,30), ZJCJ(30),
     3RRATE(10,30), RATIO(30), THETA(30), RATE(10,30), TRANSF(10,90),
     4COSTBS(4), TRANSV(10,90), JJ(10), RNGRED(10), PERDEF(10,30),
     5OPRATE(10,90,3), TIMEF(10,90,3), TIMEV(10,90,3), LCOUNT(10)
      COMMON K, L, Q, A1, A, BAS, B, BINV, RHS1, RHS, COST, RFIX, RVAR,
     1RNGUP, RNGLO, M, FC, ZJCJ, RATE, RRATE, TRANSF, TRANSV, COSTBS,
     2JJK, MARKQ, OPRATE, TIMEF, TIMEV
      INTEGER ENTER, BAS, Q
      REAL MTHETA
      IF (II.EQ.3) JJF=JJK+3
      IF (II.EQ.4) JJF=JJK+4
C
C   DETERMINE NEGATIVE RHS VALUE ON WHICH TO DUAL SIMPLEX
C
      DO 25 I=1,II
      IF (RHS(K,L,I).LT.-1.E-04) GO TO 26
      GO TO 25
   26 NEG=I
      GO TO 35
   25 CONTINUE
      GO TO 40
   35 CONTINUE
      MARK=0
      MTHETA=1000000.0
C
```

```
C   USE MINIMUM RATIO RULE TO DETERMINE ENTERING BASIC VARIABLE
C
       DO 28 J=1,JJF
       IF (A(K,NEG,J).GE.-1.E-04) GO TO 27
       THETA(J)=ZJCJ(J)/A(K,NEG,J)
    29 IF (THETA(J).LT.MTHETA) GO TO 30
       GO TO 28
    30 MTHETA=THETA(J)
       ENTER=J
       GO TO 28
C
C   'MARK' IS A COUNTER TO CHECK IF FEASIBILITY CAN BE RESTORED
C
    27 MARK=MARK+1
    28 CONTINUE
       IF (MARK.EQ.JJF) GO TO 36
C
C   REVISE BASIS MATRIX FOR ENTERING AND LEAVING VARIABLE
C
       DO 31 I=1,II
       B(K,I,NEG)=A1(K,I,ENTER)
    31 CONTINUE
       COSTBS(NEG)=ABS(COST(K,ENTER))
       BAS(K,L,NEG)=ENTER
       GO TO 40
    36 PRINT, 'THE SOLUTION IS INFEASIBLE'
       STOP
    40 RETURN
       END
C
```

```
C
C   SUBROUTINE TO FIND THE INVERSE OF THE BASIS MATRIX USING THE GAUSS
C   ELIMINATION PROCEDURE.
C
      SUBROUTINE INV (II,*)
      DIMENSION A1(10,4,30), BAS(10,90,4), RHS(10,90,4), COST(10,30),
     1B(10,4,8), BINV(10,4,8),RHS1(10,1,4), RFIX(10,90), RVAR(10,90), RN
     2GUP(10,90), RNGLO(10,90), Y(10,90), FC(10), A(10,4,30), ZJCJ(30),
     3RRATE(10,30), RATIO(30), THETA(30), RATE(10,30), TRANSF(10,90),
     4CLSTBS(4), TRANSV(10,90), JJ(10), RNGRED(10), PERDEF(10,30),
     5OPRATE(10,90,3), TIMEF(10,90,3), TIMEV(10,90,3), LCOUNT(10)
      DIMENSION BHOLD(1,4,8)
      COMMON K, Z, Q, A1, A, BAS, B, BINV, RHS1, RHS, COST, RFIX, RVAR,
     1RNGUP, RNGLO, Y, FC, ZJCJ, RATE, RRATE, TRANSF, TRANSV, COSTBS,
     2JJK, MARKQ, OPRATE, TIMEF, TIMEV
      DATA BHOLD/32*0./
      INTEGER Z, Y, R, Q
      M=II
      M2=2*M
      DO 32 I=1,II
      DO 34 J=1,II
C
C   HOLD BASIS FOR RECONSTRUCTING AT END OF SUBROUTINE.
C
      BHOLD(1,I,J)=B(K,I,J)
   34 CONTINUE
   32 CONTINUE
C
C   ADD IDENTITY MATRIX TO RIGHT OF BASIS MATRIX
C
      JC=II+1
      DO 11 I=1,II
```

```
      DO 12 J=JO,M2
      B(K,I,J)=0.0
   12 CONTINUE
   11 CONTINUE
      IF (K.EQ.1) GO TO 14
      DO 15 I=1,II
      JI=I+4
      B(K,I,JI)=1
   15 CONTINUE
      GO TO 17
   14 DO 16 I=1,II
      JI=I+3
      B(K,I,JI)=1
   16 CONTINUE
   17 DO 50 I=1,M
      I1=I+1
      IF (ABS(B(K,I,I)).LE.1.E-06) GO TO 85
C
C   CBTAINING UPPER TRIANGULAR MATRIX
C
   52 T=B(K,I,I)
      DO 60 J=I,M2
   60 B(K,I,J)=B(K,I,J)/T
      DO 70 R=I1,M
      IF (I1.GT.M) GO TO 70
      T1=B(K,R,I)
      DO 80 L=I,M2
      B(K,R,L)=B(K,R,L)-T1*B(K,I,L)
   80 CONTINUE
   70 CONTINUF
      GO TO 50
C
```

```
C  ROW SWITCHING TO ELIMINATE '0' ON DIAGONAL OF BASIS
C
   85 DO 90 J=I1,M
      IF (ABS(B(K,J,I)).LE.1.E-06) GO TO 90
      DO 95 R=1,M2
      T2=B(K,I,R)
      B(K,I,R)=B(K,J,R)
      B(K,J,R)=T2
   95 CONTINUE
      GO TO 52
   90 CONTINUE
      PRINT, 'THE BASIS MATRIX IS SINGULAR AND CANNOT BE INVERTED'
      RETURN 1
   50 CONTINUE
C
C  BACKSUBSTITUTION TO OBTAIN MATRIX INVERSE
C
      DO 10 N=1,M
      M1=M+N
      DO 20 NV=2,M
      L=M+1-NV
      M5=M-L
      DO 30 NN=1,M5
      M3=M+1-NN
      B(K,L,M1)=B(K,L,M1)-B(K,L,M3)*B(K,M3,M1)
   30 CONTINUE
   20 CONTINUE
   10 CONTINUE
      DO 40 L=1,M
      DO 45 N=1,M
      M1=M+N
      BINV(K,L,N)=B(K,L,M1)
```

```
   45 CONTINUE
   40 CONTINUE
      DO 53 I=1,II
      DO 54 J=1,II
      B(K,I,J)=BHOLD(1,I,J)
   54 CONTINUE
   53 CONTINUE
      RETURN
      END
C
```

```
C
C   SUBROUTINE TO UPDATE CONSTRAINT MATRIX, ZJCJ'S, RHS, RETURNS AND
C   PRODUCTION RANGE INFORMATION.
C
       SUBROUTINE UPDATE (NEG, ENTER, KEEP, IPARAM, *)
       DIMENSION A1(10,4,30), BAS(10,90,4), RHS(10,90,4), COST(10,30),
      1B(10,4,8), BINV(10,4,8),RHS1(10,1,4), RFIX(10,90), RVAR(10,90), RN
      2GUP(10,90), RNGLO(10,90), M(10,90), FC(10), A(10,4,30), ZJCJ(30),
      3RRATE(10,30), RATIO(30), THETA(30), RATE(10,30), TRANSF(10,90),
      4COSTBS(4), TRANSV(10,90), JJ(10), RNGRED(10), PERDEF(10,30),
      5OPRATE(10,90,3), TIMEF(10,90,3), TIMEV(10,90,3), LCOUNT(10)
       COMMON K, L, Q, A1, A, BAS, B, BINV, RHS1, RHS, COST, RFIX, RVAR,
      1RNGUP, RNGLO, M, FC, ZJCJ, RATE, RRATE, TRANSF, TRANSV, COSTBS,
      2JJK, MARKQ, OPRATE, TIMEF, TIMEV
       INTEGER H, Q, ENTER, BAS
       JA=JJK+1
       KK=K-1
       J4=JJK+4
C
C   INSURE RETAINING VALUE OF MARKQ = 1 IF DUAL SIMPLEXING TO FEASIBILITY
C
       IF (IPARAM.EQ.1) GO TO 5
       MARKQ=2
     5 IF (K.EQ.1) GO TO 40
       II=4
       JJF=J4
       GO TO 41
    40 II=3
       JJF=JJK+3
C
C   ZERO OUT 'A' CONSTRAINT MATRIX
C
```

```
   41 DO 12 I=1,II
      DO 15 J=1,JJF
      A(K,I,J)=0.0
   15 CONTINUE
   12 CONTINUE
C
C  UPDATE VALUES IN THE CONSTRAINT MATRIX
C  BY PREMULTIPLYING INITIAL CONSTRAINT MATRIX BY BASIS INVERSE
C
      DO 42 I=1,II
      DO 44 J=1,JJF
      DO 46 H=1,II
      A(K,I,J)=BINV(K,I,H)*A1(K,H,J)+A(K,I,J)
   46 CONTINUE
   44 CONTINUE
   42 CONTINUE
C
C  ZERO OUT 'RHS' COLUMN
C
      DO 14 I=1,II
      RHS(K,L,I)=0.0
   14 CONTINUE
C
C  UPDATE OF RHS VALUES BY PREMULTIPLYING WITH BASIS INVERSE.
C
      DO 50 I=1,II
      DO 52 J=1,II
      RHS(K,L,I)=RHS(K,L,I)+BINV(K,I,J)*RHS1(K,1,J)
   52 CONTINUE
   50 CONTINUE
C
C  UPDATE OF COST COEFFICIENTS USING BASIC COST VECTOR AND COLUMNS OF
```

```
C  UPDATED SIMPLEX TABLEAU.
C
      DO 10 J=1,JJF
   10 ZJCJ(J)=0.0
      DO 54 J=1,JJF
      DO 56 I=1,II
      ZJCJ(J)=COSTBS(I)*A(K,I,J)+ZJCJ(J)
      IF (I.EQ.II) GO TO 55
      GO TO 56
   55 ZJCJ(J)=COST(K,J)+ZJCJ(J)
      GO TO 54
   56 CONTINUE
   54 CONTINUE
C
C  RETURN TO CALL STATEMENT IF NO NEED FOR UPDATING RANGE INFORMATION.
C
      IF (IPARAM.EQ.1) GO TO 98
      GO TO 97
   98 RETURN 1
C
C  UPDATE OF RETURNS IN TERMS OF FIXED AND VARIABLE COMPONENTS
C
   97 RFIX(K,L)=0.0
      RVAR(K,L)=0.0
      DO 58 I=1,II
      RFIX(K,L)=COSTBS(I)*RHS(K,L,I)+RFIX(K,L)
      RVAR(K,L)=COSTBS(I)*A(K,I,JA)+RVAR(K,L)
   58 CONTINUE
      IF (K.EQ.1) GO TO 59
      RFIX(K,L)=RFIX(K,L)+RFIX(KK,Q)-RVAR(K,L)*RHS1(K,1,1)+M(K,L)*FC(K)
      GO TO 80
   59 RFIX(K,L)=RFIX(K,L)-RVAR(K,L)*RHS1(K,1,1)+M(K,L)*FC(K)
```

```
   80 CONTINUE
C
C  UPDATE OF RANGES AND TRANSITION VALUES FOR REVERSING THROUGH THE SYSTEM
C
C  RECALCULATE NEG TO FACILITATE UPDATE OF RANGES FOR CURRENT 'L' .
C
      DO 85 I=1,II
      IF (A(K,I,JA).LT.-1.E-04) GO TO 86
      GO TO 85
   86 NEG=I
      GO TO 88
   85 CONTINUE
   88 CONTINUE
      DO 20 I=1,3
      TIMEF(K,L,I)=RHS(K,L,I)-(A(K,I,JA)*RHS1(K,1,1))
      TIMEV(K,L,I)=A(K,I,JA)
   20 CONTINUE
      IVAR1=BAS(K,L,1)
      IVAR2=BAS(K,L,2)
      IVAR3=BAS(K,L,3)
      OPRATE(K,L,1)=RATE(K,IVAR1)
      OPRATE(K,L,2)=RATE(K,IVAR2)
      OPRATE(K,L,3)=RATE(K,IVAR3)
      IF (L.EQ.1) RNGUP(K,L)=RHS1(K,1,1)
      IF (L.EQ.1) GO TO 60
      LL=L-1
      RNGLO(K,L)=RNGUP(K,LL)
      RNGUP(K,L)=RHS(K,L,NEG)/ABS(A(K,NEG,JA))+RHS1(K,1,1)
   60 IF (K.EQ.1) GO TO 67
      IF ((K.GT.1).AND.(L.EQ.1)) GO TO 64
      TRANSF(K,L)=RHS(K,L,II)-(A(K,II,JA)*RHS1(K,1,1))
      TRANSV(K,L)=A(K,II,JA)
```

```
      GO TO 70
   64 TRANSF(K,1)=0
      TRANSV(K,1)=A(K,II,JA)
C
C  SPLIT OF RANGES TO ACCOUNT FOR CHANGE IN RETURNS COST FROM PREVIOUS STAGE
C
   70 RNGADJ=(RNGUP(KK,Q)-TRANSF(K,L))/TRANSV(K,L)
      IF (RNGUP(K,L).GT.RNGADJ) GO TO 65
      GO TO 67
   65 RNGUP(K,L)=RNGADJ
      MARKQ=1
      Q=Q+1
      COST(K,J4)=RVAR(KK,Q)
      COSTBS(4)=ABS(COST(K,J4))
   67 IF(K.EQ.1) TRANSV(K,L)=RATE(K,KEEP)/RRATE(K,KEEP)
      RETURN
      END
```

```
C
C  SUBROUTINE TO CHECK IF IT IS ECONOMICALLY FEASIBLE TO ADD A MACHINE.
C
      SUBROUTINE MACHCK (NEG, ENTER, KEEP, II, IPARAM)
      DIMENSION A1(10,4,30), BAS(10,90,4), RHS(10,90,4), COST(10,30),
     1B(10,4,8), BINV(10,4,8),RHS1(10,1,4), RFIX(10,90), RVAR(10,90), RN
     2GUP(10,90), RNGLO(10,90), M(10,90), FC(10), A(10,4,30), ZJCJ(30),
     3RRATE(10,30), RATIO(30), THETA(30), RATE(10,30), TRANSF(10,90),
     4COSTBS(4), TRANSV(10,90), JJ(10), RNGRED(10), PERDEF(10,30),
     5OPRATE(10,90,3), TIMEF(10,90,3), TIMEV(10,90,3), LCOUNT(10)
      DIMENSION HTIMEF(10,90,3), HTIMEV(10,90,3), HOPR(10,90,3)
      DIMENSION BCK(10,4,4), BCKINV(10,4,4), BASKP(10,90,4)
      COMMON K, L, Q, A1, A, BAS, B, BINV, RHS1, RHS, COST, RFIX, RVAR,
     1RNGUP, RNGLO, M, FC, ZJCJ, RATE, RRATE, TRANSF, TRANSV, COSTBS,
     2JJK, MARKQ, OPRATE, TIMEF, TIMEV
      INTEGER ENTER, ENTHLD, HOLD, Q, BAS, BASHD1,BASHD2,BASHD3,BASHD4
      DATA HTIMEF/2700*0.0/, HTIMEV/2700*0.0/, HOPR/2700*0.0/
      DATA BCK/160*0./, BCKINV/160*0./, BASKP/3600*0.0/
      J4=JJK+4
      KK=K-1
C
C  HOLD CURRENT RHS1 IN CASE NO MACHINE IS ADDED
C
      RHS1H1=RHS1(K,1,1)
      RHS1H2=RHS1(K,1,2)
      RHS1H3=RHS1(K,1,3)
      RHS1H4=RHS1(K,1,4)
C
C  ADD MACHINE HOURS TO LAST TABLEAU AND UPDATE
C  REVISING RHS TO INCORPORATE MORE HOURS AND CHANGED BASELINE PRODUCTION
C
      RFIXCK=RFIX(K,L)
```

```
      RVARCK=RVAR(K,L)
      RHS1(K,1,1)=RNGLO(K,L)
      RHS1(K,1,2)=RHS1(K,1,2)+8.0
      RHS1(K,1,3)=RHS1(K,1,3)+8.0
      RHS1(K,1,4)=0
C
C   HOLD PRODUCTION AND TRANSITION DATA FOR LOWER PORTION OF RANGE SPLIT
C
      DO 20 I=1,3
      HTIMEF(K,L,I)=TIMEF(K,L,I)
      HTIMEV(K,L,I)=TIMEV(K,L,I)
   20 CONTINUE
      HTRNSF=TRANSF(K,L)
      HTRNSV=TRANSV(K,L)
      DO 30 I=1,3
   30 HOPR(K,L,I)=OPRATE(K,L,I)
C
C   HOLDING CURRENT BASIS AND INVERSE IN CASE NO MACHINE IS ADDED
C
      NEGHLD=NEG
      ENTHLD=ENTER
      HOLD=L
      KEHOLD=KEEP
      DO 62 I=1,II
      DO 64 J=1,II
      BCK(K,I,J)=B(K,I,J)
      BCKINV(K,I,J)=BINV(K,I,J)
   64 CONTINUE
   62 CONTINUE
C
C   HOLD CURRENT BASIC VARIABLES
C
```

```
      BASHD1=BAS(K,L,1)
      BASHD2=BAS(K,L,2)
      BASHD3=BAS(K,L,3)
      BASHD4=BAS(K,L,4)
C
C  RETURN COST VALUES TO THOSE BEFORE LAST RANGE SPLIT IN CASE NO MACH ADDED
C
      QQ=Q-1
      IF (MARKQ.EQ.1) COST(K,J4)=RVAR(KK,QQ)
      COSTBS(4)=ABS(COST(K,J4))
C
C  HOLD COST VALUES FOR CURRENT BASIC VARIABLES
C
      BCHLD1=COSTBS(1)
      BCHLD2=COSTBS(2)
      BCHLD3=COSTBS(3)
      BCHLD4=COSTBS(4)
C
C  HOLDING CURRENT RANGES IN CASE NO MACHINE IS ADDED
C
      RNGCKL=RNGLO(K,L)
      RNGCKU=RNGUP(K,L)
      IPARAM=1
C
C  DRIVE SOLUTION TO FEASIBILITY AFTER A MACHINE IS ADDED
C
      CALL UPDATE (NEG, ENTER, KEEP, IPARAM, &66)
   66 CALL RHSCK (NEG, ENTER, II, KEEP)
      CALL INV (II,&13)
      IPARAM=1
      CALL UPDATE (NEG, ENTER, KEEP, IPARAM, &99)
   99 CONTINUE
```

```
C
C   CHECK TO INSURE THAT THERE IS INFEASIBILITY IN RHS
C
       DO 67 I=1,II
       IF (RHS(K,L,I).LT.-1.E-03) GO TO 66
   67 CONTINUE
C
C   UPDATE ROUTINE TO REINSTATE ALL PARAMETER VALUES FOR CALCULATIONS
C
       IPARAM=2
       IF (MARKQ.EQ.1) Q=Q-1
       IF (K.GT.1) COST(K,J4)=RVAR(KK,Q)
       COSTBS(4)=ABS(COST(K,J4))
       CALL UPDATE (NEG, ENTER, KEEP, IPARAM, &99)
       IF ((RNGUP(K,L).GT.RNGCKL).AND.(RNGLO(K,L).LT.RNGCKU)) GO TO 68
       GO TO 70
C
C   DETERMINE IF THE EQUATION BALANCE INDICATES A RANGE MATCH
C
   68 DIFF=ABS(RVARCK-RVAR(K,L))
       IF (DIFF.LT.1.E-03) GO TO 70
       BALPT=(RFIX(K,L)+FC(K)-RFIXCK)/(RVARCK-RVAR(K,L))
       IF ((BALPT.GT.RNGLO(K,L)).AND.(BALPT.GT.RNGCKL).AND.(BALPT.LT.RNGU
      1P(K,L)).AND.(BALPT.LT.RNGCKU)) GO TO 69
       GO TO 70
C
C   ESTABLISHING A RANGE SPLIT DUE TO ADDING A MACHINE WHILE MAINTAINING
C   THE CURRENT BASIC SOLUTION.
C
C
C   REESTABLISH BASE SOLUTION FOR LOWER PORTION OF RANGE DUE TO SPLIT
C
```

```
69 RNGUP(K,L)=BALPT
   RNGLO(K,L)=RNGCKL
   DO 40 I=1,3
   TIMEF(K,L,I)=HTIMEF(K,L,I)
   TIMEV(K,L,I)=HTIMEV(K,L,I)
40 CONTINUE
   DO 45 I=1,II
45 BASKP(K,L,I)=BAS(K,L,I)
   TRANSF(K,L)=HTRNSF
   TRANSV(K,L)=HTRNSV
   RFIX(K,L)=RFIXCK
   RVAR(K,L)=RVARCK
   BAS(K,L,1)=BASHD1
   BAS(K,L,2)=BASHD2
   BAS(K,L,3)=BASHD3
   BAS(K,L,4)=BASHD4
   DO 50 I=1,3
50 OPRATE(K,L,I)=HOPR(K,L,I)
   L=L+1
   LML=L-1
   DO 80 I=1,II
80 BAS(K,L,I)=BASKP(K,LML,I)
   IPARAM=2
   M(K,L)=M(K,LML)+1
   IF (MARKQ.EQ.1) Q=Q-1
   IF (K.GT.1) COST(K,J4)=RVAR(KK,Q)
   COSTBS(4)=ABS(COST(K,J4))
   CALL UPDATE (NEG, ENTER, KEEP, IPARAM, &99)
   RNGLO(K,L)=BALPT
   GO TO 75
C
C  REESTABLISH VALUES WHICH EXISTED FOR BASIC SOLUTION WHEN MACHCK CALL
```

```
C   SINCE NO MACHINE COULD BE FEASIBLY ADDED
C
   70 DO 72 I=1,II
      DO 74 J=1,II
      B(K,I,J)=BCK(K,I,J)
      BINV(K,I,J)=BCKINV(K,I,J)
   74 CONTINUE
   72 CONTINUE
      RHS1(K,1,1)=RHS1H1
      RHS1(K,1,2)=RHS1H2
      RHS1(K,1,3)=RHS1H3
      RHS1(K,1,4)=RHS1H4
      BAS(K,L,1)=BASHD1
      BAS(K,L,2)=BASHD2
      BAS(K,L,3)=BASHD3
      BAS(K,L,4)=BASHD4
      COSTBS(1)=BCHLD1
      COSTBS(2)=BCHLD2
      COSTBS(3)=BCHLD3
      COSTBS(4)=BCHLD4
      KEEP=KEHOLD
      NEG=NEGHLD
      ENTER=ENTHLD
      L=HOLD
      IPARAM=2
      IF (MARKQ.EQ.1) Q=Q-1
      IF (K.GT.1) COST(K,J4)=RVAR(KK,Q)
      COSTBS(4)=ABS(COST(K,J4))
      CALL UPDATE (NEG, ENTER, KEEP, IPARAM, &99)
      GO TO 75
   13 STOP
   75 RETURN
      END
```

```
C
C   SUBROUTINE TO ADD A MACHINE WHEN NO INCREASE IN PRODUCTION CAN BE
C   ATTAINED THROUGH RATE INCREASES.
C
      SUBROUTINE MACHAD (NEG, ENTER, II, KEEP, IPARAM)
      DIMENSION A1(10,4,30), BAS(10,90,4), RHS(10,90,4), COST(10,30),
     1B(10,4,8), BINV(10,4,8),RHS1(10,1,4), RFIX(10,90), RVAR(10,90), RN
     2GUP(10,90), RNGLO(10,90), M(10,90), FC(10), A(10,4,30), ZJCJ(30),
     3RRATE(10,30), RATIO(30), THETA(30), RATE(10,30), TRANSF(10,90),
     4COSTBS(4), TRANSV(10,90), JJ(10), RNGRED(10), PERDEF(10,30),
     5OPRATE(10,90,3), TIMEF(10,90,3), TIMEV(10,90,3), LCOUNT(10)
      COMMON K, L, Q, A1, A, BAS, B, BINV, RHS1, RHS, COST, RFIX, RVAR,
     1RNGUP, RNGLO, M, FC, ZJCJ, RATE, RRATE, TRANSF, TRANSV, COSTBS,
     2JJK, MARKQ, OPRATE, TIMEF, TIMEV
      INTEGER ENTER, Q
      LL=L-1
C
C   INCREASE INITIAL RHS TO REFLECT ADDING A MACHINE AND RANGE LOWER
C   BOUND CORRESPONDING TO PREVIOUS RANGE UPPER BOUND.
C
      RHS1(K,1,1)=RNGUP(K,LL)
      RHS1(K,1,2)=RHS1(K,1,2)+8.0
      RHS1(K,1,3)=RHS1(K,1,3)+8.0
      RHS1(K,1,4)=0.0
      IPARAM=1
C
C   DRIVE SOLUTION TO FEASIBILITY AFTER ADDING A MACHINE
C
      CALL UPDATE (NEG, ENTER, KEEP, IPARAM, &80)
   80 CALL RHSCK (NEG, ENTER, II, KEEP)
      CALL INV (II,&13)
      IPARAM=1
```

```
      CALL UPDATE (NEG, ENTER, KEEP, IPARAM, &99)
   99 CONTINUE
      DO 82 I=1,II
      IF (RHS(K,L,I).LT.-1.E-03) GO TO 80
   82 CONTINUE
C
C  ESTABLISH COMPLETE TABLEAU AND RANGE INFORMATION FOR FEASIBLE
C  SOLUTION WITH MACHINE ADDED.
C
      IPARAM=2
      CALL UPDATE (NEG, ENTER, KEEP, IPARAM, &99)
      GO TO 85
   13 STOP
   85 RETURN
      END
```

APPENDIX C

OPERATIONAL DATA RESULTS FOR MACHINE REQUIREMENTS PLANNING MODEL USING 4 STAGE EXAMPLE

NOTATION FOR THE INTERPRETATION OF HEADINGS FOR RESULTS

STAGE	- Number of the machining center of the system
RNGLO	- Lower Bound of the production range for the current solution
RNGUP	- Upper bound of the production range
MACH	- Number of machines at the machining center
TRANSF	- Fixed value of the transition function which specifies the input to the current stage and the output from the previous stage
TRANSV	- Variable value of the transition function which is multiplied by the production output required
FIXRTN	- Fixed portion of the cost function for producing a specified output through the current stage
VARRTN	- Variable portion of the cost function which must be multiplied by the production output required
BASVRBL	- Index number of the basic variable which is analogous to the rate of operation. The variables are indexed through twice the number of operating rates to account for both first and second shifts. Second shift rates are indicated by index numbers which begin one greater than the number of operating rates for the machine.
OPRRATE	- The operating rate of the machine
FIXTIME	- Fixed portion of the amount of time in hours which the machine should be operated at the specified rate
VARTIME	- Variable amount of time for which the machine is to be operated. This value must be multiplied by the output needed from the stage and added to FIXTIME to obtain the total time of operation. Total units processed at the stage can also be obtained by multiplying the time by the operating rate and summing for each stage.

STAGE	RNGLO	RNGUP	MACH	TRANSF	TRANSV	FIXRTN	VARRTN	BASVRUL	OPRRATE	FIXTIME	VARTIME
1	0.00	191.80	1	0.00	1.0428	80.00	-0.4171				
								1	25.00	-0.00	0.0417
								14	0.00	8.00	-0.0417
								15	0.00	8.00	0.0000
1	191.80	229.92	1	0.00	1.0428	79.50	0.4197				
								1	25.00	48.25	-0.2099
								2	30.00	-40.25	0.2099
								15	0.00	8.00	0.0000
1	229.92	267.40	1	0.00	1.0428	77.85	0.4269				
								3	35.00	-49.08	0.2134
								2	30.00	57.08	-0.2134
								15	0.00	8.00	0.0000
1	267.40	497.32	1	0.00	1.0428	63.60	0.4802				
								3	35.00	8.00	0.0000
								8	30.00	-9.30	0.0348
								15	0.00	17.30	-0.0348
1	497.32	534.80	1	0.00	1.0428	58.25	0.4909				
								3	35.00	8.00	-0.0000
								8	30.00	114.15	-0.2134
								9	35.00	-106.15	0.2134
1	534.80	609.04	1	0.00	1.0428	-13.45	0.6250				
								3	35.00	65.63	-0.1078
								5	45.00	-57.63	0.1078
								9	35.00	8.00	0.0000

STAGE	RNGLO	RNGUP	MACH	TRANSF	TRANSV	FIXRTN	VARRTN	UASVRBL	OPRRATE	FIXTIME	VARTIME
1	609.04	645.64	1	0.00	1.0428	-45.48	0.6776				
								10	40.00	-133.12	0.2186
								5	45.00	8.00	0.0000
								9	35.00	141.12	-0.2186
1	645.64	683.28	1	0.00	1.0428	-122.60	0.7970				
								10	40.00	145.23	-0.2125
								5	45.00	8.00	0.0000
								11	45.00	-137.23	0.2125
1	683.28	719.64	1	0.00	1.0428	-359.75	1.1441				
								6	50.00	-150.34	0.2200
								5	45.00	158.34	-0.2200
								11	45.00	8.00	0.0000
1	719.64	728.19	1	0.00	1.0428	47.21	0.4802				
								3	35.00	16.00	0.0000
								8	30.00	-18.61	0.0348
								15	0.00	34.61	-0.0348
1	728.19	994.64	2	0.00	1.0428	127.21	0.4802				
								3	35.00	16.00	0.0000
								8	30.00	-18.61	0.0348
								15	0.00	34.61	-0.0348
1	994.64	1069.60	2	0.00	1.0428	116.50	0.4909				
								3	35.00	16.00	-0.0000
								8	30.00	228.30	-0.2134
								9	35.00	-212.30	0.2134

STAGE	ANGLD	ANGUP	MACH	TRANSF	TRANSV	FIXRTN	VARRTN	BASVRBL	OPRRATE	FIXTIME	VARTIME
2	0.00	171.66	1	0.00	1.1173	60.00	-0.4804				
								4	25.00	-0.00	0.0467
								12	0.00	8.00	-0.0447
								13	0.00	8.00	0.0000
2	171.66	179.00	1	-0.00	1.1173	139.50	0.9494				
								4	25.00	-0.00	0.0447
								12	0.00	8.00	-0.0447
								13	0.00	8.00	0.0000
2	179.00	204.88	1	-6.94	1.1561	121.19	1.0517				
								4	25.00	49.39	-0.2312
								5	30.00	-41.39	0.2312
								13	0.00	8.00	0.0000
2	204.88	213.60	1	-6.94	1.1561	119.49	1.0600				
								4	25.00	49.39	-0.2312
								5	30.00	-41.39	0.2312
								13	0.00	8.00	0.0000
2	213.60	238.12	1	1.34	1.1173	110.37	1.1027				
								9	25.00	-9.55	0.0447
								5	30.00	8.00	0.0000
								13	0.00	17.55	-0.0447
2	238.12	392.60	1	1.34	1.1173	96.20	1.1622				
								9	25.00	-9.55	0.0447
								5	30.00	8.00	0.0000
								13	0.00	17.55	-0.0447

STAGE	RNGLO	RNGUP	MACH	TRANSF	TRANSV	FIXRTN	VARRTN	BASVRBL	OPRRATE	FIXTIME	VARTIME
2	392.60	427.20	1	-13.87	1.1561	25.91	1.3412				
								9	25.00	98.77	-0.2312
								5	30.00	8.00	0.0000
								10	30.00	-90.77	0.2312
2	427.20	442.70	2	2.68	1.1173	128.79	1.1622				
								9	25.00	-19.09	0.0447
								5	30.00	16.00	0.0000
								13	0.00	35.09	-0.0447
2	442.70	476.24	2	2.68	1.1173	123.47	1.1742				
								9	25.00	-19.09	0.0447
								5	30.00	16.00	0.0000
								13	0.00	35.09	-0.0447
2	476.24	542.69	2	2.68	1.1173	52.13	1.3240				
								9	25.00	-19.09	0.0447
								5	30.00	16.00	0.0000
								13	0.00	35.09	-0.0447
2	542.69	575.45	2	2.68	1.1173	20.24	1.3828				
								9	25.00	-19.09	0.0447
								5	30.00	16.00	0.0000
								13	0.00	35.09	-0.0447
2	575.45	609.13	2	2.68	1.1173	-56.56	1.5162				
								9	25.00	-19.09	0.0447
								5	30.00	16.00	0.0000
								13	0.00	35.09	-0.0447

2	609.13	641.68	2	2.68	1.1173	-292.78	1.9040				
								9	25.00	-19.09	0.0447
								5	30.00	16.00	0.0000
								13	0.00	35.09	-0.0447
STAGE	RNGLO	RNGUP	MACH	TRANSF	TRANSV	FIXRTN	VARRTN	BASVRBL	OPRRATE	FIXTIME	VARTIME
2	641.68	649.33	2	2.68	1.1173	112.40	1.1622				
								9	25.00	-19.09	0.0447
								5	30.00	16.00	0.0000
								13	0.00	35.09	-0.0447
2	649.33	785.20	2	2.68	1.1173	192.40	1.1622				
								9	25.00	-19.09	0.0447
								5	30.00	16.00	0.0000
								13	0.00	35.09	-0.0447
2	785.20	854.40	2	-27.75	1.1561	51.82	1.3412				
								9	25.00	197.55	-0.2312
								5	30.00	16.00	0.0000
								10	30.00	-181.55	0.2312
2	854.40	886.60	3	4.02	1.1173	224.99	1.1622				
								9	25.00	-28.64	0.0447
								5	30.00	24.00	0.0000
								13	0.00	52.64	-0.0447
2	886.60	953.69	3	4.02	1.1173	214.33	1.1742				
								9	25.00	-28.64	0.0447
								5	30.00	24.00	0.0000
								13	0.00	52.64	-0.0447

STAGE	RNGLO	RNGUP	MACH	TRANSF	TRANSV	FIXRTN	VARRTN	BASVRBL	OPRRATE	FIXTIME	VARTIME
3	0.00	163.94	1	0.00	1.0471	40.00	-0.1361				
								1	40.00	-0.00	0.0262
								12	0.00	8.00	-0.0262
								13	0.00	8.00	0.0000
3	163.94	170.94	1	-0.00	1.0471	179.50	1.1303				
								1	40.00	-0.00	0.0262
								12	0.00	8.00	-0.0262
								13	0.00	8.00	0.0000
3	170.94	195.66	1	-0.00	1.0471	161.19	1.2374				
								1	40.00	-0.00	0.0262
								12	0.00	8.00	-0.0262
								13	0.00	8.00	0.0000
3	195.66	203.99	1	-0.00	1.0471	159.49	1.2461				
								1	40.00	-0.00	0.0262
								12	0.00	8.00	-0.0262
								13	0.00	8.00	0.0000
3	203.99	227.41	1	-0.00	1.0471	150.37	1.2908				
								1	40.00	-0.00	0.0262
								12	0.00	8.00	-0.0262
								13	0.00	8.00	0.0000
3	227.41	305.60	1	-0.00	1.0471	136.20	1.3531				
								1	40.00	-0.00	0.0262
								12	0.00	8.00	-0.0262
								13	0.00	8.00	0.0000

STAGE	RNGLO	RNGUP	MACH	TRANSF	TRANSV	FIXRTN	VARRTN	BASVRBL	OPRRATE	FIXTIME	VARTIME
3	305.60	343.08	1	-6.15	1.0672	128.25	1.3791				
								1	40.00	73.23	-0.2134
								2	45.00	-65.23	0.2134
								13	0.00	8.00	0.0000
3	343.08	374.15	1	-0.00	1.0493	121.80	1.3979				
								7	45.00	-8.00	0.0233
								2	45.00	8.00	0.0000
								13	0.00	16.00	-0.0233
3	374.15	407.12	1	-0.00	1.0493	51.51	1.5858				
								7	45.00	-8.00	0.0233
								2	45.00	8.00	0.0000
								13	0.00	16.00	-0.0233
3	407.12	421.89	1	-0.00	1.0493	154.39	1.3979				
								7	45.00	-8.00	0.0233
								2	45.00	8.00	0.0000
								13	0.00	16.00	-0.0233
3	421.89	453.86	1	-0.00	1.0493	149.07	1.4105				
								7	45.00	-8.00	0.0233
								2	45.00	8.00	0.0000
								13	0.00	16.00	-0.0233
3	453.86	517.18	1	-0.00	1.0493	77.73	1.5677				
								7	45.00	-8.00	0.0233
								2	45.00	8.00	0.0000
								13	0.00	16.00	-0.0233

STAGE	RNGLO	RNGUP	MACH	TRANSF	TRANSV	FIXRTN	VARRTN	BASVRBL	OPRRATE	FIXTIME	VARTIME
3	517.18	548.40	1	-0.00	1.0493	45.84	1.6294				
								7	45.00	-8.00	0.0233
								2	45.00	8.00	0.0000
								13	0.00	16.00	-0.0233
3	548.40	580.51	1	-0.00	1.0493	-30.96	1.7694				
								7	45.00	-8.00	0.0233
								2	45.00	8.00	0.0000
								13	0.00	16.00	-0.0233
3	580.51	611.52	1	-0.00	1.0493	-267.18	2.1763				
								7	45.00	-8.00	0.0233
								2	45.00	8.00	0.0000
								13	0.00	16.00	-0.0233
3	611.52	618.81	1	-0.00	1.0493	138.00	1.3979				
								7	45.00	-8.00	0.0233
								2	45.00	8.00	0.0000
								13	0.00	16.00	-0.0233
3	618.81	686.16	1	-0.00	1.0493	218.00	1.3979				
								7	45.00	-8.00	0.0233
								2	45.00	8.00	0.0000
								13	0.00	16.00	-0.0233
3	686.16	746.73	1	-18.60	1.0764	75.04	1.6062				
								7	45.00	8.00	0.0000
								2	45.00	57.24	-0.0718
								5	60.00	-49.24	0.0718

STAGE	RNGLO	RNGUP	MACH	TRANSF	TRANSV	FIXRIN	VARRIN	BASVRBL	OPRRATE	FIXTIME	VARTIME
3	766.73	797.64	1	-18.60	1.0764	-68.87	1.7990				
								7	45.00	8.00	0.0000
								2	45.00	57.24	-0.0718
								5	60.00	-49.24	0.0718
3	797.64	806.18	1	-0.00	1.0493	63.02	1.5858				
								7	45.00	-16.00	0.0233
								2	45.00	16.00	0.0000
								13	0.00	32.00	-0.0233
4	0.00	112.44	1	0.00	1.0672	70.00	-1.4194				
								3	15.00	-0.00	0.0711
								12	0.00	8.00	-0.0711
								13	0.00	8.00	0.0000
4	112.44	153.43	1	-0.51	1.0718	58.63	2.0216				
								3	15.00	8.00	0.0000
								9	20.00	-6.03	0.0536
								13	0.00	14.03	-0.0536
4	153.43	159.97	1	-0.51	1.0718	197.61	3.0871				
								3	15.00	8.00	0.0000
								9	20.00	-6.03	0.0536
								13	0.00	14.03	-0.0536
4	159.97	183.03	1	-0.51	1.0718	179.25	3.2019				
								3	15.00	8.00	0.0000
								9	20.00	-6.03	0.0536
								13	0.00	14.03	-0.0536

STAGE	RNGLO	RNGUP	MACH	TRANSF	TRANSV	FIXRIN	VARRIN	BASVRBL	OPRRATE	FIXTIME	VARTIME
3	604.18	814.24	2	-0.00	1.0493	103.02	1.5858				
								7	45.00	-16.00	0.0233
								2	45.00	16.00	0.0000
								13	0.00	32.00	-0.0233
3	814.24	844.93	2	-0.00	1.0493	276.19	1.3979				
								7	45.00	-16.00	0.0233
								2	45.00	16.00	0.0000
								13	0.00	32.00	-0.0233
4	183.03	190.80	1	-0.51	1.0718	177.55	3.2112				
								3	15.00	8.00	0.0000
								9	20.00	-6.03	0.0536
								13	0.00	14.03	-0.0536
4	190.80	212.65	1	-0.51	1.0718	168.41	3.2591				
								3	15.00	8.00	0.0000
								9	20.00	-6.03	0.0536
								13	0.00	14.03	-0.0536
4	212.65	261.72	1	-0.51	1.0718	154.20	3.3259				
								3	15.00	8.00	0.0000
								9	20.00	-6.03	0.0536
								13	0.00	14.03	-0.0536
4	261.72	285.30	1	-4.17	1.0858	68.97	3.6516				
								3	15.00	64.83	-0.2172
								9	20.00	8.00	0.0000
								4	20.00	-56.83	0.2172

STAGE	RNGLO	RNGUP	MACH	TRANSF	TRANSV	FIXKIN	VARKIN	BASVKBL	UPRRATE	FIXTIME	VARTIME
4	285.30	298.56	1	-4.17	1.0858	60.92	3.6798				
								3	15.00	64.83	-0.2172
								9	20.00	8.00	0.0000
								4	20.00	-56.83	0.2172
4	298.56	314.83	1	-1.03	1.0718	94.23	3.3538				
								3	15.00	16.00	0.0000
								9	20.00	-12.05	0.0536
								13	0.00	28.05	-0.0536
4	314.83	321.05	2	-1.03	1.0718	164.23	3.3538				
								3	15.00	16.00	0.0000
								9	20.00	-12.05	0.0536
								13	0.00	28.05	-0.0536
4	321.05	350.04	2	-1.03	1.0718	157.76	3.3740				
								3	15.00	16.00	0.0000
								9	20.00	-12.05	0.0536
								13	0.00	28.05	-0.0536
4	350.04	380.80	2	-1.03	1.0718	87.28	3.5753				
								3	15.00	16.00	0.0000
								9	20.00	-12.05	0.0536
								13	0.00	28.05	-0.0536
4	380.80	394.59	2	-1.03	1.0718	190.35	3.3740				
								3	15.00	16.00	0.0000
								9	20.00	-12.05	0.0536
								13	0.00	28.05	-0.0536
4	394.59	424.41	2	-1.03	1.0718	185.01	3.3875				
								3	15.00	16.00	0.0000
								9	20.00	-12.05	0.0536
								13	0.00	28.05	-0.0536

STAGE	RNGLO	RNGUP	MACH	TRANSF	TRANSV	FIXRTN	VARRTN	BASVRBL	OPRRATE	FIXTIME	VARTIME
4	424.41	483.49	2	-1.03	1.0718	113.51	3.5560				
								3	15.00	16.00	0.0000
								9	20.00	-12.05	0.0536
								13	0.00	28.05	-0.0536
4	483.49	512.62	2	-1.03	1.0718	81.56	3.6220				
								3	15.00	16.00	0.0000
								9	20.00	-12.05	0.0536
								13	0.00	28.05	-0.0536
4	512.62	523.44	2	-1.03	1.0718	4.62	3.7721				
								3	15.00	16.00	0.0000
								9	20.00	-12.05	0.0536
								13	0.00	28.05	-0.0536
4	523.44	542.33	2	-8.34	1.0858	-168.88	4.1036				
								3	15.00	129.67	-0.2172
								9	20.00	16.00	0.0000
								4	20.00	-113.67	0.2172
4	542.33	570.89	2	-8.34	1.0858	-408.50	4.5454				
								3	15.00	129.67	-0.2172
								9	20.00	16.00	0.0000
								4	20.00	-113.67	0.2172

STAGE	RNGLO	RNGUP	MACH	TRANSF	TRANSV	FIXRTN	VARRTN	HASVRBL	OPRRATE	FIXTIME	VARTIME
4	570.89	577.60	2	-8.34	1.0858	3.17	3.7002				
								3	15.00	129.67	-0.2172
								9	20.00	16.00	0.0000
								4	20.00	-113.67	0.2172
4	577.60	578.55	2	-1.54	1.0718	201.94	3.3740				
								3	15.00	24.00	0.0000
								9	20.00	-18.08	0.0536
								13	0.00	42.08	-0.0536
4	578.55	641.63	3	-1.54	1.0718	271.94	3.3740				
								3	15.00	24.00	0.0000
								9	20.00	-18.08	0.0536
								13	0.00	42.08	-0.0536
4	641.63	698.14	3	-1.54	1.0718	128.66	3.5973				
								3	15.00	24.00	0.0000
								9	20.00	-18.08	0.0536
								13	0.00	42.08	-0.0536
4	698.14	745.64	3	-1.54	1.0718	-15.55	3.8038				
								3	15.00	24.00	0.0000
								9	20.00	-18.08	0.0536
								13	0.00	42.08	-0.0536
4	745.64	753.61	3	-1.54	1.0718	116.67	3.5753				
								3	15.00	24.00	0.0000
								9	20.00	-18.08	0.0536
								13	0.00	42.08	-0.0536

VITA

Glenn M. Hayes, Jr. was born May 23, 1945 at Kinston, North Carolina. He graduated from Radford High School, Honolulu, Hawaii, in 1963 and attended the University of Virginia where he was graduated in 1967 with a Bachelor of Engineering Degree in Mechanical Engineering. Following graduation, he was employed by the Procter and Gamble Manufacturing Company as a Production Supervisor in Baltimore, Maryland.

In 1969, Mr. Hayes entered the United States Army and was commissioned as a Second Lieutenant in 1970. After serving in a variety of supply and maintenance management positions, both in the United States and overseas, CPT Hayes was assigned to Aberdeen Proving Ground, Maryland, in 1976 to further his military schooling. Following this assignment, he has been pursuing a Master of Science Degree in Industrial Engineering and Operations Research at Virginia Polytechnic Institute and State University.

Glenn M. Hayes, Jr.

A GENERALIZED MACHINE REQUIREMENTS PLANNING ALGORITHM FOR SERIAL FLOW MACHINING SYSTEMS

by

Glenn M. Hayes, Jr.

(ABSTRACT)

The machine requirements planning problem is one which is applicable throughout the manufacturing industry. The use of automation and computer technology makes the use of machine requirements modelling particularly attractive, and the vast capital investment in machining equipment offers significant opportunity for savings through optimization. The machine requirements problem must be modelled as a mixed integer linear program. However, a dynamic programming solution procedure with linear programming postoptimality techniques at each stage of the machining process is utilized to provide an efficient, flexible algorithm. The machine requirements model is capable of optimizing any type of machining system, whether of a discrete or continuous operating mode. Through discretizing the machine parameters problem, by limiting tool changes to between passes of a tool, any continuous mode system is approximated as discrete for optimization purposes and a direct interface with the machine requirements planning model is provided.

The dynamic programming solution procedure is compared with a mixed integer procedure. The DP formulation is not only more efficient in

both time and core but provides sensitivity information and offers a broad spectrum of further application into more complex aspects of manufacturing systems.