Mathematical modelling of flexible manufacturing systems.

Arvind Madhav. Padhye

University of Windsor

Follow this and additional works at: https://scholar.uwindsor.ca/etd

Recommended Citation
https://scholar.uwindsor.ca/etd/2677

This online database contains the full-text of PhD dissertations and Masters' theses of University of Windsor students from 1954 forward. These documents are made available for personal study and research purposes only, in accordance with the Canadian Copyright Act and the Creative Commons license—CC BY-NC-ND (Attribution, Non-Commercial, No Derivative Works). Under this license, works must always be attributed to the copyright holder (original author), cannot be used for any commercial purposes, and may not be altered. Any other use would require the permission of the copyright holder. Students may inquire about withdrawing their dissertation and/or thesis from this database. For additional inquiries, please contact the repository administrator via email (scholarship@uwindsor.ca) or by telephone at 519-253-3000ext. 3208.
NOTICE

The quality of this microfiche is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Previously copyrighted materials (journal articles, published tests, etc.) are not filmed.

Reproduction in full or in part of this film is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30.

THIS DISSERTATION HAS BEEN MICROFILMED EXACTLY AS RECEIVED

AVIS

La qualité de cette microfiche dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

Les documents qui font déjà l'objet d'un droit d'auteur (articles de revue, examens publiés, etc.) ne sont pas microfilmés.

La reproduction, même partielle, de ce microfilm est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30.

LA THÈSE A ÉTÉ MICROFILMÉE TELLE QUE NOUS L'AVONS RÊCUE
MATHEMATICAL MODELLING OF FLEXIBLE MANUFACTURING SYSTEMS

by

(C) Arvind Madhav Padhye

A Thesis
submitted to the
Faculty of Graduate Studies and Research
through the Department of
Industrial Engineering in Partial Fulfillment
of the requirements for the Degree
of Master of Applied Science at
the University of Windsor

Windsor, Ontario, Canada

1986
Permission has been granted to the National Library of Canada to microfilm this thesis and to lend or sell copies of the film.

The author (copyright owner) has reserved other publication rights, and neither the thesis nor extensive extracts from it may be printed or otherwise reproduced without his/her written permission.

I hereby declare that I am the sole author of this thesis. I authorize the University of Windsor to lend this thesis to other institutions or individuals for the purpose of scholarly research.

A. M. Padhye

I further authorize the University of Windsor to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

A. M. Padhye
The University of Winsor requires the signatures of all persons using or photocopying this thesis. Please sign below, and give address and date.
ABSTRACT

The subject of automated manufacturing has received increasing attention in the past few years, both from the research and the implementational view points. Mathematical modelling of the operation of automated manufacturing systems has lately been the focus of attention of several researchers in the field of applied Operations Research due to the need to effectively control the performance of such sophisticated systems. This research deals with the production planning problems of Flexible Manufacturing Systems. The mathematical programming formulations of the operation allocation and the part mix determination problems in FMS are proposed.

The operation allocation problem has been formulated as a 0-1 nonlinear integer program. The part mix determination problem is formulated as a general integer program. Alternate formulations of the objective functions have been proposed for both problems. The formulations consider the important and real life planning aspects of re-fixturing and tool availability. A realistic example is solved using the SAS/OR (Version 5) package and the computational behaviour of the formulations is discussed.
ACKNOWLEDGEMENTS

I would like to express my deep gratitude towards Dr. R. S. Lashkari and Dr. S. P. Dutta for their support and guidance in conducting this research. My special thanks are due to Tom Williams and Jacquie Mummery (Industrial Engg.), and to the computer consultants of the University of Windsor for their invaluable help.
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>LAYOUT OF THE FMS</td>
<td>8</td>
</tr>
<tr>
<td>2.</td>
<td>SAMPLE PART - GEAR BOX COVER</td>
<td>25</td>
</tr>
</tbody>
</table>
# List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Matrix of $F(i,k,p,q)$ for part type $i$, opn. $k$</td>
<td>43</td>
</tr>
<tr>
<td>2. Sample operation allocation for part type $i$</td>
<td>47</td>
</tr>
<tr>
<td>3. Allocation of operations of set $A_4$ to machines</td>
<td>52</td>
</tr>
<tr>
<td>4. Sample operation allocation to machine $j$</td>
<td>56</td>
</tr>
<tr>
<td>(Limited tool availability)</td>
<td></td>
</tr>
<tr>
<td>5. Summary of machine data</td>
<td>66</td>
</tr>
<tr>
<td>6. Summary of parts data</td>
<td>68</td>
</tr>
<tr>
<td>7. Possibilities of allocation of operations</td>
<td>68</td>
</tr>
<tr>
<td>8. Upper &amp; lower bounds on the number of parts</td>
<td>70</td>
</tr>
<tr>
<td>of each type in the system</td>
<td></td>
</tr>
<tr>
<td>9. Availability data for common fixture elements</td>
<td>70</td>
</tr>
<tr>
<td>10. Sizes of operation allocation formulations using</td>
<td></td>
</tr>
<tr>
<td>different linearization strategies (minimization of transport)</td>
<td>79</td>
</tr>
<tr>
<td>11. Sizes of formulations using linearization strategy 2</td>
<td>82</td>
</tr>
<tr>
<td>12. Optimal solutions of operation allocation problem</td>
<td>83</td>
</tr>
<tr>
<td>13. Optimal solution of operation allocation problem</td>
<td></td>
</tr>
<tr>
<td>(minimize transport) effect of location of machines</td>
<td>84</td>
</tr>
</tbody>
</table>
14. OPTIMAL SOLUTION OF OPERATION ALLOCATION PROBLEM
(MINIMIZE TRANSPORT) EFFECT OF TOOL MAGAZINE
CAPACITIES ........................................... 86
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>I.</td>
<td></td>
</tr>
<tr>
<td>II.</td>
<td></td>
</tr>
<tr>
<td>III.</td>
<td></td>
</tr>
<tr>
<td>IV.</td>
<td></td>
</tr>
<tr>
<td>V.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT</td>
<td>vi</td>
</tr>
<tr>
<td>ACKNOWLEDGEMENTS</td>
<td>vii</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>viii</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>ix</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>I.</td>
<td>INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>II.</td>
<td>FLEXIBLE MANUFACTURING SYSTEMS</td>
<td>4</td>
</tr>
<tr>
<td>II.</td>
<td>2.1 Components of FMS</td>
<td>4</td>
</tr>
<tr>
<td>II.</td>
<td>2.2 Configuration of FMS</td>
<td>9</td>
</tr>
<tr>
<td>II.</td>
<td>2.3 Mode of Operation of FMS</td>
<td>11</td>
</tr>
<tr>
<td>III.</td>
<td>LITERATURE SURVEY</td>
<td>14</td>
</tr>
<tr>
<td>IV.</td>
<td>PLANNING PROBLEMS OF FMS</td>
<td>22</td>
</tr>
<tr>
<td>IV.</td>
<td>4.1 Production Planning Problems of FMS</td>
<td>22</td>
</tr>
<tr>
<td>IV.</td>
<td>4.2 Operation Allocation in FMS</td>
<td>24</td>
</tr>
<tr>
<td>IV.</td>
<td>4.3 Part Mix Determination in FMS</td>
<td>28</td>
</tr>
<tr>
<td>IV.</td>
<td>4.4 Effects of Operation Allocation and Part Mix on FMS Operation</td>
<td>31</td>
</tr>
<tr>
<td>IV.</td>
<td>4.4.1 Transport activities</td>
<td>31</td>
</tr>
<tr>
<td>IV.</td>
<td>4.4.2 Cutting tools</td>
<td>33</td>
</tr>
<tr>
<td>IV.</td>
<td>4.4.3 Refixturing activities</td>
<td>35</td>
</tr>
<tr>
<td>IV.</td>
<td>4.4.4 Throughput and machine utilization</td>
<td>37</td>
</tr>
<tr>
<td>V.</td>
<td>MATHEMATICAL MODELLING</td>
<td>39</td>
</tr>
<tr>
<td>V.</td>
<td>5.1 Assumptions and Notation used</td>
<td>39</td>
</tr>
<tr>
<td>V.</td>
<td>5.2 Operation Allocation - Formulation of Objective Functions</td>
<td>42</td>
</tr>
<tr>
<td>V.</td>
<td>5.2.1 Minimization of refixturing activities</td>
<td>42</td>
</tr>
<tr>
<td>V.</td>
<td>5.2.2 Minimization of transport load</td>
<td>46</td>
</tr>
<tr>
<td>V.</td>
<td>5.3 Operation Allocation - Formulation of Constraints</td>
<td>50</td>
</tr>
<tr>
<td>V.</td>
<td>5.4 Part Mix Determination - Formulation of Objective Functions</td>
<td>58</td>
</tr>
</tbody>
</table>
5.4.1 Minimization of workload imbalances ............................................ 58
5.4.2 Minimization of refixturing activities ............................................. 59
5.4.3 Minimization of transport load ....................................................... 60
5.5 Part Mix Determination - Formulation of Constraints ............................. 60

VI. COMPUTATIONAL RESULTS ................................................................. 63

6.1 Description of the Problem ................................................................. 63
6.2 Linearization Strategies ........................................................................ 71
   6.2.1 Linearization strategy 1 ................................................................. 72
   6.2.2 Linearization strategy 2 ................................................................. 73
   6.2.3 Linearization strategy 3 ................................................................. 74
   6.2.4 Linearization strategy 4 ................................................................. 75
   6.2.5 Linearization strategy 5 ................................................................. 77
   6.2.6 Linearization of the proposed formulations ...................................... 78
6.3 Optimal Solutions .................................................................................. 80
6.4 Computational Experience ..................................................................... 87

VII. SUMMARY .......................................................................................... 98

REFERENCES .............................................................................................. 101

APPENDICES

A. PART SKETCHES, PROCESS SHEETS AND REFIXTURING INFORMATION ........ 104
B. CUTTING TOOLS INFORMATION ............................................................ 117
C. EQUATIONS OF FORMULATIONS ........................................................... 120
D. COMPUTER PROGRAM LISTING ............................................................. 126

VITA AUCTORIS ......................................................................................... 163
Chapter I

INTRODUCTION

The subject of automated manufacturing has received increasing attention in the past few years, both from the research and the implementational viewpoints, due mainly to the advances made in the computer control of machine tools. In fact, the progress made in the areas of numerically controlled machines, automated materials handling and computer communications, combined with the availability of increasingly cheaper computing power, has now made it feasible to fulfill the long felt need for greater flexibility in production systems. Implicit in the concept of flexibility is a much wider scope for improvement in manufacturing productivity and ultimately the profitability of operations. However, experience has shown that the progress made in the development of physical aspects of automated manufacturing systems has not necessarily led to corresponding gains in manufacturing productivity. It is being realised, both by the researchers and the practitioners, that the economic utility of such advanced manufacturing strategies depends not only on the physical sophistication but also, and equally, on the
effectiveness of the operating strategies of the systems. The latter is essentially a complex managerial decision problem and has lately been the focus of attention of several researchers in the field of applied Operations Research. This research deals with some production planning problems of Flexible Manufacturing Systems (FMS). The word "FMS" is, perhaps, the most overused word in the field of automated production systems. Therefore, it is necessary to define precisely the type and configuration of the automated production system under study. This is done in Chapter 2. The configuration assumed is fairly general and is in conformity with the trends in the contemporary installations of FMS. Chapter 3 reviews the previous research in modelling the FMS planning problems and discusses the relevance of this research to the effective management of the system. The nature of the FMS planning problems and the two planning problems to be modelled are discussed in Chapter 4, viz.,

- Operation allocation problem
- Part mix determination problem

Chapter 5 discusses the mathematical modelling of these two problems. The operation allocation problem is formulated as a nonlinear 0-1 integer program. The part mix determination problem is formulated as a general integer program. Alternate objective functions are proposed and realistic system constraints are defined. The formulations consider
the important planning aspects of refixturing and tool availablity which have not been reported in the literature so far. New constraints have been incorporated to account for the forced grouping of operations and the limited availability of standard fixture elements shared by different fixtures. The availability of alternate formulations of the FMS planning problems facilitates the manipulations of the resource allocations to the FMS, required in order to achieve the target system performance. The linearization strategies and computational results of the proposed formulations are given in Chapter 6. Five different linearization strategies are investigated in order to choose the best one(s) for the proposed formulations. A complete example of the proposed formulations is presented using realistic data. The optimal solutions to the formulations are obtained using the SAS/OR (Version 5) package which employs the well known Branch and Bound method of solving a mixed linear integer program. The computational behaviour of the formulations is discussed. Chapter 7 summarizes the modelling work with some comments on the scope for further research.
Chapter II

FLEXIBLE MANUFACTURING SYSTEMS

A flexible manufacturing system (FMS) is an integrated, computer controlled complex of automated materials handling devices and numerically controlled (NC) machine tools that can simultaneously process medium sized volumes of a variety of part types. [Stecke, 1983].

2.1 Components of FMS:

The development of flexible manufacturing systems is a relatively recent phenomenon although the components of FMS have had independent existence for quite some time. An FMS, as defined above, usually consists of the following major components.

(i) Numerically controlled machines
(ii) Automated materials handling system
(iii) Automatic tool changers and robots
(iv) Pallets and fixtures
(v) Control computer(s)

Numerically controlled machines and equipment are at the heart of an FMS. They differ from the conventional machine tools in the way they are equipped and controlled to perform different pre-determined sequences of operations.
The movements of the machine spindles and work tables are effected by various sets of servomotors which can be controlled electrically. It is, therefore, possible to effect the movements of tools and work tables by sending pre-set electrical pulses to the motors. Electronic controllers can be programmed either manually or by computer for the generation and sequencing of such pulses. Once programmed the machine tool can perform the assigned sequences of movements without any manual intervention. The tools required for carrying out the programmed operations are provided in a tool magazine located on or near the NC machine. The task of locating a specific tool and loading it on the machine spindle is performed by an automatic tool changer acting under the command of a control computer. There is, therefore, almost no manual intervention required for machine setup when the machine is required to perform two completely different operations successively on two different parts. The tools are specially designed such that each tool can fit in a slot of standard dimensions in the tool magazine. A typical tool magazine contains about 50 to 100 different tools. The result of such elaborate automatic control is that the machine can be set up, speedily and without manual intervention, to perform the next operation on the oncoming part, once the identity of the part is determined electronically by special sensors located on the transport mechanisms. Reduced machine setup times is one of
the main advantages of FMS over the conventional manufacturing systems.
The transport of parts within the FMS is usually automated.
There are two basic modes of automated transport.

(i) Moving conveyors
(ii) Automated Guided Vehicles (AGV)

The conveyors are cheaper to install but they introduce rigidity in the system because the parts must always follow the same route and direction and the movements of parts are dependent on each other. The AGV’s are generally much more flexible in that they can move independently and in less restricted fashion than the parts on a conveyor. The movement of AGV’s is controlled by central supervisory computers which are programmed to route the AGV’s in the best possible way.

All parts to be processed in the FMS are mounted on ‘pallets’ of standard dimensions. Pallets are standard part mounting surfaces which can be loaded on the conveyor, AGV or the machine tables. The parts remain fixed to the pallets at all times. The pallets are so designed as to suit a wide variety of part types, although special pallets are sometimes required for unusual part types. Fixtures are devices required to orient and hold parts on the pallets in precise positions as required by the processing steps and machines. A part may need a different fixture for each processing step, leading to intermittent refixturing. Such refixturing
is usually done manually at a refixturing center, normally located in or near the central parts storage.
Most of the FMS installations incorporate a central parts storage to store pallets with semi-finished parts. The size of the central storage is often large since it may be used to hold the finished parts as well. It determines the maximum number of parts that can be loaded in the system simultaneously.
The control computer is, perhaps, the most important component of the FMS. It performs a wide range of control functions including,

(i) Keeping track of the location and processing stages of different parts within the system on a real time basis.
(ii) Keeping track of the location and availability of cutting tools within the system.
(iii) Keeping track of the transport mechanism status and directing the movements of carts etc.
(iv) Sending appropriate NC programs to the controllers of the NC machines to perform the desired operations on the loaded part.
(v) Scheduling and sequencing the parts to be processed on the machines.

There is usually more than one computer required to manage the control functions of the FMS. These computers may be organised in a hierarchical structure within the plant,
Figure 1: Layout of the FMS
each computer communicating with others. In sophisticated FMS's very little or no manual intervention may be needed to perform the routine production activities as long as the control computer has been fed the required information and the required resources, like tools, are provided in the system. Due to the computer controlled automation installed in the FMS its ability to speedily adapt to changes in the external production environment is much greater than the conventional, manually controlled manufacturing systems and hence these are called 'flexible' manufacturing systems. An excellent summary of such manufacturing flexibilities of the FMS is given by Browne et al. (1984).

2.2 Configuration of FMS:

It is easy to see that due to the inherent generality of the definition of the FMS (given in Chapter 2), it is possible to have several completely different manufacturing setups referred to as 'FMS'. It is, therefore, necessary to define the configuration of the FMS assumed for the subsequent analysis as precisely as possible.

Referring to Fig.1, the configuration of the FMS under consideration is discussed below.

There are M machines in the system. Each of these machines is assumed to have general processing capabilities so that a range of different manufacturing operations can be
performed on the machine. It is possible to assign specific operations to be done on specific machines or the machines can be provided with identical sets of tools so that any type of part can be processed on any machine. In the following discussion it is assumed that all machines are not identical and certain specific operations have to be assigned to specific machines. A machine \( j \) is assumed to be equipped with a tool magazine having capacity of \( t_j \) slots \( (j=1,\ldots,M) \) and is provided with automatic tool changers to replace tools quickly. These machines process parts which are transported to and from the machines by automatic materials handling system. This system consists of transport carts, i.e., loading and unloading carts. The movement of the carts is computer controlled on a real time basis. There is a central storage for semi-finished parts in the system. Space is usually available in the form of rotary or linear pallet changers for storing parts in front of each machine and after each machine. The transport carts shuttle between the central storage and the machines. After leaving a machine \( j \), the part is transported to the central storage by a cart (distance traveled = \( U_j, j=1,\ldots,M \)). Similarly, a part from the central storage is transported to a machine \( j \) by another cart (distance traveled = \( L_j, j=1,\ldots,M \)). Since the total storage space within the system is finite it is assumed that the entire system can accommodate a
maximum of $N$ parts at any time. Each part usually occupies the same space irrespective of its type because each part is mounted on a pallet of identical dimensions to suit the dimensions of the transport carts and the machine tables. Since the central storage is assumed to have enough capacity to store $N$ parts at any time, space is normally available in the central storage for an arriving part from the machines.

In general, there will be more than one type of part being processed in the system at any given time. The sequence of operations for each part is assumed to be known and fixed. A part can visit a machine more than once and a part leaving machine 'i' can visit the same machine next if its routing requires so.

2.3 Mode of Operation of FMS:

Consider any one machine. The part being processed on this machine is transferred to the output buffer of this machine on completion of the operation, from where it is picked up by an unloading cart. The machine will be free to accept the next part from the input buffer. Each transfer of the part from the input buffer to the machine and from the machine to the output buffer takes little time. The part may or may not leave the machine after completion of one operation depending on whether the next operation is planned on the
same machine. Even if the next operation is planned on the same machine the part will have to leave the machine if refixturing is required before the next operation. Since the system can accomodate only N parts it is possible to have a varying number of parts in the system due to variations in the schedules of the upstream production shops. However, one of the control strategies employed in FMS is to maintain a constant number of parts of each type in the system at all times [Solberg, 1977]. This reduces the complexity of control, partly because the resource requirements can be estimated and provided in advance (e.g., cutting tools). The part types to be processed simultaneously also may be determined in advance. In this strategy, the parts entering the system are controlled such that the number of parts of type i, n_i, is held constant at all times, i.e.,

\[ n_1 + n_2 + n_3 + \ldots + n_R = N \]

where, R is the number of part types present in the system. A part, after completing its last operation, will be carted away to the central storage where it will be replaced by a new unprocessed part of the same type. (This procedure is not as unrealistic as it may seem at the first glance. In a typical production situation the parts do not arrive at the shop in a random fashion. The production is 'scheduled' so that the parts move within the plant in batches. It is common to have batches of parts of different types waiting
at the FMS to get processed. It is, then, easy to see that
to maintain a constant number of parts in the FMS, the
policy to be followed is to replace a leaving part by a
part of the same type.)

The configuration of the system assumed for this research
is based on the descriptions of similar systems given by
Ranky (1983) and Hartley (1984) and the current periodicals
in the area of automated production systems such as The FMS
Magazine, etc.
CHAPTER III

LITERATURE SURVEY

The planning problems of the FMS relate to the design as well as the operation of the FMS. Stecke (1984) has given an excellent summary of the production planning problems of FMS. The problems have been classified as,

(i) FMS design problems
(ii) FMS planning problems
(iii) FMS scheduling problems
(iv) FMS control problems

The first, second and fourth types of problems could be termed as 'static' planning problems since they refer to static (one time) allocations of FMS resources. The scheduling problems involve dynamic allocation of resources and may be termed as 'dynamic' planning problems. Section 4.1 discusses these problems briefly.

A comprehensive review of the FMS modelling techniques is presented by Wilhem and Sarin (1983) and Buzzacot and Shanthikumar (1980). The basic modelling approaches are,

(i) Networks of Queues (analytical modelling)
(ii) Simulation
(iii) Mathematical programming
Each approach is useful for a particular class of FMS problems. For example, the simulation approach can be effectively used to model the dynamic planning problems such as sequencing and scheduling, whereas network based queueing models are suitable for determining the average performance measures of the FMS. Mathematical programming is applicable in certain static planning decision models, since the complexity of dynamic planning decisions, such as scheduling, is difficult to model by the existing mathematical programming techniques [Wilhem and Sarin, 1983]. The use of queueing networks to realistically model FMS is severely restricted by the fact that the existing models of the networks of queues do not allow incorporation of various realistic features of FMS like limited buffer capacities at the stations, complex server disciplines and distributions etc. Mathematical programming has been used by researchers in order to provide optimal resource allocation as input to the simulation models of the system. So far, simulation models have had the widest scope of application to the FMS problems due partly to the scarcity of other types of models [Acree, 1983, El Maraghy, 1982, Stecke and Solberg, 1981, Wilhem and Shin, 1985, etc.]. This research deals with the static production planning problems of FMS modelled by mathematical programming techniques. A review of the FMS planning problems can also be found in Sinha and Hollier (1984).
A realistic mathematical programming formulation of the FMS planning problem appears in Steke and Bolberg (1979) in which several 0-1 integer programming formulations of the FMS loading problem have been presented. The importance of these formulations lies in the fact that these have been used successfully to model a real life Sundstrand/Caterpillar FMS. Shanker and Tzen (1985) have reported a similar analysis of the FMS loading and dispatching problem and have attempted to verify the quality of such solutions by employing a simulation program using the solutions of the mathematical programming model as inputs. This analysis has been done for a randomly generated FMS configuration. Kimemia and Gershwin (1985) have proposed a network flow optimization approach to determine the optimal part routing policy for an FMS after the part mix and operation allocation decisions have been taken. This approach is proposed for on-line control of FMS. Nof et al. (1979) have discussed the significance of operational control policies such as part type selection, part mix determination, part entry, etc., and have argued that optimal control policies do exist and must be pursued. Kusiak (1983) has reported some integer programming formulations of the FMS loading problem which are simple. Although computationally attractive, such simple formulations tend to be unrealistic in that they do not consider all the important system constraints peculiar to
FMS. (The tooling constraint proposed by Stecke and Solberg(1979) is an example of such realistic formulations). These are, however, valuable in gaining further insights into the operation of the FMS.

The availability of efficient solution algorithms has been the main constraint in the real life application of the mathematical programming models. The FMS loading formulations inevitably involve 0-1 integer variables. Both Stecke and Solberg(1979) and Shanker and Tzen(1985) have used commercially available integer programming (IP) codes to obtain solutions to the formulations. Both have reported that the amount of computer time required to solve the formulations for the medium size FMS configurations is high due to the inefficiencies of the IP algorithms. Stecke (1983) has discussed the computational behaviour of the integer programming formulations of the FMS loading problem.

Although this may not be true for all formulations, it is a well known fact that the computational efficiency of the existing integer programming algorithms decreases rapidly with the increase in the number of integer variables [Glover,1975]. It appears that the applications of such mathematical programming formulations will have to be restricted to small to medium sized FMS installations until more efficient integer programming codes are available. There is, however, no other modelling approach
available for problems such as operation allocation except repeated simulation runs of the system, which is basically a trial and error approach to such problems.

The performance of the FMS is a result of several interacting variables of the system. It is often difficult to isolate the effects of a particular variable on the performance of the system. Also, no particular modelling formulation can be claimed to be universally superior to the others because different FMS configurations would react differently to the same changes in a particular variable of the system. Based on the literature survey it appears, therefore, that no one set of formulations could be proposed for universal application. Stecke and Solberg (1979) have given an example of the tendency of the practitioners to load the system in order to balance the workload on machines as much as possible. Even a casual survey of the literature related to the operational analysis of the conventional manufacturing systems shows that the principle of work load balancing is applied whenever possible (e.g. assembly line balancing). Although quite adequate in some cases, it has been shown that for certain configurations of FMS it is not the best objective for achieving superior system performance [Stecke and Solberg, 1979]. Therefore, it is necessary that alternative formulations be proposed for the same problem. This research is partly
motivated by these considerations. Two alternate formulations of the operation allocation problem have been proposed. These take into account the important planning aspects of refixturing and part travel which do not seem to have figured in the FMS modelling literature so far. Also, the constraint set of the problems has been expanded and made more realistic by addition of new constraints to account for the limited availability of cutting tools. It is absolutely necessary to include as many real life constraints as possible in the planning of an FMS. In highly automated production systems it is obviously desirable to have the optimal control and resource allocation strategies decided by the control software of the system with minimum human intervention. The inclusion of more real life constraints would serve to reduce human attention for manipulating the strategies since the system's own solution will be more realistic. It is felt that the present research would aid efforts in this direction.

It is necessary to understand why the proposed formulations of the part mix determination problem must be included with those of the operation allocation problem. The part mix determination is necessary to decide how many parts of each type should be loaded into the system at any one time. Obviously, the choice of a part mix would affect both the magnitude and distribution of workload in the system.
The workload in the system is also dependent on the operation allocation policy. Without the knowledge of the operation allocation policy it would be difficult to assess the workload resulting from any part mix. The part mix determination is, thus, not an independent decision. It must be made with regard to other operational policies such as operation allocation. Also, it must be made consistently with the objectives of operation allocation since both decisions affect the workload in the system. The part mix formulations, although simple, have been included in this research to emphasize the need for such consistent planning. The survey of FMS modelling literature reveals little mathematical treatment of the part mix problem, perhaps due to the fact that the mathematical programming approach to the more basic problem of operation allocation has been relatively scarce in the literature. So it is not surprising that the dependent problem of part mix determination has not been adequately discussed. Stecke and Solberg (1979) have mentioned the importance of the these two planning problems.

There are few manufacturing systems existing today which are truly 'flexible'. Due to the characteristically high investment costs and high levels of technology involved, the installation of such systems has been monopolised by large corporations. This conclusion is supported by a survey of the FMS installations in U.S.A. by Kochan (1985).
A major amount of research in this field is being conducted commercially by the corporations, the suppliers of FMS and the consultants. Due to the proprietary nature of such work, little or no information is made public. As a result, a relatively small amount of real-life, practical research has been reported in the literature. Stecke and Solberg (1979) have argued that in spite of the large amount of work reported in the areas of conventional manufacturing systems, little published material is available which can be applied to the realistic planning problems of FMS.
CHAPTER IV

PLANNING PROBLEMS OF FMS

4.1 Production Planning Problems of FMS

The planning of FMS covers a broad range of activities from selection and layout of machines and equipment to designing preventive maintenance schedules for machines. The sophisticated technology of FMS has complicated the decision process considerably because many more possible consequences of an action must now be evaluated before taking the decisions. For example, the decision logic for diverting a part away from a machine, which has suddenly become inoperative, must be planned and incorporated into the scheduling logic before the system is activated. A failure to do so may often result in expensive idle time, not only for the affected machines, but perhaps for the entire FMS due, for example, to the blocked conveyor. The increased risk of underutilising highly capital intensive manufacturing facilities by poor planning has lead the FMS practitioners to attempt to systematically identify and model the planning problems of FMS. Holz (1985) has discussed the importance of planning for successful implementation of
FMS. The problem classification of Stecke (1984) is again used below to briefly explain the nature of the problems.

(i) FMS design problems:
These are the problems which have to be solved once the decision to install an FMS has been taken. The problems concern selection of hardware and software for the FMS. These include,

- selection of machines and equipment
- determination of the control software
- determination of part types to be manufactured in the FMS.
- determination of layout of machines and equipment, etc.

(ii) FMS (pre-release) planning problems:
These involve decisions to be taken before parts are loaded (released) into the installed FMS. These include,

- selection of a set of part types for immediate production.
- determination of the proportion at which the selected part types will be processed by the FMS.
- allocation of specific processing operations and tools to specific machines.
- determination of number of carts to be made operational, etc.

(iii) FMS scheduling problems:
These problems pertain to the real time control of the production activities within the FMS when it is
operational. These include,

- determination of sequence of parts to be processed on a machine

- determination of alternate part routing, etc.

(iv) FMS control problems

These are the problems of ensuring that the machines and parts are meeting the intended standards of performance and quality respectively. These include,

- design of preventive maintenance schedules for machines and equipment.

- development of automatic tool and parts inspection systems, etc.

This research deals with two specific pre-release planning problems (type (ii)), namely,

(i) Operation allocation problem

(ii) Part mix determination problem

These are discussed separately in the following sections.

4.2 Operation Allocation in FMS

As mentioned before, an FMS is capable of simultaneously processing several different types of parts. Each part requires to undergo one or more steps of processing. These processing steps are usually divided into one or more groups called 'operations'. An operation may consist of one or more processing steps. The division of processing steps
Figure 2: Sample part - Gear Box Cover
into operations is done logically by considering the nature of processes involved and the technological implications (e.g. reference surface availability). Each processing step requires one or more specific cutting tools. Therefore, an operation also requires a set of specific tools. As an example, consider the following 10 processing steps of a gear box cover casting as shown in Fig.2.

**PROCESSING STEPS**

1. ROUGH MILL BOTTOM FACE
2. FINISH MILL BOTTOM FACE
3. ROUGH MILL TOP EDGE
4. DRILL 4 X 10 mm Dia. HOLES
5. REAM 4 X 10 mm Dia. HOLES
6. DRILL 12 mm Dia. CENTER HOLE
7. ENLARGE CENTER HOLE TO 16 mm Dia.
8. ENLARGE CENTER HOLE TO 20 mm Dia.
9. ROUGH BORE CENTER HOLE TO 22 mm Dia.
10. FINISH BORE TO SIZE.

**CUTTING TOOLS**

- FACE MILLING CUTTER (M1)
- FACE MILLING CUTTER (M1)
- SIDE MILLING CUTTER (M2)
- DRILL (D1)
- REAMER (R1)
- DRILL (D2)
- DRILL (D3)
- DRILL (D4)
- BORING TOOL (B1)
- BORING TOOL (B2)

These 10 processing steps can be grouped into two different sets of 'operations' as follows.

**OPERATIONS SET 1**

- Operation #1: Steps 1,2
- Operation #2: Step 3
- Operation #3: Steps 4,5
- Operation #4: Steps 6,7,8
- Operation #5: Steps 9,10

**OPERATIONS SET 2**

- Operation #1: Steps 1,2
- Operation #2: Step 3
- Operation #3: Steps 4,5,6
- Operation #4: Steps 7,8,9,10

Usually it is the process planner who groups the processing steps into operations. It is possible, however, to alter such grouping to suit the FMS requirements. In this research it is assumed that the processing steps have already been grouped into operations and the exact tool
requirements for all operations are, thus, known. In the FMS environment an operation usually consists of the processing steps which can be performed successively on the same machine. Thus, a part needs to visit one machine for completing one operation.

The machines of the FMS usually have general processing capabilities due to the numerical control and the provision of tool magazines as explained before. So, theoretically, any operation could be performed on any machine as long as tools are made available on the machine. Availability of such general purpose machines should, then, ease the production planning and control problem considerably since the only control action required is to locate an idle machine for loading the next part. In practice, this may not happen due to several reasons as follows.

(i) Each machine has a limited number of tool slots to hold the tools in the tool magazine. This restricts the number of different operations that can be performed on any one machine.

(ii) It is often complicated, if not impossible, to perform certain types of operations on certain types of machines due to the technological and machine constraints, e.g., the drilling operations in the above example are best done on a vertical spindle machine, although it may be possible to do these
operations on a horizontal milling machine with special fixtures.

(iii) Considerations like workload balancing, part travel, number of refixturing often dictate machine preferences.

It is, therefore, necessary to assign each operation of each part type to one or more specific machines before the actual production can commence and right tools must be provided on right machines. The operation allocation problem is the problem of assigning each operation of each part type to the machines of the FMS, within the capacity and technological constraints of the system, so as to optimize the stated objective of such allocation.

4.3 Part Mix Determination in FMS

A recent survey of major U.S. companies indicates that the number of different part types that can be processed in an FMS varies from less than 10 to 150 or more [Kochan, 1985]. However, the number of different part types processed simultaneously at any given time is usually not so high. It is realistic to assume that 2 to 8 different part types may be processed simultaneously by a medium sized FMS [Mortimer, 1984].

The parts to be processed are loaded on 'pallets' which move within the system as explained before. There is
obviously a limited amount of space available inside the FMS to accommodate these pallets. The number of pallets that can, theoretically, be loaded within the system is seldom approached by the number of pallets actually loaded due to the problems of traffic congestion and fixture availability. The operational policy regarding the number of parts of each type may vary in each situation. In an environment where the sequence and types of parts arriving at the FMS from the upstream production lines is uncertain, it is reasonable to expect that the number of pallets in the system will fluctuate in order to accommodate unexpected arrivals. However, FMS's of today form only a part of the overall production process of firms, operating according to batch production schedules, and hence are subject to scheduled part inputs instead of random ones. Therefore, the part types still move in relatively well defined batches in most of the plants today and the set of part types selected for production in an FMS can remain constant over a reasonable length of time (few shifts). (Admittedly, the situation may arise in future when the FMS will be expected to serve a randomly varying input of parts. However, based on the reported capabilities of the existing systems it is felt that the kind of integration and sophistication of facilities required to deal with such situations will be beyond practical considerations for quite some time to come. Such production planning
situations, if and when they occur, will be much more complex and new definitions and solutions to the planning problems may have to be sought.)

Apart from the limited availability of space within the FMS, there are other reasons why adoption of part mix policy is essential.

The number of fixtures required for each part type is usually limited (Fixtures are often very expensive due to the precision machining required for manufacturing them).

It is not possible to load a large number of parts in the system even if space is available because each part must have the right fixture before it is loaded into the system.

If standardized fixture components are being used in different fixtures then a limited availability of such components restricts the total number of parts requiring fixtures with standardized components.

As mentioned in Chapter 3, it is necessary to adopt a part mix policy which is consistent with the operation allocation policy so as to achieve the desired objectives of pre-release planning of FMS. One of the operational policies of FMS in the scheduled production situation is to maintain a constant number of parts of each type in the system at all times as long as the selected set of part types remains unchanged. Since the types of parts being processed are known, this automatically sets the total number of parts (pallets) in the system to a constant value.
The part mix determination problem is to determine the relative proportion (or number) of parts to be maintained in the system at all times such that the total number of parts in the system is fixed.

4.4 Effects of Operation Allocation And Part Mix on FMS Operations

4.4.1 Transport activities

The load on the transporter mechanism in FMS is directly related to,

(i) Layout of FMS components
(ii) Operation allocation for a given set of part types
(iii) Relative proportion of each part type in the FMS.

The layout of FMS obviously affects the load on the transporter mechanism because the distances between machines and central storage are dependent on the physical locations of machines. In the type of layout shown in Fig. 1 the location of central storage with respect to all machines is important because parts are frequently transported to and from the central storage. If layout of the FMS is known then the load on the transporter mechanism can be estimated by determining the frequency of visits of parts to each machine.

The frequency of visits of parts to machines is a function of both operation allocation and part mix. For the same
part mix, allocation of a greater number of operations to machines farther away from the central storage would result in parts visiting these machines more often and for each visit of a part to a machine the loading/unloading carts will have to move a greater distance. On the other hand, for the same operation allocation, the increase in the proportion of one type of part can impose more load on the transporter mechanism than that imposed by other part types. This may happen, for example, if the operations of the part type, of which the proportion is increased, are assigned to machines away from the central storage. The transport load imposed by one unit of a part type also depends on how many of its operations are planned successively on the same machine(s). If two consecutive operations of a part type are assigned to the same machine, the part will not be transported back to the central storage after the first of the two operations is finished, but will be retained on the machine for the second operation, unless refixturing is required for the second operation. It may, then, be a good policy to assign consecutive operations of a part type to the same machine. Stecke and Solberg(1979) have formulated the FMS loading problem by considering this aspect of operation allocation and has observed that such allocation could frequently result in greater workload imbalances on the machines which may or may not be desirable.
The FMS could be set up to cope with the changes in the transporter load mainly by changing the number of AGV's or by changing the speed of conveyor. But, clearly, the margin for such changes is limited and any excesses of transporter load must be controlled by altering the resource allocation policies such as operation allocation or part mix.

4.4.2 Cutting tools

The distribution of tools in FMS is clearly dependent on the operation allocation to machines since each operation has a specific set of tools associated with it which must be provided in the tool magazine of the machine to which the operation is assigned. Since a tool occupies one or more tool slots in the tool magazine, each operation has its own tool slots requirement depending on the number of tools required for the operation. A particular allocation of operations to a machine is infeasible if the total tool slots requirement of all the operations assigned to it exceeds the tool slots available in the tool magazine. It should be noted that the tool slots requirement of a set of operations is not necessarily equal to the sum of the tool slot requirements of each operation within the set. This is so because some of the operations in the set may require one or more identical tools or large adjacent tools may share one slot by overlapping. Since it is possible to
interchange tools quickly on an NC machine and since only one part can be processed on the machine at a time, it is sufficient to provide only one such tool in the tool magazine of that machine.

The allocations of operations requiring one or more identical tools to the same machine can thus lead to lower tool requirement for the FMS. This consideration is important since cutting tools are usually quite expensive. Operation allocation is constrained in yet another way by the availability of tools. If a specific tool is available in limited quantities (say, two numbers) then clearly a particular operation allocation assigning all operations using this tool to more than 2 different machines is infeasible. Operation allocation in the FMS is thus directly affected by the cutting tool availability.

If one operation is assigned to only one machine, varying proportions of part types would not affect the number of tools required in the FMS. However, the number of tools required could change if an operation is allocated to more than one machine to cope with an unplanned increase in the proportion of a particular part type. Also, the part mix could affect the operation allocation, and hence the tool distribution within the FMS, if the objective of operation allocation is related to the part mix. For example, if the operations are allocated so as to minimize workload imbalances across the machines then the
operation allocation becomes dependent on part mix since
the workload imbalances are a function of both part mix and
operation allocation. On the other hand, tool distribution
may be independent of the part mix if the objective of
operation allocation is independent of part mix.

4.4.3 Refixture activities

As mentioned before, parts are positioned and mounted on
the pallets by standard or specially designed fixtures. It
may be necessary to refixture a part at some stage during
its processing. Such refixturing may involve a mere
reorientation of the part using the same fixture or a
complete change of fixtures. In any case, the part has to
leave the machine and visit a refixture station, usually
located near the central storage area. Such refixturing is
necessitated by one or more of the following reasons.

(i) The reference surface for the next operation may not
be accessible due to the current pattern of holding
the part.

(ii) The machine spindle may not be able to reach and/or
move on the work surface as required because of the
current position of the part.

(iii) A specific type of fixture may be needed to perform
the next operation due to the nature of the machining
process involved.
Although the sequence of operations for a part usually remains unchanged, different operation allocations can result in different sequences of machines visited by a part within the FMS. Different sequences of machine visits may necessitate more or less refixturing activity for the same part type since the part may have to be re-oriented more or less number of times for each different sequence of machines visited. If all the machines in the FMS are identical then any sequence of machine visits would require the same amount of refixturing for a particular part type but it is common to have NC machines of different sizes and capabilities within the same FMS (e.g. vertical spindle and horizontal spindle NC machines).

Whereas operation allocation determines the sequence of machine visits, and hence the number of refixturing per unit of a part type, the part mix can affect the total fixturing load on the refixturing station. For example, given a specific operation allocation, an increase in the proportion of a part type requiring a greater number of refixturing will increase the load on the refixturing station and vice versa. The amount of refixturing required may become a consideration for the systems which are expected to be operational continuously for several hours with minimum manual intervention. A greater amount of refixturing would normally imply more frequent interruptions during the operational run of the FMS which
conflicts with the need to increase the 'cutting time'
proportion of the available time.

4.4.4 Throughput and machine utilization:

The throughput and machine utilizations of FMS are the two
primary measures of performance which are usually evaluated
to investigate the effectiveness of the strategies of
operation of FMS. It is easy to see that these two measures
would be complex functions of the system parameters and the
resource allocation strategies such as operation allocation
and part mix determination. It is often difficult to
isolate the effects of a particular operation allocation
and part mix and make generalized conclusions. The effects
of a particular strategy of resource allocation, like
operation allocation, can be judged by running a detailed
analytical or simulation model of the specific FMS
configuration under consideration [Shanker and Tzen, 1985].
Such models usually view the FMS as a network of queues and
attempt to determine the average values of measures like
throughput and machine utilizations. The solutions of the
operation allocation and part mix problems provide inputs
to such model building. Even after analysing the results of
simulation models, the effects of a particular resource
allocation strategy may not be quite distinguishable since
the operation of the FMS is greatly affected by other
operational policies of the FMS like scheduling and sequencing, etc. Such analysis, however, may provide valuable indicators to the effectiveness of the strategies employed. For example, if the simulation runs indicate unusually high transport time proportion it may be worth generating a fresh operation allocation with the objective of reducing the transport distances for part types. This would require modelling the operation allocation problem with the objective of minimizing transport distances and suitably altering the part mix in the FMS.
Chapter V

MATHEMATICAL MODELLING

The formulations of objective functions of the operation allocation and part mix determination are discussed in sections 5.2 and 5.4 respectively. The constraint formulations are explained in sections 5.3 and 5.5 respectively. The following section describes the assumptions and notation employed in the subsequent sections.

5.1 Assumptions and Notation Used:

The assumptions made for the purpose of modelling are as follows.

(i) Detailed process sheets are available for each part type giving information about cutting tools and fixtures required, etc.

(ii) Part types have been selected for production and the selected set of part types remains unchanged at least for a few shifts.

(iii) At least one complete set of each tool type is available.
(iv) An operation is assigned to only one machine.
(v) One part is mounted on one pallet.
(vi) All pallets are of identical dimensions.
(vii) Duplication of a tool is not permitted in the same tool magazine.
(viii) The refixturing station is located in the central parts storage area.

The notation used is as follows.

\[ M = \text{Number of machines in the FMS.} \]
\[ N = \text{Number of parts to be maintained in the system at all times.} \]
\[ R = \text{Number of part types to be processed simultaneously in the FMS.} \]
\[ O_i = \text{Number of different operations to be performed on part type } i, \ (i=1,2,...,R). \]
\[ e_i = \text{Estimated or externally imposed part mix ratio for part type 'i', } (i=1,2,...,R). \]
\[ T_i = \text{Total processing time for a part of type } i, \ (i=1,2,...,R). \]
\[ W_{ij} = \text{Processing time of part type } i, \ (i=1,2,...,R) \text{ on machine } j, \ (j=1,2,...,M). \]
\[ t_j = \text{Number of tool slots available in the tool magazine of machine } j, \ (j=1,2,...,M). \]
\[ I_j = \text{Workload imbalance on machine } j, \ (j=1,2,...,M). \]
\[ U_j = \text{Unloading distance between central storage and machine } j, \ (j=1,2,...,M). \]
L_j = Loading distance between central storage and machine j, (j=1,2,...,M).

UB_i = Maximum number of parts of type i allowed to be loaded in the system, (i=1,2,...,R).

LB_i = Minimum number of parts of type i allowed to be loaded in the system, (i=1,2,...,R).

τ(K) = Number of elements of a set K.

B = Set of all operations of all part types.

n_B = Total number of operations to be allocated \[ n_B = \tau(B) \].

G = Set of operations to be grouped together.

A_H = Set of operations requiring tool 'H'.

n_H = Total number of operations requiring tool 'H' \[ n_H = \tau(A_H) \].

A_CF = Set of part types requiring a common fixture component 'CF'.

n_CF = Available number of common fixture elements 'CF'.

F_X_i = Total refixturing load due to part type i, (i=1,...,R).

D_i = Total transport load due to part type i, (i=1,...,R).

TD = Index set of all tool types required in the FMS for processing all part types.

(c_i)_{CF} = Number of standard fixture components of type 'CF' required to mount one part of type i on the pallet.

(i,k) = Denotes operation number 'k' of part type 'i'.
\( F(i,k,p,q) = \) Number of refixturing required by operation 'k' of part type 'i', when (k-1)th operation is done on machine 'p' and kth operation is done on machine 'q'.

\[ i=1,2,...,R; \quad k=1,2,...,O_i; \quad p=1,2,...,M; \quad q=1,2,...,M. \]

The decision variables of the operation allocation problem are,

\[ X_{i,j,k} = 1 \text{ if operation } k \text{ of part type } i \text{ is assigned to machine } j. \]

\[ = 0 \text{ otherwise } (i=1,...,R; \quad j=1,...,M; \quad k=1,...,O_i). \]

The decision variables of the part mix problem are,

\[ X_i = \text{Number of parts of type 'i' to be maintained in the system at all times } (X_i \text{ are integers}, i=1,...,R). \]

5.2 Operation Allocation - Formulation of Objective Functions:

5.2.1 Minimization of refixturing activities:

Each part type 'i' requires \( O_i \) number of operations which must be performed in the given sequence. Consider operation number 'k' of part type 'i'. The (k-1)th and the kth operation may be performed on any one of the M machines available, subject to the constraints of the problem. Having finished the (k-1)th operation on a particular machine, the part may have to be refixtured, if the kth
TABLE - 1

MATRIX OF $F(i,k,p,q)$'s FOR PART TYPE $i$, OPN. $k$

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>...</th>
<th>M</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
<td>...</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M</td>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
</tbody>
</table>
operation is performed on certain machines and need not be refixtured if the kth operation is performed on certain other machines in the FMS. Specifically, if the (k-1)th operation is performed on machine 'p' and the kth operation is performed on machine 'q', then,

\[ F(i, k, p, q) = \begin{cases} 
0 & \text{if refixturing is not required between opn.} \,(k-1) \text{ and } k, \\
1 & \text{if refixturing is required between opn.} \,(k-1) \text{ and } k.
\end{cases} \]

For each operation 'k', (k=2,3,...,M), of part type 'i', a matrix of \( F(i, k, p, q) \)'s can be formed based on the process planning expertise within the company as shown in Table-1.

For operation 'k' of part type 'i',

Number of refixturing =

\[ Q(i, k) = \sum_{p=1}^{M} X_{i, p, (k-1)} \left( \sum_{q=1}^{M} X_{i, q, k} \cdot F(i, k, p, q) \right) \]

Total number of refixturing for part type i =

\[ Q_i = \sum_{k=1}^{M} Q(i, k) \]

Total number of refixturing for all part types processed =

\[ Z = \sum_{i=1}^{R} T_F(i) \quad \quad \quad \text{(1)} \]

The objective function is, minimize Z.

Instead of measuring the refixturing activity in terms of the number of refixturings, it is possible to measure the
'degree of complexity' of refixturing. To do this, each factor \( F(i,k,p,q) \) can be multiplied by an externally assigned weightage \( w(i,k,p,q) \). The assignment of \( w(i,k,p,q) \) could be made in different ways. It could be made to reflect the relative degree of complexity of refixturing the part type 'i' for operation 'k' on machine 'q', when the \((k-1)\)th operation is performed on machine 'p', by assigning a value between 0 and 10 (or any other suitable scale) depending on the degree of complexity of refixturing. The amount of subjectivity in assigning these weightages could be reduced by using the estimated time requirement for refixturing as a measure of the degree of complexity. The objective function 'Z' will then be a measure of refixturing complexity instead of the total number of refixturings.

The use of this objective function in formulating the operation allocation problem requires that matrices \( (F(i,k,p,q)) \) and \( (w(i,k,p,q)) \) be formed for each operation 'k' of each part type 'i'. This type of work has traditionally been the responsibility of the process planning department within the company. Clearly, the number of possibilities to be considered is large, since each matrix has \( (M^2) \) elements. But such information would also be required for other purposes like alternate route generation for parts etc. So it is felt that the use of this objective function would not impose additional burden
of data collection on the planners. It may be noted, in passing, that such lengthy data preparation activities (usually a one-time exercise followed by short periodical revisions) occur rather frequently in the field of Computer Integrated Manufacturing Systems planning.

5.2.2 Minimization of transport load:

The assumed configuration of the FMS (Fig.-1) shows two different sets of transport paths, for loading the parts on the machines from the central storage and for unloading the parts from the machine to the central storage, respectively. As explained in section 4.4.1, the operation allocation policy has a major impact on the amount of distance traveled by the parts within the FMS. One of the objectives of operation allocation could be to reduce such travel to the minimum possible extent.

Consider a sample operation allocation for a part type 'i' as shown in Table 2. Assume that this part has four operations. The entries in Table 2 are values of \( X_{ijk} \)’s.

Now, consider the expression,

\[
(E1)_i = \sum_{k=1}^{M} \sum_{j=1}^{M} (X_{i,j,k} - X_{i,j,k+1})^2 (1/2)
\]

= The number of times part type 'i' changes machines.
TABLE 2

SAMPLE OPERATION ALLOCATION FOR PART TYPE i

<table>
<thead>
<tr>
<th>Operations (k)</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Stecke and Solberg (1979) have reported this as one of the formulations of the FMS loading problem.

Let, \( U_j \) = unloading distance of machine \( j \) from central storage.

\( L_j \) = loading distance of machine \( j \) from central storage.

Now, consider the following expressions.

\[
(E2)_i = \sum_{k=1}^{M} \left( \sum_{j=1}^{I} (X_{i,j,k} - X_{i,j,k+1})^2 \cdot (X_{i,j,k} - U_j + X_{i,j,k+1} \cdot L_j) \right)
\]

= total distance traveled by part type 'i' after completing operation 1 and before returning to the central storage after finishing the last operation.

The second bracketed expression ensures that appropriate distance is added to \((E2)_i\) when the first bracketed expression has value 1 (i.e. when a machine is changed).

\[
(E3)_i = \sum_{j=1}^{M} (L_j - X_{i,j,1})
\]

= Loading distance traveled by part 'i' for the first operation.

\[
(E4)_i = \sum_{j=1}^{M} (U_j - X_{i,j,0})
\]

= Unloading distance traveled by part 'i' after the last operation

So,

\[
(E2)_i + (E3)_i + (E4)_i = \text{The distance traveled by one unit of part type 'i' within the FMS.}
\]
This expression for the total distance covered by part type 'i' is correct only if the part does not have to leave the machine when its next operation is planned on the same machine. (As in the example above; for operation 1 and 2 and machine 2). In general, this may not be true and the part may have to leave the machine for refixturing even if its next operation is planned on the same machine. This information can, however, be obtained from the matrix of \( F(i,k,p,q)'s \). This additional distance traveled by part type 'i' is given by the expression,

\[
(E5)_i = \sum_{k=1}^{M} \sum_{j=1}^{M} (X_i,j,k - X_i,j,k+1) \cdot F(i,k+1,j,j) \cdot (L_j + U_j)
\]

So, the total distance traveled by part type 'i' within the FMS is,

\[
D_i = (E2)_i + (E3)_i + (E4)_i + (E5)_i
\]

The objective of operation allocation could be,

\[
\text{Minimize } Z = \sum_{i=1}^{R} D_i 
\]

...... (2)

It is possible, as before, to apply weightages, \( w(i) \), \( i=1,..,R \), to the distances traveled by part types. If the part mix ratio is estimated or is externally assigned as \( e_i, i=1,..,R \), then it can be used as \( w(i), i=1,..,R \). The objective function would then be,

\[
\text{Minimize } Z = \sum_{i=1}^{R} D_i \cdot w(i)
\]
5.3 Operation Allocation - Formulation of Constraints

Each operation of each part type must be assigned to one machine, i.e.,

$$\sum_{j=1}^{M} X_{i,j,k} = 1 \quad \text{for each operation} \ (i,k) \quad \ldots \quad (3) \quad i=1, \ldots, R; \ k=1, \ldots, D_1$$

It is possible that due to other considerations like system throughput it may be decided to provide more than one machine for one operation. Although the machines in the FMS usually have quite general processing capabilities there are definite limitations to the type and range of operations that they can perform. Also, some of the machines may be provided to do only a specific type of operation like turning. In such cases, it must be ensured that certain operations are not assigned to certain specific machines, i.e.,

$$X_{i1,j1,k1} = 0 \quad \text{for part} \ i_1, \ \text{operation} \ k_1 \ \text{and machine} \ j_1 \quad \ldots \quad (4a)$$

Obviously, similar constraints can be provided to ensure that certain operations are always assigned to a specific machine by making the right hand side of the Eq. (4a) equal to 1.

It may be necessary, at times, to ensure that certain operations are always grouped together on a machine. For example, consider the use of an automatic in-process gauging equipment which can be installed on or near any machine but which is available in limited quantities
(say, one). It may then be necessary to ensure that all operations requiring this equipment are grouped on the same machine. This constraint can be expressed as,

\[ \sum_{j=1}^{M} \prod_{(i,k) \in G} x_{i,j,k} = 1 \quad \ldots \quad (4b) \]

where, \( G \) is the set of operations to be grouped together.

Since the cutting tools are often expensive, it is not unusual to operate the system with a limited number of certain very expensive tools. If a specific tool is available in limited quantity (\( h \)), then the operations requiring this tool can not be allocated to more than \( h \) different machines in the FMS. The formulation of this constraint is explained below.

Consider a tool \( 'H' \) of which only \( h \) sets are available (\( h < M \); for, if \( h \geq M \) then one tool can be assigned to each machine, if required).

Let \( A_H = \{(i,k)/ \text{operation } (i,k) \text{ needs tool } 'H'\} \) be the set of operations requiring this particular tool \( 'H' \). The operations belonging to the set \( A_H \) may be of different part types. It is necessary that all operations in the set \( A_H \) must be assigned to at most \( h \) number of different machines. Let the set \( A_H \) consist of \( n_H \) different operations. Consider a sample operation allocation matrix for the operations of the set \( A_H \) as shown in Table 3. The elements of the matrix are values of \( X_{i,j,k} \)'s. Assume \( n_H=4 \).
TABLE - 3

ALLOCATION OF OPERATIONS OF SET $A_H$ TO MACHINES.

<table>
<thead>
<tr>
<th>Operations $(i,k) \in A_H$</th>
<th>machines $(j)$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>$S_{j,H}$</td>
<td>1</td>
</tr>
<tr>
<td>$Y_{j,H}$</td>
<td>1</td>
</tr>
</tbody>
</table>
Consider a machine \( j \). Then,

\[
S_{j,H} = \sum_{(i,k) \in A_H} X_{i,j,k} = \text{sum of entries in column \( j \).}
\]

Let \( Y_{j,H} \) be defined as follows,

\[
Y_{j,H} = \begin{cases} 
1 & \text{if } S_{j,H} > 0 \\
0 & \text{if } S_{j,H} = 0 
\end{cases} 
\]

\[ M \sum_{j=1}^{M} Y_{j,H} \leq h \] for tool type \( H \) \hspace{1cm} (\text{6})

This constraint, in its present form, can not be used unless \( Y_{j,H} \) is expressed as a function of the decision variables \( X_{i,j,k} \). This can be done as follows.

Referring to Table 3, consider all possible pairs \((X_{i1,j,k1}, X_{i2,j,k2})\) of the entries in the \( j \)th column. There are \( \binom{n_H}{2} \) possible pairs that can be made out of the \( n_H \) elements of column \( j \). Since \( S_{j,H} \) entries are nonzero \((=1)\) in each columns, \( \binom{S_{j,H}}{2} \) of these pairs will have value 1. So the sum of all the \( \binom{n_H}{2} \) pairs will be equal to \( \binom{S_{j,H}}{2} \), i.e.,

\[
\sum (X_{i1,j,k1} - X_{i2,j,k2}) = \binom{S_{j,H}}{2} \hspace{1cm} \text{for all operation combinations } ((i_1,k_1),(i_2,k_2)) \text{ belonging to set } A.
\]
In general, there can be \( \binom{n}{r} \) different terms that can be formed, each a product of \( r \) different elements of column \( j \). The sum of these entries for column \( j \) will be equal to 
\[
\binom{S_j}{r}
\]
Let \( V_{r,j,H} = \binom{S_j}{r} \)

Now, consider the expression,
\[
Y_{j,H} = \sum_{r=1}^{n} (-1)^{r+1} \cdot V_{r,j,H}
\]
\[
= \sum_{r=1}^{n} (-1)^{r+1} \cdot \binom{S_j}{r}
\]
\[
= 1
\]
( This is because \( \sum_{r=1}^{n} (-1)^{r+1} \cdot \binom{n}{r} = 1 \) )

It is easy to see that \( Y_{j,H} = 0 \), if \( S_j,H = 0 \) (i.e., if there is no nonzero entry in the \( j \)th column). So now \( Y_{j,H} \) is expressed as a function of the decision variables \( X_{i,j,k} \) and satisfies Eq.(5). Now, Eq.(6) can be written as,
\[
\sum_{j=1}^{M} \sum_{r=1}^{n} (-1)^{r+1} \cdot V_{r,j,H} \leq h \quad \text{for a tool 'H'.}
\]
\[
\ldots \ldots \quad (7a)
\]

This constraint will be repeated for each different tool 'H' which is available in limited quantities 'h', \( h < M \).

There is yet another type of complex tooling constraint, proposed by Stecke and Solberg(1979) to determine the
realistic tool slot requirements of each machine. A brief explanation of the formulation of this constraint follows using the notation similar to section 5.3.

Let $B = \text{Set of all operations of all part types in the system.}$

$n_B = \text{Number of operations belonging to set } B.$

Consider a particular tool "H" and a sample operation allocation $(X_{i,j,k})$ to machine $j$ as shown in Table 4.

Assume $n_B = 5.$

We transform $X_{i,j,k}$ to $X_{i,j,k,\text{H}}$ by,

$$X_{i,j,k,\text{H}} = X_{i,j,k} - C_{i,k,\text{H}}$$

where, $C_{i,k,\text{H}} = 1$ if operation $(i,k)$ requires tool "H" $= 0$ otherwise

The $(C_{i,k,\text{H}})$s are constants for a given problem and are derived from the process sheets of parts. (In the above example, assume that operation 5 does not need tool "H", so $C_{i,k,\text{H}} = 0$ for this operation.)

$S_{j,\text{H}} \text{ now applies to } (X_{i,j,k,\text{H}})$s instead of $(X_{i,j,k})$s, i.e.,

$$S_{j,\text{H}} = \sum_{(i,k) \in B} X_{i,j,k,\text{H}}$$

The definition of $Y_{j,\text{H}}$ is the same as before, i.e.,

$$Y_{j,\text{H}} = \sum_{r=1}^{n_B} (-1)^{r+1} \cdot V_{r,j,\text{H}}$$

i.e., $Y_{j,\text{H}}$ is as defined by Eq. (5).
TABLE 4

SAMPLE OPERATION ALLOCATION TO MACHINE $j$
(LIMITED TOOL AVAILABILITY)

<table>
<thead>
<tr>
<th>Opn.</th>
<th>$X_{i,j,k}$</th>
<th>$X_{i,j,k,H}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$S_{j,H}$</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>$Y_{j,H}$</td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>
(Now, \( Y_{j,H} \) is the number of tools of type \( 'H' \) required to be provided on machine \( j \)). Clearly, it is unnecessary to assign more than one tool of type \( 'H' \) to machine \( j \) since the same tool can be used for all the operations. The following constraint ensures that the tool slots usage of tool magazines is calculated by taking this fact into account.

Let each tool \( 'H' \) require \( t'_H \) tool slots in the magazine. Following similar logic as in section 5.3, the number of tools of type \( 'H' \) required on machine \( j = Y_{j,H} \)

\[
= \sum_{r=1}^{n_B} (-1)^{r+1} \cdot V_{r,j,H}
\]

and the number of tool slots required for tool of type \( 'H' \) on machine \( j = \)

\[
\sum_{r=1}^{n_B} (-1)^{r+1} \cdot V_{r,j,H} \cdot t'_H
\]

So far we have considered only one tool \( 'H' \). Now considering all tool types in the system, i.e. all \( H \in TD \), the total number of tool slots required on machine \( j = \)

\[
\sum_{H \in TD} \sum_{r=1}^{n_B} (-1)^{r+1} \cdot V_{r,j,H} \cdot t'_H
\]

Since each tool magazine contains \( t_j \) slots, the tool slot capacity constraint is,

\[
\sum_{H \in TD} \sum_{r=1}^{n_B} (-1)^{r+1} \cdot V_{r,j,H} \cdot t'_H \leq t_j \quad \ldots \quad (7b)
\]
The formulation of operation allocation problem thus consists of two different objective functions given by Eqs. (1) and (2), and the constraint set given by Eqs. (3), (4a), (4b), (7a), (7b). Both the objective functions and some constraints are nonlinear. The operation allocation problem has thus been formulated as a nonlinear 0-1 integer programming problem.

5.4 Part Mix Determination - Formulation of Objective Functions:

It is assumed that the operation allocation information is available for the formulation of the part mix problem. The following section describes some formulations of the objective functions. The constraints of this problem are explained in section 5.5.

5.4.1 Minimization of workload imbalances:

Since the operation allocation for each of the part types is known, it is easy to determine the total workload imposed on machine 'j' by one unit of part type 'i'. It is the sum of the processing times of all operations of part type 'i', assigned to the machine 'j' \((W_{ij})\).

The total workload on the machine at the time of loading is,

\[
W_j = \sum_{i=1}^{R} X_i \cdot W_{ij}
\]
The total work content of the FMS at the time of loading can be expressed as,

$$ W_T = \sum_{i=1}^{R} X_i \cdot T_i $$

So, the average workload on a machine at the time of loading is,

$$ W_{avg} = \frac{W_T}{M} $$

The workload imbalance on machine 'j' at the time of loading is defined as,

$$ I_j = |W_j - W_{avg}| $$

The objective function is,

$$ \text{Minimize } Z = \sum_{j=1}^{M} I_j $$

OR

$$ \text{Minimize } Z = \sum_{j=1}^{M} (W_j - W_{avg})^2 $$

5.4.2 Minimization of refixturing activities:

The number of refixturings required per unit of part types can be determined if the operation allocation information is available. If \((FX)_i\) is the refixturing load per unit of part type 'i' then the total refixturing load on the system at the time of loading is,

$$ Z = \sum_{i=1}^{R} (FX)_i \cdot X_i $$

The objective function is,

$$ \text{Minimize } Z = \sum_{i=1}^{R} (FX)_i \cdot X_i $$
5.4.3 Minimization of transport load:

The work load on the transport mechanism at any time is the distance yet to be covered by the parts within the system at that time. From the operation allocation information the total distance travelled by part type 'i' is known to be $D_i$. The total work load on the transporter mechanism at the time of loading is,

$$ Z = \sum_{i=1}^{R} X_i \cdot D_i $$

The objective function is,

$$ \text{Minimize } Z = \sum_{i=1}^{R} X_i \cdot D_i \quad \text{...... (10)} $$

5.5 Part Mix Determination - Formulation of Constraints:

The constraints of the part mix determination problem are much simpler than those of the operation allocation problem. The tooling does not impose restrictions since it is assumed that at least one set of tools of each type is available, so whatever the relative proportion of a part type may be, it will eventually get processed within the system. The upper bounds on the value of $X_i$’s may be provided by the number of fixtures available for one or more operations of the part types. A part may need different fixtures for each of its operations.
Fixtures are usually precision machined assemblies and hence tend to be quite expensive even for relatively simple designs. If a fixture for part 'i' is available in limited quantities, $UB_i$, then the constraints of the form,

$$X_i \leq UB_i \quad \text{for } i=1,\ldots,R; \quad UB_i \leq N$$

..... (11)

are introduced in the formulation.

Since 'R' part types have already been selected from several other part types for simultaneous processing, it is logical to impose positive lower bounds on $X_i$'s., i.e.,

$$X_i \geq LB_i \quad \text{for } i=1,\ldots,R; \quad LB_i \geq 1$$

..... (12)

As mentioned earlier, the operational policy of the FMS is to maintain a constant number ($N$) of parts within the system at all times, i.e.,

$$X_1 + X_2 + X_3 + \ldots + X_R = N$$

..... (13)

Similar to the constraints imposed by the fixture availability, certain additional constraints may have to be introduced if standardized fixture elements are used in building the fixtures.

Due to the potential diversity of part types and operations in the FMS, development of standardized fixture elements is of interest to the FMS practitioners [Drake, 1984]. The standardized fixture elements are fixture components which can be used in a variety of different fixtures in a modular fashion. Use of such elements can result in reduced expenditure on specially designed fixtures for new part types. The limited availability of one or more such
standardized fixture elements would restrict the number of parts requiring fixtures using such standardized elements. Specifically, let $A_{CF}$ be the index set of part types which are mounted on the pallet using one or more number of the standard fixture element 'CF' and $(c_i)_{CF} =$ number of standard fixture elements of type 'CF' required per part of type 'i'. Let 'CF' be available in quantity = $n_{CF}$. It is, then, necessary to impose the following constraint for each such fixture element 'CF' which is available in limited quantities.

$$\sum_{i \in A_{CF}} x_i \cdot (c_i)_{CF} \leq n_{CF} \quad \text{(14)}$$

The formulation of part mix determination problem thus consists of three different objectives given by Eqs. (8), (9), (10) and the constraint set given by Eqs. (11), (12), (13), (14). Except for Eq. (8) all the others are linear functions of integer variables $x_i \ (i=1,\ldots,R)$. Eq. (8) is a nonlinear function of $x_i$'s $\ (i=1,\ldots,R)$. 
Chapter VI

COMPUTATIONAL RESULTS

The formulations proposed in the previous sections were applied to a realistic problem of FMS planning. The results are presented in the following sections. Section 6.1 describes the data used for the application of the proposed formulations. The strategies used to linearize the objective functions and the constraints are discussed in section 6.2. Section 6.3 includes solutions to the problems and discussion of the results. The computational experience with these formulations is discussed in section 6.4.

6.1 Description of the Problems

As mentioned in section 2.3, the assumed configuration of the FMS is based on the published literature describing the existing systems. The part types and their process plans are based on previous experience in the process planning department of an automobile company. Most of the information required for making use of the proposed formulations is normally available in the companies in some form of documentation.
The system consists of 4 general purpose CNC machines (two horizontal spindle machines and two vertical spindle machines). Each machine is equipped with a tool magazine having 50 to 100 slots for storing tools. In general, the tool magazine capacity available for loading a given set of parts at a given time may be less than the maximum possible because the tool magazines may contain tools required for some other part types not being processed at that time. The available tool magazine capacities are assumed to be less than the maximum capacities of the respective tool magazines.

There is a central parts storage available for storing semifinished parts, which is large enough to store about 20 to 30 parts at any given time. Whether the system will be loaded with a specific number of parts will, however, depend on the downstream production requirements in the plant and the fixture and pallet availability. For this research, it is assumed that 20 parts of the currently selected part types will be maintained in the system at any time. As long as the set of selected part types remains unchanged, the policy of introducing new parts in the system will be to replace a completed part by a raw part of the same type.

The loading and unloading distances for a given machine are assumed to be the same (i.e. $L_j = U_j$, $j = 1, 2, 3, 4$), however, not all machines are located at the same distance.
from the central storage. A summary of the machine data is
given in Table 5. Machines 1 and 3 are assumed to be
smaller than 2 and 4 respectively, in terms of the physical
specifications such as stroke of spindles, worktables, etc.
Such variations in the physical characteristics of machines
can affect the refixturing requirements of parts since
reorientation of parts may be required more frequently.
Some such instances are given in Appendix - A. The
refixturing station is located in the central storage area.
There are 4 different part types that have to be processed
simultaneously in the FMS. The part sketches, their
respective process plans, and the refixturing data are
given in Appendix-A. The chosen part types are typical
machined castings, commonly used in similar forms in
machines, automobiles and other mechanical assemblies. A
typical production situation involving machining of such
parts may be found in the machine shop of a sub-contracting
firm which receives raw parts from the original equipment
manufacturers (OEM) and supplies semi-finished parts to
them. In such cases, the raw parts would arrive in batches
(to be returned in batches), forcing the sub-contractor to
plan according to the schedules of the OEMs. Due to the
ancillary nature of the business, the sub-contractor would
end up machining a variety of parts from different
customers over a period of time. It is, therefore,
important for the sub-contractor to find ways for reducing
### TABLE 5

**SUMMARY OF MACHINE DATA**

<table>
<thead>
<tr>
<th>No.</th>
<th>MACHINE DESCRIPTION</th>
<th>AVAILABLE TOOL SLOTS</th>
<th>LOADING DISTANCE</th>
<th>UNLOADING DISTANCE</th>
</tr>
</thead>
<tbody>
<tr>
<td>j</td>
<td></td>
<td>tj</td>
<td>Lj met.</td>
<td>Uj met.</td>
</tr>
<tr>
<td>1</td>
<td>Horizontal spindle CNC machine- small</td>
<td>15</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>2</td>
<td>Horizontal spindle CNC machine- large</td>
<td>28</td>
<td>30</td>
<td>30</td>
</tr>
<tr>
<td>3</td>
<td>Vertical spindle CNC machine- small</td>
<td>20</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>4</td>
<td>Vertical spindle CNC machine- large</td>
<td>35</td>
<td>25</td>
<td>25</td>
</tr>
</tbody>
</table>
his dependence on critical and long lead time items like specially designed fixtures required to mount the parts on pallets, in order to be competitive in the business. One effective way to reduce such dependence on fixtures is to use standardized fixture modules to "build-up" fixtures for a variety of part types. The constraints given by Eq. (14) are important in such situations.

The process plans of the parts consist of descriptions of operations, tools required for each operation and estimated process time for each operation. The refixturing data cover all possibilities of refixturing for each operation of each part type. The degrees of refixturing complexity, \( w(i,k,p,q) \), for all operations \((i,k) \) \( i=1, \ldots, R; \) \( k=2, \ldots, O_i \) have been assessed on a scale of 0 to 10 (for \( p=1, \ldots, 4; q=1, \ldots, 4 \)). A higher value of \( w(i,k,p,q) \) indicates a greater degree of complexity of refixturing. A summary of the parts data is given in Table 6. Due to the characteristics of the parts, operations and machines, it is not possible to perform certain operations on certain machines. Table 7 summarizes the possibilities of allocation for all part types considered here.

The tool requirements of all operations are summarized in Appendix B. It is assumed that tool number 11 and tool number 13 (both milling cutters) are available in limited quantities (two of each). Also, the last operations of part types 2 and 4 [operation \((i=2, k=4)\) and operation \((i=4, k=4)\) ]
TABLE - 6
SUMMARY OF PARTS DATA

<table>
<thead>
<tr>
<th>PART No.</th>
<th>PART DESCRIPTION</th>
<th>No. OF OPERATIONS</th>
<th>TOTAL PROCESSING TIME (Mins)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>SIDE COVER</td>
<td>3</td>
<td>38</td>
</tr>
<tr>
<td>2</td>
<td>MOTOR HOUSING</td>
<td>4</td>
<td>54</td>
</tr>
<tr>
<td>3</td>
<td>END COVER</td>
<td>3</td>
<td>35</td>
</tr>
<tr>
<td>4</td>
<td>HOUSING COVER</td>
<td>4</td>
<td>46</td>
</tr>
</tbody>
</table>

TABLE - 7
POSSIBILITIES OF ALLOCATION OF OPERATIONS

<table>
<thead>
<tr>
<th>PART NO.</th>
<th>OPERATION NO.</th>
<th>MACHINE (j)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1   2   3   4</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1   1   0   1</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>1   1   1   1</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>0   1   0   1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0   1   1   1</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>0   1   0   0</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>1   1   0   0</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1   1   1   1</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>1   1   1   1</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>1   1   0   0</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1   1   0   0</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>1   1   0   0</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>1   1   1   1</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>1   1   1   1</td>
</tr>
</tbody>
</table>

* - "0" indicates that allocation is not possible.
  "1" indicates that allocation is possible.
must be grouped together. These part types require finish boring of a critical diameter \( D_2 \). It is necessary to do precise measurement of these diameters using the only available automatic gauging device before the parts are allowed to leave the system. Since the device can be mounted on or near any one machine, these operations must be allocated to the same machine.

The total number of parts to be maintained in the system at any given time is 20. The lower and upper bounds on the number of each type of part in the system are given in Table 8. These bounds usually result from the fixture availability and the downstream production schedules of each part type in the plant. Another restriction may result due to the sharing of common fixture elements for the parts as explained in section 5.5.

It is possible that some of the common fixture elements are available in limited quantities, thereby limiting the total number of those parts in the system, which need such common fixture elements. Table 9 gives such restrictions assumed for this problem.

The equations of the proposed formulations were formed using all the above data. The equations of the objective functions and the constraints are given in Appendix-C. Many of the equations include products of 0-1 integer variables. The strategies available for linearizing these equations are discussed in the following section.
TABLE - 8

UPPER & LOWER BOUNDS ON THE NUMBER OF PARTS OF EACH TYPE IN THE SYSTEM

<table>
<thead>
<tr>
<th>PART NO.</th>
<th>LOWER BOUND</th>
<th>UPPER BOUND</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>12</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>4</td>
</tr>
</tbody>
</table>

TABLE G

AVAILABILITY DATA FOR COMMON FIXTURE ELEMENTS

<table>
<thead>
<tr>
<th>FIXTURE ELEMENT 'CF'</th>
<th>AVAILABILITY 'CF'</th>
<th>PART TYPE USING 'CF'</th>
<th>USAGE PER PART ((b_i)CF)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>15</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>
6.2 Linearization Strategies

The objective functions and the constraints of the proposed formulations involve product terms of 0-1 integer variables. (The expression "product terms" will, hereafter, refer to the products of 0-1 integer variables). Most of the commercially available integer programming codes require the problem to be in linear form. Therefore, it is necessary to use a linearization strategy to produce a completely linear formulation of the problem. Different strategies are reported in the literature for this purpose. The notation used in this section is explained below.

\( \tau(K) \) = Number of elements of a set \( K \).

\( x_j \) = 0-1 integer variables \( (j \in N') \).

\( N' \) = Index set of all 0-1 integer variables occurring in all product terms in the formulation.

\( P_T \) = Set of all product terms occurring in the formulation.

\( P_L \) = Index set of all linearization variables in the formulation.

\( S'g \) = Index set of 0-1 integer variables occurring in a given product term \( g \), \( (g \in P_L) \).
\[ n_g = \text{Number of 0-1 integer variables occurring in a given product term } g, \ (g \in P_I) \]
\[ Z'g = \text{The 0-1 linearization variable replacing a given product term } g, \ (g \in P_I) \]
\[ Zg = \text{The continuous linearization variable replacing a given product term } g, \ (g \in P_I) \]

Example 1:
Consider the product terms \( x_1 \cdot x_2 \cdot x_3 \), \( x_1 \cdot x_2 \cdot x_4 \), \( x_2 \cdot x_3 \). The linearization variables replacing these product terms would be,
\[ Z_1 = Z'_1 = x_1 \cdot x_2 \cdot x_3 \]
\[ Z_2 = Z'_2 = x_1 \cdot x_2 \cdot x_4 \]
\[ Z_3 = Z'_3 = x_2 \cdot x_3 \]
\[ P_I = \{ 1,2,3 \}; \ N' = \{ 1,2,3,4 \}; \]
\[ P_I = \{ (x_1 \cdot x_2 \cdot x_3), (x_1 \cdot x_2 \cdot x_4), (x_2 \cdot x_3) \} \]
For the product term \( x_1 \cdot x_2 \cdot x_4 \),
\[ g = 2; \ S'g = S'_2 = \{ 1,2,4 \} \]

6.2.1 Linearization strategy 1:

Watters(1967) has proposed the following basic linearization strategy for products of 0-1 integer variables.

(i) Replace each product term \((x_j)^k, \ (k>0)\), by \(x_j\)
(ii) Replace each product term $g$ by a new 0–1 linearization variable $Z'g$ with the addition of the following set of constraints for all product terms $g \in P_I$.

\[ \sum_{j \in S'_g} x_j - Z'g \leq (n_g - 1) \quad \text{for all } g \in P_I \quad \ldots \quad (15) \]

\[ - \sum_{j \in S'_g} x_j + n_g Z'g \leq 0 \quad \text{for all } g \in P_I \quad \ldots \quad (16) \]

$(Z'g = 0,1; x_j = 0,1$ for all $j \in S'_g$, for all $g \in P_I$)

Example:

To linearize the term $x_1x_2x_3$, replace it in the formulation by a 0–1 linearization variable $Z'_1$ and add the following constraints,

\[ x_1 + x_2 + x_3 - Z'_1 \leq (3 - 1) \]

\[ -(x_1 + x_2 + x_3) + 3Z'_1 \leq 0 \]

$Z'_1 = 0,1; x_1, x_2, x_3 = 0,1$.

6.2.2 Linearization strategy 2:

The introduction of an additional 0–1 integer variable in strategy 1 can be avoided by the following strategy proposed by Glover and Woolsey (1974).

Replace each product term $g$ by a new continuous linearization variable $Z_g$ with the addition of the following set of constraints for each product term $g$.

\[ \sum_{j \in S'_g} x_j - Z_g \leq (n_g - 1) \quad \text{for all } g \in P_I \quad \ldots \quad (17) \]

\[ x_j \geq Z_g \quad \text{for all } j \in S'_g \quad \text{for all } g \in P_I \quad \ldots \quad (18) \]

$Z_g \geq 0; x_j = 0,1$ for all $j \in S'_g$, for all $g \in P_I$. 

Example

To linearize the term $x_1 \cdot x_2 \cdot x_3$, replace it by a continuous linearization variable $z_1$ and add the following constraints,

$$x_1 + x_2 + x_3 - z_1 \leq 3 - 1$$

$$x_1 \geq z_1$$
$$x_2 \geq z_1$$
$$x_3 \geq z_1$$
$$z_1 \geq 0; x_1 \cdot x_2 \cdot x_3 = 0, 1$$

6.2.3 Linearization strategy 3:

A modification of strategy 2 has been proposed [Glover and Woolsey, 1974, Goldman, 1983] to reduce the number of additional linearization constraints as follows.

Let $m_j = \text{number of product terms containing variable } x_j$ 

$(j \in N')$ in the formulation.

$P_j = \text{Index set of linearization variables}$

$\text{corresponding to the } m_j \text{ product terms containing variable } x_j$ $(j \in N')$ in the formulation.

Replace each product term $g$ $(g \in P_j)$ by a continuous linearization variable $z_g$ and add constraints given by Eq. (17). Replace the entire constraint set given by Eq. (18); by the following,

$$m_j \cdot x_j \geq \sum_{g \in P_j} z_g \quad \text{for all } j \in N'$$

$$0 \leq z_g \leq 1; x_j = 0, 1 \text{ for all } g \in P_j \text{ for all } j \in N'.$$
**Example:**

Consider the product terms \((x_1 \cdot x_2 \cdot x_3), (x_2 \cdot x_4)\) and \((x_1 \cdot x_3 \cdot x_4)\).

- \(N' = \{1, 2, 3, 4\}; \ P_1 = \{1, 2, 3\}\)
- \(m_1 = 2; \ m_2 = 2; \ m_3 = 2; \ m_4 = 2\)
- \(P_1 = \{1, 3\}; \ P_2 = \{1, 2\}; \ P_3 = \{1, 3\}; \ P_4 = \{2, 3\}\)

Replace all product terms by the respective linearization variables \(z_g\) \((g \in P_1)\) as follows,

- \(z_1 = x_1 \cdot x_2 \cdot x_3\)
- \(z_2 = x_2 \cdot x_4\)
- \(z_3 = x_1 \cdot x_3 \cdot x_4\)

and add the constraints,

- \(x_1 + x_2 + x_3 - z_1 \leq (3-1)\)
- \(x_2 + x_4 - z_2 \leq (2-1)\)
- \(x_1 + x_3 + x_4 - z_3 \leq (3-1)\)
- \(2x_1 \geq z_1 + z_3\)
- \(2x_2 \geq z_1 + z_2\)
- \(2x_3 \geq z_1 + z_3\)
- \(2x_4 \geq z_2 + z_3\)

\(0 \leq z_g \leq 1\) for \(g = 1, 2, 3\); \(x_1, x_2, x_3, x_4, x_5 = 0, 1\)

**6.2.4 Linearization strategy 4:**

The number of additional linearization constraints can be reduced in yet another way [Glover and Woolsey, 1973].

Consider a set \(K_v\) of product terms all of which have a 0-1
variable \( x_v \) (\( v \in N' \)) in common and \( \tau(S'g) = 2 \) for all \( g \in K_v \) (i.e., products of two zero-one integer variables).

Let,

- \( N_{K_v} \) = Index set of all 0-1 variables \( x_j \) appearing in set \( K_v \).
- \( P_{K_v} \) = Index set of linearization variables replacing the product terms of set \( K_v \) [\( \tau(K_v) = \tau(P_{K_v}) \)].

Replace each product term in set \( K_v \) by a continuous linearization variable \( Z_g \) (\( g \in P_{K_v} \)) and add the constraints given by Eq. (18) for all \( j \in S'g \), \( g \in P_{K_v} \).

Also, add the following constraint.

\[
\tau(K_v) \cdot x_v + \sum_{j \in N_{K_v}} x_j - \sum_{g \in P_{K_v}} Z_g \leq \tau(K_v) \quad \text{ .... (20)}
\]

\( 0 \leq Z_g \leq 1; x_j = 0, 1 \) for all \( j \in S'g \); for all \( g \in P_{K_v} \).

By choosing different values of \( v \) (\( v \in N' \)) it may be possible to linearize all product terms \( g \) of size \( \tau(S'g) = 2 \) in the formulation.

[It may be remembered that it is sufficient for a given product term to be considered in only one set \( K_v \)]. Since the objective functions of the operation allocation formulations introduce several product terms of two 0-1 integer variables, the reduction in the number of linearization constraints achieved by this strategy, as compared to strategy 1, can be significant.
Example:

Consider the product terms, \( x_1 \cdot x_2, x_1 \cdot x_3 \text{ and} \ x_1 \cdot x_4 \).

Let \( v=1 \). So, \( K_v = \{ (x_1 \cdot x_2), (x_1 \cdot x_3), (x_1 \cdot x_4) \} \).

\( \tau(K_v) = 3; \ N_{K_v} = \{1,2,3,4\} \).

Let \( Z_1 = x_1 \cdot x_2, \ Z_2 = x_1 \cdot x_3, \text{ and} \ Z_3 = x_1 \cdot x_4 \). So, \( P_{K_v} = \{1,2,3\} \).

Add constraints given by Eq. (18), viz.,

\[
\begin{align*}
\quad x_1 & \geq Z_1 \\
\quad x_1 & \geq Z_2 \\
\quad x_1 & \geq Z_3 \\
\quad x_2 & \geq Z_1 \\
\quad x_3 & \geq Z_2 \\
\quad x_4 & \geq Z_3
\end{align*}
\]

and add the constraint given by Eq. (20), viz.,

\[
3x_1 + (x_2 + x_3 + x_4) - (Z_1 + Z_2 + Z_3) \leq 3
\]

\( 0 \leq Z_g \leq 1 \text{ for } g=1,2,3; \ x_j = 0,1 \text{ for } j=1,2,3,4. \)

6.2.5 Linearization strategy 5:

It is possible to combine strategies 3 and 4 to reduce the number of additional linearization constraints as follows.

Use Eq. (20) for all product terms \( g \) of size \( \tau(S'g) = 2, \ (g \in P_1). \)

Use Eq. (17) for all product terms \( g \) of size \( \tau(S'g) > 2, \ (g \in P_1). \)

Use Eq. (19) for all product terms \( g, \ (g \in P_1), \)

\( 0 \leq Z_g \leq 1; \ x_j = 0,1 \text{ for all } g \in P_1, \text{ all } j \in S'g. \)
6.2.6 Linearization of the proposed formulations:

All the strategies discussed above were applied to the equations of formulations except strategy 1. This strategy introduces one additional 0-1 integer variable for each product term in the formulations. The difficulty of integer programming typically depends more strongly on the number of integer variables than on the number of continuous variables [Glover and Woolsey, 1974]. This conclusion was verified by formulating and solving several sample problems and strategy 1 was not pursued later, since it always resulted in a large number of 0-1 integer variables and excessive computational time for solution on the computer.

All the remaining linearization strategies were successfully implemented for the proposed formulations. The number of additional continuous variables introduced by all the strategies is the same, however, the number of additional linearization constraints varies depending on the strategy employed. Strategy 2 produced the maximum number of linearization constraints and strategy 5 produced the minimum. As an indication of the sizes of formulations produced by different strategies, a summary of the sizes of one of the formulations, resulting from each of the strategies is given in Table 10, using the data of section 6.1. The computational implications of these strategies are discussed in section 6.4.
<table>
<thead>
<tr>
<th>STRATEGY NUMBER</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Decision Variables ((X_{i,j,k}))</td>
<td>38</td>
<td>38</td>
<td>38</td>
<td>38</td>
</tr>
<tr>
<td>Linearization variables ((Z_g))</td>
<td>76</td>
<td>76</td>
<td>76</td>
<td>76</td>
</tr>
<tr>
<td>Basic formulation constraints</td>
<td>21</td>
<td>21</td>
<td>21</td>
<td>21</td>
</tr>
<tr>
<td>Linearization constraints</td>
<td>251</td>
<td>113</td>
<td>222</td>
<td>84</td>
</tr>
</tbody>
</table>
The part mix determination problem is formulated as a general integer programming problem. All the proposed equations are linear except the objective function, minimizing the workload imbalance, given by Eq. (8). This nonlinear objective function can be first converted to a function of 0-1 integer variables by using the fact that an integer variable with finite upperbound can be uniquely expressed as a function of a finite number of 0-1 integer variables. Specifically, if an integer variable \( X_i \) is upperbounded such that,

\[
0 \leq X_i \leq u \quad \text{where,} \quad 2^{d-1} < u \leq 2^d, \quad d \text{ is integer}
\]

then \( X_i \) can be expressed uniquely as,

\[
X_i = \sum_{k=1}^{d} 2^k y_k
\]

where, \( y_k = 0,1 \) for \( k=0,1,2,...,d \).

Using the upperbound (UB\(_i\)) as the value of ‘d’ for the respective decision variables \( X_i ,(i=1,...,R) \), the Eq. (8) can be converted into a function of 0-1 integer variables. It can, then, be linearized by using one of the linearization strategies proposed earlier.

6.3 Optimal Solutions:

The formulations given in Appendix-C were linearized and solved using the integer programming routine of the software package SAS/OR (Version 5) on an IBM 4381.
computer. Table 11 gives the final size of the formulations which resulted by using linearization strategy 1. The optimal solutions of all formulations are given in Table 12.

The optimal solutions given above are for a specific set of FMS parameters described in section 6.1. It is possible to examine the changes in the allocations resulting from the changing input parameters.

Under normal circumstances the loading and unloading distances of machines from the central parts storage will not change for a given setup of machines. However, such changes are possible for a system in which there are several qualified machines available for processing a set of parts but only a given number of these machines are to be chosen for allocating the operations of the selected set of part types. There will be a finite number of combinations of such machines which can be chosen. In such cases it may be necessary to attempt solutions of the operation allocation problem by minimizing the transport load for each such combination of machines.

As an example, consider the distances $L_j$ and $U_j$ ($j=1,2,3,4$) which are different from those given in Table 5. The optimal solution to the operation allocation formulation, obtained by keeping all the other parameters the same, is given in Table 13, which has a different pattern of allocations than the one in Table 12.
### TABLE 11

**SIZES OF FORMULATIONS USING LINEARIZATION STRATEGY 2**

#### OPERATION ALLOCATION PROBLEM

<table>
<thead>
<tr>
<th></th>
<th>MINIMIZE TRANSPORT</th>
<th>MINIMIZE REFIXTURING</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. of 0-1 (X_{i,j,k})</td>
<td>38</td>
<td>38</td>
</tr>
<tr>
<td>No. of linearization variables</td>
<td>76</td>
<td>104</td>
</tr>
<tr>
<td>Unique Allocation constraints</td>
<td>14</td>
<td>14</td>
</tr>
<tr>
<td>Grouped Operation constraints</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Scarce Tools constraints</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Tool Magazine Capacity constraints</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Linearization constraints [Eq. (17)]</td>
<td>76</td>
<td>104</td>
</tr>
<tr>
<td>Linearization constraints [Eq. (18)]</td>
<td>175</td>
<td>231</td>
</tr>
</tbody>
</table>

#### PART MIX DETERMINATION PROBLEM

<table>
<thead>
<tr>
<th></th>
<th>MINIMIZE WORKLOAD</th>
<th>MINIMIZE TRANSPORT</th>
<th>MINIMIZE REFIXTURING</th>
</tr>
</thead>
<tbody>
<tr>
<td>General integer variables</td>
<td>0</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>0-1 variables</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Linearization variables</td>
<td>0</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Basic constraints</td>
<td>273</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Linearization constraints</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

"*" - Nonlinear problem, hence additional variables and constraints are required for linearization.
TABLE - 12
OPTIMAL SOLUTIONS OF OPERATION ALLOCATION PROBLEM

<table>
<thead>
<tr>
<th>PART</th>
<th>OPN. No.</th>
<th>MINIMIZE TRANSPORT</th>
<th>MINIMIZE REFIXTURING</th>
</tr>
</thead>
<tbody>
<tr>
<td>i</td>
<td>k</td>
<td>1 2 3 4</td>
<td>1 2 3 4</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1 0 0 0</td>
<td>1 0 0 0</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>0 0 0 1</td>
<td>0 0 1 0</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>0 0 0 1</td>
<td>0 0 0 1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0 0 1 0</td>
<td>0 0 0 1</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>0 1 0 0</td>
<td>0 1 0 0</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>0 1 0 0</td>
<td>0 1 0 0</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>0 0 1 0</td>
<td>0 0 0 1</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>0 0 1 0</td>
<td>0 0 0 1</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>0 1 0 0</td>
<td>0 1 0 0</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>0 1 0 0</td>
<td>0 1 0 0</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>0 1 0 0</td>
<td>0 1 0 0</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>1 0 0 0</td>
<td>1 0 0 0</td>
</tr>
</tbody>
</table>

Note: "1" indicates an allocation.

DISTANCE TRAVELED BY EACH PART TYPE & REFIXTURING LOAD FOR EACH PART TYPE BASED ON THE OPTIMAL OPERATION ALLOCATION

<table>
<thead>
<tr>
<th>PART TYPE</th>
<th>DISTANCE</th>
<th>REFIX. LOAD</th>
</tr>
</thead>
<tbody>
<tr>
<td>i</td>
<td>D_i met.</td>
<td>(FX)_i units</td>
</tr>
<tr>
<td>1</td>
<td>90</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>140</td>
<td>13</td>
</tr>
<tr>
<td>3</td>
<td>120</td>
<td>6</td>
</tr>
<tr>
<td>4</td>
<td>160</td>
<td>15</td>
</tr>
</tbody>
</table>

OPTIMAL SOLUTIONS OF PART MIX DETERMINATION PROBLEM

<table>
<thead>
<tr>
<th>TYPE OF OBJ. FUNCTION</th>
<th>X_1</th>
<th>X_2</th>
<th>X_3</th>
<th>X_4</th>
</tr>
</thead>
<tbody>
<tr>
<td>MINIMIZE WORKLOAD IMBALANCE</td>
<td>9</td>
<td>4</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>MINIMIZE TRANSPORT ACTIVITIES</td>
<td>9</td>
<td>2</td>
<td>8</td>
<td>1</td>
</tr>
<tr>
<td>MINIMIZE REFIXTURING ACTIVITIES</td>
<td>9</td>
<td>2</td>
<td>8</td>
<td>1</td>
</tr>
</tbody>
</table>
### TABLE - 13

**OPTIMAL SOLUTIONS OF OPERATION ALLOCATION PROBLEM (MINIMIZE TRANSPORT) EFFECT OF LOCATION OF MACHINES**

<table>
<thead>
<tr>
<th>MACHINE NO. (j)</th>
<th>$L_j$ (met.)</th>
<th>$U_j$ (met.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>30</td>
<td>30</td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>3</td>
<td>25</td>
<td>25</td>
</tr>
<tr>
<td>4</td>
<td>20</td>
<td>20</td>
</tr>
</tbody>
</table>

### TABLE - 14

**PART NO.** | **OPERATION** | **MACHINE (j)** |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$i$</td>
<td>$k$</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>0</td>
</tr>
</tbody>
</table>

*Note: "1" indicates an allocation.*
When the machines are equidistant from the central storage (i.e., when \( L_j = k1 \) constant, \( U_j = k2 \) constant, for \( j = 1, \ldots, M \)), the objective of minimization of transport activities is similar to one of the allocation objectives defined by Stecke and Solberg (1979), viz., minimization of part visits to machines, except for the possibility of additional part travel due to refixturing which we have incorporated in the objective function by the terms \((ES)_i, i = 1, \ldots, R\), (see section 5.2.2).

The available tool magazine capacities obviously influence the allocation of operations. It may be possible to marginally change the available tool capacity of one or more tool magazines by temporarily unloading some tools not required by the current mix of parts. Under normal circumstances, the tool magazines may not be purged of all tools not required immediately since this may result in greater down time of machines for loading and unloading of tools. This will be necessary when, for example, the operation allocation is indicated to be infeasible due to tight tool magazine capacity constraints. Table 14 gives the values of tool magazine capacities available (which are marginally different from those in Table 5), and the optimal operation allocation using those values. Comparing it with Table 12, the difference in the allocation is evident.
### TABLE - 14

**OPTIMAL SOLUTIONS OF OPERATION ALLOCATION PROBLEM (MINIMIZE TRANSPORT) EFFECT OF TOOL MAGAZINE CAPACITIES**

<table>
<thead>
<tr>
<th>MACHINE NUMBER (j)</th>
<th>TOOL MAGAZINE CAPACITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>j</td>
<td>t_j slots</td>
</tr>
<tr>
<td>1</td>
<td>25</td>
</tr>
<tr>
<td>2</td>
<td>28</td>
</tr>
<tr>
<td>3</td>
<td>20</td>
</tr>
<tr>
<td>4</td>
<td>35</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PART NO.</th>
<th>OPERATION</th>
<th>MACHINE (j)</th>
</tr>
</thead>
<tbody>
<tr>
<td>i</td>
<td>k</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>0</td>
</tr>
</tbody>
</table>

**Note:** "1" indicates an allocation.
6.4 Computational Experience:

A computer program has been written to prepare the proposed formulations in the form acceptable to the SAS/OR package. This program accepts all the relevant parameters of the FMS as inputs and produces the formulation as an input file for the package. It is, thus, possible to quickly generate formulations for different FMS configurations. Several sample formulations were thus created and solved using this software. The following discussion of the computational behaviour of the proposed formulations is based on the results of such investigations.

Most commercially available integer programming (IP) codes use the Branch and Bound (B&B) method for the solution of a linear integer program (e.g., SAS/OR, MPSX-MIP, LINDO, etc.). However, for a given formulation, the computational behaviour of each software package can vary markedly due to the inherent flexibility of the B&B approach, viz., the strategy of search of the Branch and Bound tree. A judicious choice of the strategy for searching the B&B tree can substantially reduce the number of computations involved. Many formulations involving up to 50 zero-one integer variables yielded optimal solutions in less than 25 to 30 minutes on the IBM 4381 computer. For smaller problems with 25 to 30 zero-one variables the time
required was 2 to 5 minutes. Given that a formulation has feasible solutions, an important and easily verifiable parameter of the efficiency of an IP code is the time required to find the optimal solution to the formulation. It is a known fact that usually the time required to solve an integer program is primarily dependent on the number of integer variables in the formulation [Garfinkel and Nemhauser, 1972; Taha, 1975]. It is, of course, difficult to generalise the experience obtained with the SAS/OR package and the IBM 4381 computer to other packages and computers because the required time can vary widely depending on the B&B search strategy and the speed of the computer.

However, it may be possible to estimate the physical dimensions of an actual FMS planning problem that would yield computationally manageable formulations using a given IP code and the given computer. For example, the size of the FMS and the associated resources corresponding to the formulations involving upto fifty 0-1 integer variables may be estimated as follows.

The maximum number of zero-one integer variables in the operation allocation formulations is,

\[ R = \left( \sum_{i=1}^{n_B} q_i \right) \cdot M \]

\[ = n_B \cdot M \]
Based on the surveys of the current FMS installations [Kochan, 1985; Mortimer, 1984], it is apparent that a medium size FMS would consist of 4 to 6 machines. If each operation of each part type can be performed on any one of the machines then the formulations for allocating about 10 different operations (n_B = 10) in a medium size FMS would involve approximately 50 zero-one integer variables. However, there are technological and physical considerations which often prevent an operation to be performed on some of the machines. Thus the actual 'possibilities' of allocation of operations are often significantly less than the maximum possible (e.g., see Table 7). The distinction between an 'operation' and a processing step, as explained in section 4.2, must be remembered here. Since one operation can, and usually will, accommodate a large number of processing steps, it is common to have 2 to 5 operations per part in an FMS [Mortimer, 1984]. So, the formulations resulting from a problem involving a medium FMS simultaneously processing 2 to 5 different part types may be expected to yield integer programs with as many as fifty 0-1 integer variables. The length of time which may be considered as reasonable would vary in each production situation. However, it must be pointed out here that the operation allocation and part mix determination problems are not likely to be posed once in every hour or so in the real life situations. The set of
part types, being processed simultaneously, can normally be expected to change after a few shifts, necessitating a fresh solution to these two planning problems. This, of course, does not reduce the importance of obtaining quick solutions to these problems, but indicates that it may not be vital to obtain such solutions in minutes or seconds in a real life situation.

Since the number of integer variables in the two linear formulations of the part mix determination problem is small (=R), these two formulations (obj. functions given by Eqs. (9) and (10)) would not pose computational problems for medium, or even larger, FMS configurations. The nonlinear formulation of the part mix determination problem, given by Eq. (8), however, requires zero-one integer variables for linearization as explained in section 6.3. The number of zero-one variables replacing a variable \( X_i \) depends on the upperbound, \( (UB)_i \), on the value of \( X_i \). This upperbound, in turn, depends on the availability of fixtures etc. In cases where the number of parts of a given type 'i' is not constrained by such considerations, the logical upperbound may be the size of the common parts storage. Thus, the number of zero-one integer variables required for the only nonlinear formulation of the part mix determination problem will depend on the availability of fixtures and/or the size of the central parts store. It is realistic to expect between 20 to 30 parts in the system at
any time for a medium size FMS, thereby fixing the number of zero-one variables in this formulation of the part mix problem in the region of 25 to 30, which appears to be computationally manageable. Formulations for larger systems will, however, need more such variables and the problems of excessive computational time associated with IP codes, discussed before, will be applicable to this particular nonlinear formulation of the part mix problem. The constraint set of the part mix formulations is small and is not expected to be computationally significant for an IP code.

The actual number of zero-one integer variables in a given operation allocation formulation can be reduced by retaining one of the variables appearing in each of the constraints, given by Eq. (3), as a free variable (which ordinarily would be a zero-one variable). Since all the other variables appearing in that particular constraint are declared as zero-one integer variables, the variable which is not declared as integer is forced to assume values of 0 or 1 due to the equality.

Another factor which affects the size of the operation allocation formulations is the number of common tools between two or more operations to be allocated. The constraints given by Eq. (7b) cover all such possibilities, generating one product term of zero-one integer variables for each combination of operations sharing a common tool.
Obviously, a formulation with more number of common tools will contain more product terms than the one with less number of common tools, all other formulation parameters remaining the same. A larger number of nonlinear terms in the formulations would require a larger number of additional linearization variables and constraints. This would increase the size of the LP problem that the IP algorithm would have to solve at each node of the B&B tree, thereby increasing the computational time. Such tool sharing is quite common in the case of small tools such as drills, typically occupying one slot in the tool magazine. For a given set of part types, it may be possible to ignore such small tools for the purpose of formulating the problem, thereby preventing addition of some nonlinear terms in the formulations, provided that enough space is set aside for such tools in each tool magazine in advance. This may not be difficult since these tools would require only a few slots in all.

The number of additional linearization constraints introduced in the formulation is dependent on the choice of the linearization strategy. All the strategies given in section 6.2 were applied to sample problems and the effect of different linearization strategies on the computational time was observed. The use of strategy 2 and strategy 5 always yielded the maximum and minimum number of linearization constraints, respectively. However, the
computational time required for the solution of the same problem formulated using different strategies was not consistent with the sizes of the formulations generated by these strategies. It was observed that the operation allocation formulation generated using strategy 2 required less computational time for solution than the one generated using strategy 5. This was explained after examining the B&B tree generated by the IP algorithm for both formulations. It was observed that the use of these two strategies resulted in a completely different B&B tree search by the IP algorithm to reach the optimal solution, in spite of specifying the same general strategy\textsuperscript{\dagger} for exploring the tree. This difference was due to the fact that the partially relaxed LP solutions\textsuperscript{\ddagger} at the nodes of the B&B tree were different for the two formulations. This is possible since the two equivalent IP formulations need not yield the same LP solutions when the integrality constraints are removed. As a result, the choice of branching variables was different for the two formulations of the same problem. The search path for the formulation using strategy 5 always resulted in exploration of more

\textsuperscript{\dagger} - For selection of branching variables and the branches to be explored. Refer to SAS/OR User's Manual, Version 5.

\textsuperscript{\ddagger} - LP solutions obtained at each node of the B&B tree by relaxing the integrality conditions on the appropriate 0-1 integer variables and by adding appropriate bounds as required by the B&B method.
nodes of the B&B tree than those explored by using strategy 2, although the time required to explore one node was less for the formulation using strategy 5 due to the reduced size of the problem. Thus, the computational advantage of faster exploration of one node was compensated by the greater number of nodes explored. The time required for solving the formulations using strategies 3 and 4 was comparable to that required by the formulations using strategies 5 and 2 respectively. The difference in the times using strategy 2 or 4 was marginal. Hence, linearization strategies 2 and 4 resulted in faster solutions. This difference was also observed in the case of the only nonlinear formulation of the part mix problem but the margin of difference was smaller.

The strategy of exploring the B&B tree is really made up of two separate strategies, selection of the branching variable and selection of the next node to be explored, at each node of the tree. The SAS/OR package offers some choices in the selection of these two strategies which were explored during the investigations. The LIFO rule for selecting the next node was found to be superior than the others, such as FIFO, smallest objective value, etc. The branching variables selected according to the largest (or smallest) sum of integer infeasibilities in the partially relaxed LP problems at the nodes resulted in
faster solutions than the other choices, viz., objective function penalties, pseudocosts etc.

It is often helpful to identify a tight bound on the optimal solution of an integer program in order to accelerate the search of the B&B tree by discarding nonpromising branches. The resulting reduction in the computational time is desirable for large formulations. A common way of identifying such bound is by locating a good feasible solution prior to solving the problem on computer. The identification of a feasible solution may not be easy, especially when the constraints are many and complex.

Considering the operation allocation formulations, the difficulty in identifying feasible solutions is due to the tool magazine capacity constraints and the tool availability constraints. It was observed that the solution of the formulation required much less time if the tool magazine capacity constraints were omitted from the formulation. This fact can be made use of, at least in some cases, to quickly identify good feasible solutions for large formulations, which are expected to take large amount of time for solution.

A given formulation may be solved without the tool magazine capacity constraints. The resulting optimal solution may or may not be feasible with respect to the available tool magazine capacities. If such truncated formulation does yield an optimal solution which is
feasible with respect to the available tool magazine capacities then it is obviously the true optimal solution and the complete formulation need not be solved. If, however, such solution is not feasible with respect to the available tool magazine capacities then it can be used to locate such feasible solution. This requires rearranging the operation allocation of the truncated solution manually in order to satisfy the tool magazines capacity constraints. Such manipulation has proved simple for many of the sample problems, and reasonably tight upperbounds were established in this way. However, the tightness of the bound, thus generated, is not guaranteed in every case. For a large formulation of operation allocation (minimization of transport activities only) it may be possible to divide the problem into two sub-problems, for locating a good feasible solution to act as an upper bound, as follows.

(a) Formulate and solve the problem by considering only those part types which figure in the following two types of constraints.

(i) Grouped operations constraints [Eq. (4b)].

(ii) Tool availability constraints [Eq. (7a)].

Determine the tool magazine usage for all machines from the resulting optimal solution. Determine the spare capacity available in the tool magazines with this allocation.
(b) Formulate and solve the problem again by considering only the part types not considered in (a) and with the available tool magazine capacities = spare tool magazine capacities determined in (a).

If both (a) and (b) yield optimal solutions then the optimal solutions in (a) and (b) together form a feasible solution to the complete problem. This strategy has yielded good feasible solutions for many sample problems. However, as before, it may not work in every case. Also, it would not be worth attempting this strategy if the number of zero-one integer variables in both (a) and (b), individually, is not significantly less than the total number of zero-one integer variables in the formulations, since the time required to locate a feasible solution may then be comparable to the time required to solve the complete formulation itself.
Chapter VII

SUMMARY

Two pre-release planning problems of flexible manufacturing systems have been modelled, viz., the operation allocation problem and the part mix determination problem. The proposed formulations were applied to a realistic planning situation and the optimal solutions were obtained using the integer programming routine of the SAS/OR package. Several hypothetical configurations were modelled and solved to examine the computational aspects of the proposed formulations.

The solutions of these planning problems are essential to model the dynamic behaviour of the FMS (e.g., by simulating the actual performance of the system). The results of such dynamic modelling may necessitate manipulation of the pre-release planning decisions in order to meet the target performance levels of the FMS. The availability of alternate formulations of the pre-release planning problems can offer an increased number of systematic ways by which such manipulations can be attempted. The contribution of this research to the field of optimal planning of FMS is the new formulations of the
above two pre-release planning problems. Also, the incorporation of realistic constraints such as the grouped operations constraints, the limited tool availability constraints, and the standard fixture elements availability constraints has served to expand the constraint set of the pre-release problems, which can be used with other similar formulations.

The proposed formulations can, at least theoretically, be applied to both small and large FMS's. The time required to locate the optima will, however, depend on several factors such as, the number of integer variables in the formulations, the solution strategy used by the IP code, the speed of the computer, etc. The work reported here may be extended to incorporate additional system constraints specially applicable to a given system being planned. The basic approach for modelling the constraints can remain the same. The formulations of the part mix determination problem can be made more realistic by incorporating the vital performance measures of the system, such as utilization of machines or output rate of the system, in the objective function. Such modelling would, most probably, require incorporation of the analytical results of the queueing networks, expressed as functions of the decision variables of the part mix determination problem. Also, it is necessary to investigate the solution methodologies required to solve such planning problems when
the parts do not arrive at the system in well defined
batches but randomly or in very small batches. The FMS is,
perhaps, the most appropriate method of production in such
situations and the methodologies to solve the planning
problems may have to be revised completely to take into
account the stochastic nature of the production
environment. It may also be worth investigating how the
different objectives proposed can be considered together to
form a multi-objective optimization problem. Finally, the
computational behaviour of the proposed formulations needs
to be investigated further in order to reduce the time
required to obtain the optimal solutions to the large 0-1
mixed linear integer programs by devising special branch
and bound search strategies or by employing other methods
of solving the linear integr programs such as the cutting
planes methods.
REFERENCES


Kochan A., (1985), FMS is for the BIG Boys in the U.S.,

Kusiak A., (1983), Loading Models in Flexible Manufacturing
Systems, Proceedings of VIIIth International
Conference on Production Research, Windsor,
Canada.

Engineers, IFS (Publications) Ltd., U.K.

Nof S.Y., Barash M.M., Solberg J.J., (1979), Operational
Control of Item Flow in Versatile Manufacturing
Systems, International Journal of Production
Research, Vol 17, No. 5, 479-489.

Ranky P.G., (1983), The Design and Operation of FMS,
IFS (Publications) Ltd., North Holland Publishing
Company, U.K.

Shanker K., Tzen Y.J., (1985), A Loading and Dispatching
Problem in a Random Flexible Manufacturing System,
International Journal of Production Research,

Control Problems in Cellular Manufacture,
International Journal of Production Research,
Vol 22, No. 5, 773-789.

Manufacturing Systems, Proceedings of IVth
International Conference on Production Research,
Tokyo, Japan, 1265-1275.

Stecke K.E., (1983), Formulation and Solution of Nonlinear
Integer Production Planning Problems for Flexible

Stecke K.E., (1984), Design, Planning, Scheduling and
Control Problems of Flexible Manufacturing
Systems, Proc. ORSA/TIMS Special International
Conference on FMS: Operations Research Models and
Applications, August, Ann Arbor, Michigan.

Stecke K.E., Solberg J.J., (1979), The Optimal Planning of
Computerised Manufacturing Systems-The CMS Loading
Problem, Report #20, NSF Grant # APR74 15258,
School of Industrial Engineering,
Purdue University, West Lafayette, Indiana.


APPENDIX - A

PART SKETCHES, PROCESS SHEETS AND REFIXTURING INFORMATION
PART #1: SIDE COVER
<table>
<thead>
<tr>
<th>Opn</th>
<th>Operation Description</th>
<th>Tool Number</th>
<th>Slots per Tool</th>
<th>Time (Mins.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>Rough mill bottom face</td>
<td>M1</td>
<td>3</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>Finish mill bottom face</td>
<td>M2</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>02</td>
<td>Rough mill front face</td>
<td>M3</td>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>Finish mill front face</td>
<td>M4</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Drill 8 dia. holes thr. (10)</td>
<td>D1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ream 8 dia. holes (10)</td>
<td>R1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Deburr 8 dia. holes (10)</td>
<td>DB1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Drill 6 dia. holes thr. (8)</td>
<td>D2</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Deburr 6 dia. holes (8)</td>
<td>DB2</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Tap 6 dia. holes thr. (8)</td>
<td>T1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>03</td>
<td>Rough bore dia. D1</td>
<td>B1</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>Finish bore dia. D1 &amp; chamfer</td>
<td>B2</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>
REFIXTURING INFORMATION FOR PART TYPE 1
w(i, k, p, q) for i = 1, k = 2, 3; p = 1, 2, 3, 4; q = 1, 2, 3, 4

PART #: 1  OPERATION #: 2

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

i = 1; k = 2

PART #: 1  OPERATION #: 3

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>x</td>
<td>0</td>
<td>x</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>x</td>
<td>0</td>
<td>x</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>x</td>
<td>2</td>
<td>x</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>x</td>
<td>2</td>
<td>x</td>
<td>0</td>
</tr>
</tbody>
</table>

i = 1; k = 3

\* - F(i, k, p, q) = 1 if w(i, k, p, q) > 0
    = 0 otherwise
w(i, k, p, q) = X if operation (i, k) is not possible on
    machine p or machine q or both.
PART #2: MOTOR HOUSING
<table>
<thead>
<tr>
<th>Ope</th>
<th>Operation Description</th>
<th>Tool Number</th>
<th>Slots per Tool</th>
<th>Time (Mins.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>Rough mill bosses</td>
<td>M6</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>Drill 10 dia. holes thr. (5)</td>
<td>D4</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ream 10 dia. holes (5)</td>
<td>R5</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>02</td>
<td>Rough mill front face</td>
<td>M3</td>
<td>2</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>Finish mill front face</td>
<td>M4</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Rough bore dia. D1</td>
<td>B6</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Finish bore dia. D1 &amp; chamfer</td>
<td>B6</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Rough bore dia. D2</td>
<td>B8</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Finish bore dia. D2</td>
<td>B8</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Mill arc on front face</td>
<td>M7</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>03</td>
<td>Rough mill back face</td>
<td>M3</td>
<td>2</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>Finish mill back face</td>
<td>M4</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Rough bore dia. D3</td>
<td>B9</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Finish bore dia. D3 &amp; chamfer</td>
<td>B9</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>04</td>
<td>Rough mill side mounting face</td>
<td>M1</td>
<td>2</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>Finish mill side mounting face</td>
<td>M2</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Drill 4 dia. holes 10 deep (4)</td>
<td>D3</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Tap 4 dia. holes 8 deep (4)</td>
<td>T2</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>
### Refixturing Information for Part Type 2

\( w(i,k,p,q) \) s for \( i=2; k=2,3,4; p=1,2,3,4; q=1,2,3,4 \)

#### PART #: 2  OPERATION #: 2

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>2</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>3</td>
<td>X</td>
<td>7</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>4</td>
<td>X</td>
<td>7</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

\( i = 2; k = 2 \)

#### PART #: 2  OPERATION #: 3

\( F(2,3,2,2) = 0 \) because,
After operation 2 the worktable can rotate through 180 degrees to do operation 3.
\( F(2,3,2,1) > 0 \) because,
the part can not be rotated by 90 degrees on machine 1 as it interferes with machine parts.

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>0</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>3</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>4</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

\( i = 2; k = 3 \)

#### PART #: 2  OPERATION #: 4

\( F(2,4,1,1), F(2,4,2,1) > 0 \)
reason same as above.

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6</td>
<td>0</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>0</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>3</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>4</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

\( i = 2; k = 4 \)

\( i = F(i,k,p,q) = 1 \) if \( w(i,k,p,q) > 0 \)
\( = 0 \) otherwise
\( w(i,k,p,q) = X \) if operation \( (i,k) \) is not possible on machine \( p \) or machine \( q \) or both.
PART #3 : END COVER
<table>
<thead>
<tr>
<th>Opn</th>
<th>Operation Description</th>
<th>Tool Number</th>
<th>Slots per Tool</th>
<th>Time (Mins.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>Drill 5 dia. holes thr. (4)</td>
<td>D5</td>
<td>1</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>Ream 5 dia. holes (4)</td>
<td>R3</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Rough mill bottom face</td>
<td>M9</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Finish mill bottom face</td>
<td>M10</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Deburr 5 dia. holes (4)</td>
<td>DB3</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Rough mill square edge</td>
<td>M5</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Rough bore dia. D1 &amp; chamfer</td>
<td>B3</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>02</td>
<td>Rough mill top face</td>
<td>M9</td>
<td>2</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>Finish mill top face</td>
<td>M10</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Rough bore dia. D2</td>
<td>B4</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Finish bore dia. D2 &amp; chamfer</td>
<td>B5</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Drill 10 dia. holes thr. (2)</td>
<td>D4</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Enlarge to 15 dia. (2)</td>
<td>D6</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ream 15 dia. holes (2)</td>
<td>R4</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Deburr 15 dia. holes (2)</td>
<td>DB4</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Drill 10 dia. holes thr. (2)</td>
<td>D4</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ream 10 dia. holes (2)</td>
<td>R5</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Deburr 10 dia. holes (2)</td>
<td>DB5</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Rough mill bosses (4)</td>
<td>M6</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Deburr 10 dia. holes (4)</td>
<td>DB3</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>03</td>
<td>Drill 10 dia. holes thr. (2) on periphery</td>
<td>D4</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Enlarge to 20 dia. (2)</td>
<td>D7</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Deburr 20 dia. holes (2)</td>
<td>DB6</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>
REFIXTURING INFORMATION FOR PART TYPE $3^\dagger$

$w(i,k,p,q)$s for $i=3$; $k=2,3$; $p=1,2,3,4$; $q=1,2,3,4$

PART #: 3  OPERATION #: 2

$F(3,2,1,1) = 0$ because,
the part can be rotated by
180 degrees on the work table.

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>

$i = 3$; $k = 2$

PART #: 3  OPERATION #: 3

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4</td>
<td>4</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>4</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

$i = 3$; $k = 3$

$\dagger$ - $F(i,k,p,q) =$ 1 if $w(i,k,p,q) > 0$

= 0 otherwise

$w(i,k,p,q) = X$ if operation $(i,k)$ is not possible on
machine $p$ or machine $q$ or both.
# OPERATION SHEET

Part # 4 : Housing Cover

<table>
<thead>
<tr>
<th>Opn</th>
<th>Operation Description</th>
<th>Tool Number</th>
<th>Slots per Tool</th>
<th>Time (Mins.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>Rough mill front face</td>
<td>M8</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>Rough mill back face</td>
<td>M9</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Finish mill back face</td>
<td>M10</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>02</td>
<td>Drill 10 dia. holes (4)</td>
<td>D4</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>Rough mill bottom face</td>
<td>M1</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Finish mill bottom face</td>
<td>M2</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Deburr 10 dia. holes (4)</td>
<td>DB1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>03</td>
<td>Rough mill top face</td>
<td>M9</td>
<td>2</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td>Finish mill top face</td>
<td>M10</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Rough bore dia. D1</td>
<td>B7</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Finish bore dia. D1 &amp; chamfer</td>
<td>B7</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Rough mill bosses (4)</td>
<td>M6</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Deburr 10 dia. holes (4)</td>
<td>DB1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>04</td>
<td>Rough bore dia. D2</td>
<td>B4</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>Finish bore dia. D2 &amp; chamfer</td>
<td>B5</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>
REFIXTURED INFORMATION FOR PART TYPE 4
\( w(i,k,p,q) \) for \( i=4; k=2,3,4; p=1,2,3,4; q=1,2,3,4 \)

<table>
<thead>
<tr>
<th>PART #: 4</th>
<th>OPERATION #: 2</th>
<th>q</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1 2 3 4</td>
<td></td>
</tr>
<tr>
<td>p</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>5 5 x x</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>5 5 x x</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>x x x x</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>x x x x</td>
<td></td>
</tr>
</tbody>
</table>

\( i = 4; k = 2 \)

<table>
<thead>
<tr>
<th>PART #: 4</th>
<th>OPERATION #: 3</th>
<th>q</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1 2 3 4</td>
<td></td>
</tr>
<tr>
<td>p</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0 0 5 5</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0 0 5 5</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>x x x x</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>x x x x</td>
<td></td>
</tr>
</tbody>
</table>

\( i = 4; k = 3 \)

<table>
<thead>
<tr>
<th>PART #: 4</th>
<th>OPERATION #: 4</th>
<th>q</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1 2 3 4</td>
<td></td>
</tr>
<tr>
<td>p</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>5 5 0 0</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>5 5 0 0</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0 0 5 5</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0 0 5 5</td>
<td></td>
</tr>
</tbody>
</table>

\( i = 4; k = 4 \)

\( F(i,k,p,q) = 1 \) if \( w(i,k,p,q) > 0 \)
\( = 0 \) otherwise
\( w(i,k,p,q)' = x \) if operation \((i,k)\) is not possible on machine \( p \) or machine \( q \) or both.
APPENDIX - B

CUTTING TOOLS INFORMATION.
SUMMARISED TOOL REQUIREMENTS OF OPERATIONS

<table>
<thead>
<tr>
<th>PART</th>
<th>OPERATION</th>
<th>TOOLS REQUIRED</th>
</tr>
</thead>
<tbody>
<tr>
<td>i</td>
<td>k</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>11, 12</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>1, 2, 13, 14, 23, 28, 48, 49</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>36, 37</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>4, 16, 32</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>13, 14, 17, 41, 43</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>13, 14, 44</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>3, 11, 12, 24</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>5, 15, 19, 20, 30, 38, 50</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>4, 6, 16, 19, 20, 31, 32, 39, 40, 50, 51, 52</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>4, 7, 53</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>18, 19, 20</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>4, 11, 12, 48</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>16, 19, 20, 42, 48</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>39, 40</td>
</tr>
</tbody>
</table>
### BASIC CUTTING TOOLS DATA

<table>
<thead>
<tr>
<th>No.</th>
<th>TOOL CODE</th>
<th>TOOL DESCRIPTION</th>
<th>SLOTS PER TOOL</th>
<th>AVAILABILITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>D1</td>
<td>DRILL 8 mm.</td>
<td>1</td>
<td>OK*</td>
</tr>
<tr>
<td>2</td>
<td>D2</td>
<td>DRILL 6 mm.</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>3</td>
<td>D3</td>
<td>DRILL 4 mm.</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>4</td>
<td>D4</td>
<td>DRILL 4 mm.</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>5</td>
<td>D5</td>
<td>DRILL 5 mm.</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>6</td>
<td>D6</td>
<td>DRILL 15 mm.</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>7</td>
<td>D7</td>
<td>DRILL 20 mm.</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>10</td>
<td>M1</td>
<td>MILLING CUTTER-I</td>
<td>3</td>
<td>OK</td>
</tr>
<tr>
<td>11</td>
<td>M2</td>
<td>MILLING CUTTER-II</td>
<td>3</td>
<td>2 NOS.</td>
</tr>
<tr>
<td>12</td>
<td>M3</td>
<td>MILLING CUTTER-III</td>
<td>2</td>
<td>OK</td>
</tr>
<tr>
<td>13</td>
<td>M4</td>
<td>MILLING CUTTER-IV</td>
<td>2</td>
<td>2 NOS.</td>
</tr>
<tr>
<td>14</td>
<td>M5</td>
<td>END MILL-I</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>15</td>
<td>M6</td>
<td>BOSS MILLING CUTTER</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>16</td>
<td>M7</td>
<td>END MILL-II</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>17</td>
<td>M8</td>
<td>MILLING CUTTER-V</td>
<td>2</td>
<td>OK</td>
</tr>
<tr>
<td>18</td>
<td>M9</td>
<td>MILLING CUTTER-VI</td>
<td>2</td>
<td>OK</td>
</tr>
<tr>
<td>19</td>
<td>M10</td>
<td>MILLING CUTTER-VII</td>
<td>2</td>
<td>OK</td>
</tr>
<tr>
<td>20</td>
<td>T1</td>
<td>TAP-I</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>21</td>
<td>T2</td>
<td>TAP-II</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>22</td>
<td>T3</td>
<td>TAP-III</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>23</td>
<td>R1</td>
<td>REAMER 8 mm.</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>24</td>
<td>R2</td>
<td>REAMER 6 mm.</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>25</td>
<td>R3</td>
<td>REAMER 5 mm.</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>26</td>
<td>R4</td>
<td>REAMER 15 mm.</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>27</td>
<td>R5</td>
<td>REAMER 10 mm.</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>28</td>
<td>B1</td>
<td>BORING TOOL-I</td>
<td>3</td>
<td>OK</td>
</tr>
<tr>
<td>29</td>
<td>B2</td>
<td>BORING TOOL-II</td>
<td>3</td>
<td>OK</td>
</tr>
<tr>
<td>30</td>
<td>B3</td>
<td>BORING TOOL-III</td>
<td>2</td>
<td>OK</td>
</tr>
<tr>
<td>31</td>
<td>B4</td>
<td>BORING TOOL-IV</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>32</td>
<td>B5</td>
<td>BORING TOOL-V</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>33</td>
<td>B6</td>
<td>BORING TOOL-VI</td>
<td>3</td>
<td>OK</td>
</tr>
<tr>
<td>34</td>
<td>B7</td>
<td>BORING TOOL-VII</td>
<td>2</td>
<td>OK</td>
</tr>
<tr>
<td>35</td>
<td>B8</td>
<td>BORING TOOL-VIII</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>36</td>
<td>B9</td>
<td>BORING TOOL-IX</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>37</td>
<td>DB1</td>
<td>DEBURRING TOOL-I</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>38</td>
<td>DB2</td>
<td>DEBURRING TOOL-II</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>39</td>
<td>DB3</td>
<td>DEBURRING TOOL-III</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>40</td>
<td>DB4</td>
<td>DEBURRING TOOL-IV</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>41</td>
<td>DB5</td>
<td>DEBURRING TOOL-V</td>
<td>1</td>
<td>OK</td>
</tr>
<tr>
<td>42</td>
<td>DB6</td>
<td>DEBURRING TOOL-VI</td>
<td>1</td>
<td>OK</td>
</tr>
</tbody>
</table>

* - "OK" indicates that the tool is available in sufficient quantity.
APPENDIX - C

EQUATIONS OF FORMULATIONS

(Using the data in section 6.1)
Objective function:

Minimize \( Z \)

\[
\begin{align*}
2(x_1,1,2 \cdot x_1,4,3) & + 2(x_1,2,2 \cdot x_1,4,1) + 2(x_1,3,2 \cdot x_1,2,3) \\
+ 6(x_2,2,2 \cdot x_3,1,3) & + 6(x_2,1,5 \cdot x_3,1,4) \\
+ 3(x_3,2,1 \cdot x_4,3,2) & + 3(x_3,3,1 \cdot x_4,1,2) \\
+ 3(x_3,3,1 \cdot x_4,2,2) & + 3(x_3,3,1 \cdot x_4,4,2) \\
+ 3(x_3,4,1 \cdot x_4,1,2) & + 3(x_3,4,1 \cdot x_4,2,2) \\
+ 3(x_3,4,1 \cdot x_4,4,2) & + 4(x_3,1,2 \cdot x_4,1,3) \\
+ 4(x_3,2,2 \cdot x_4,1,3) & + 4(x_3,2,2 \cdot x_4,2,3) \\
+ 5(x_4,1,2 \cdot x_5,3,4) & + 5(x_4,2,2 \cdot x_5,3,4) \\
+ 5(x_4,1,3 \cdot x_5,1,4) & + 5(x_4,3,3 \cdot x_5,2,4) \\
+ 5(x_4,1,3 \cdot x_5,2,4) & + 5(x_4,3,3 \cdot x_5,3,4) \\
+ 5(x_4,4,3 \cdot x_5,3,4) & + 5(x_4,4,3 \cdot x_5,4,4)
\end{align*}
\]

Subject to,

\[
\begin{align*}
(1) & \quad x_1,1,1 + x_1,2,1 + x_1,4,1 = 1 \\
(2) & \quad x_1,1,2 + x_1,2,2 + x_1,3,2 + x_1,4,2 = 1 \\
(3) & \quad x_2,2,3 + x_1,4,3 = 1 \\
(4) & \quad x_2,3,1 + x_2,4,1 = 1 \\
(5) & \quad x_2,2,2 = 1 \\
(6) & \quad x_2,1,3 + x_2,2,3 = 1 \\
(7) & \quad x_2,1,4 + x_2,2,4 = 1 \\
(8) & \quad x_3,1,1 + x_3,2,1 + x_3,3,1 + x_3,4,1 = 1 \\
(9) & \quad x_3,1,2 + x_3,2,2 + x_3,3,2 + x_3,4,2 = 1 \\
(10) & \quad x_3,1,3 + x_3,2,3 = 1 \\
(11) & \quad x_4,1,1 + x_4,2,1 = 1 \\
(12) & \quad x_4,1,2 + x_4,2,2 = 1 \\
(13) & \quad x_4,1,3 + x_4,2,3 + x_4,3,3 + x_4,4,3 = 1 \\
(14) & \quad x_4,1,4 + x_4,2,4 + x_4,3,4 + x_4,4,4 = 1 \\
(15) & \quad (x_3,1,4 \cdot x_5,1,4) + (x_3,2,4 \cdot x_5,2,4) = 1 \\
(16) & \quad x_1,1,2 + x_1,2,2 + x_1,3,2 + x_1,4,2 + x_3,1,3 + x_3,2,2 \\
+ x_3,2,3 - (x_1,1,2 \cdot x_3,1,3) - (x_1,2,2 \cdot x_3,2,2) \\
- (x_1,2,2 \cdot x_3,2,3) - (x_3,2,2 \cdot x_5,2,3) \\
+ (x_1,2,2 \cdot x_3,2,2 \cdot x_3,2,3) \leq 2 \\
(17) & \quad x_1,1,1 + x_1,2,1 + x_1,4,1 + x_3,1,4 + x_3,2,4 + x_5,1,2 \\
+ x_5,2,2 - (x_1,1,1 \cdot x_3,1,4) - (x_1,1,1 \cdot x_5,1,2) \\
- (x_1,2,1 \cdot x_3,2,4) - (x_1,2,1 \cdot x_5,2,2) \\
- (x_3,2,4 \cdot x_5,2,2) \leq 2
\end{align*}
\]
(18) \[ 6x_{1,1} + 10x_{1,1,2} + 5x_{3,1,3} + 8x_{3,1,4} + 10x_{4,1,1} + 14x_{4,1,2} + 3x_{4,1,3} + 6x_{5,1,1} + 8x_{5,1,3} + 2x_{5,1,4} \]
\[ + 6x_{5,1,3} - (x_{4,1,2}x_{4,1,3}) - 4(x_{1,1,2}x_{3,1,3}) \]
\[ - 6(x_{1,1,1}x_{3,1,4}) - 6(x_{1,1,1}x_{5,1,2}) \]
\[ - 6(x_{3,1,1}x_{5,1,2}) + 6(x_{1,1,1}x_{3,1,2}x_{5,1,2}) \]
\[ - (x_{4,1,2}x_{5,1,2}) - 5(x_{4,1,1}x_{4,1,2}) \]
\[ - 4(x_{4,1,1}x_{5,1,1}) - 4(x_{4,1,1}x_{5,1,3}) \]
\[ - 4(x_{4,1,1}x_{5,1,2}) - 4(x_{4,1,1}x_{5,1,3}) \]
\[ - 5(x_{4,1,2}x_{5,1,1}) - 2(x_{4,1,2}x_{5,1,3}) \]
\[ - (x_{4,1,3}x_{5,1,2}) - 4(x_{5,1,1}x_{5,1,3}) \]
\[ - 4(x_{5,1,2}x_{5,1,3}) - (x_{1,1,2}x_{5,1,1}x_{5,1,3}) \]
\[ + 4(x_{4,1,1}x_{4,1,2}x_{5,1,1}) + 4(x_{4,1,1}x_{4,1,2}x_{5,1,3}) \]
\[ + 4(x_{4,1,1}x_{4,1,2}x_{5,1,1}x_{5,1,3}) \]
\[ - 4(x_{4,1,1}x_{4,1,2}x_{5,1,1}x_{5,1,3}) \]
\[ \leq 15 \]

(19) \[ 6x_{1,2,1} + 10x_{1,2,2} + 6x_{1,2,3} + 9x_{1,2,4} + 5x_{3,2,3} + 8x_{3,2,4} + 10x_{4,2,1} + 14x_{4,2,2} + 3x_{4,2,3} + 6x_{5,2,1} + 8x_{5,2,2} + 8x_{5,2,3} + 2x_{5,2,4} \]
\[ - (x_{4,2,2}x_{4,2,3}) - 4(x_{1,2,2}x_{3,2,2}) \]
\[ - 4(x_{1,2,2}x_{5,2,2}) - 4(x_{3,2,2}x_{5,2,2}) \]
\[ + 4(x_{1,2,2}x_{3,2,2}x_{5,2,2}) - 6(x_{1,2,1}x_{3,2,4}) \]
\[ - 6(x_{1,2,1}x_{3,2,2}x_{5,2,2}) \]
\[ + 6(x_{1,2,1}x_{3,2,4}x_{5,2,2}) - (x_{1,2,1}x_{3,2,2}x_{5,2,2}) \]
\[ - (x_{1,2,1}x_{3,2,4}x_{5,2,2}) - 5(x_{4,2,1}x_{4,2,2}) \]
\[ - 4(x_{4,2,1}x_{5,2,1}) - 4(x_{4,2,1}x_{5,2,3}) \]
\[ - 4(x_{4,2,2}x_{5,2,1}) - (x_{4,2,2}x_{5,2,3}) \]
\[ - 5(x_{4,2,2}x_{5,2,3}) - 2(x_{4,2,2}x_{5,2,4}) \]
\[ - (x_{4,2,3}x_{5,2,2}) - 4(x_{5,2,1}x_{5,2,3}) \]
\[ - (x_{5,2,2}x_{5,2,3}) + (x_{1,2,2}x_{5,2,2}x_{5,2,3}) \]
\[ + 4(x_{4,2,1}x_{4,2,2}x_{5,2,1}) + 4(x_{4,2,1}x_{4,2,2}x_{5,2,3}) \]
\[ + 4(x_{4,2,1}x_{5,2,1}x_{5,2,3}) + (x_{4,2,2}x_{4,2,3}x_{5,2,2}) \]
\[ + 4(x_{4,2,2}x_{5,2,1}x_{5,2,3}) \]
\[ - (x_{4,2,1}x_{4,2,2}x_{5,2,1}x_{5,2,3}) \]
\[ \leq 28 \]

(20) \[ 10x_{1,3,2} + 3x_{3,3,1} + 10x_{4,3,1} + 14x_{4,3,2} + 8x_{5,3,3} + 2x_{5,3,4} \]
\[ - 5(x_{4,3,1}x_{4,3,2}) - (x_{1,3,2}x_{5,3,3}) \]
\[ - 3(x_{3,3,1}x_{4,3,2}) - (x_{5,3,1}x_{5,3,3}) \]
\[ - 4(x_{4,3,1}x_{5,3,3}) - 5(x_{4,3,2}x_{5,3,3}) \]
\[ - 2(x_{4,3,2}x_{5,3,4}) + (x_{3,3,1}x_{4,3,2}x_{5,3,3}) \]
\[ + 4(x_{4,3,1}x_{4,3,2}x_{5,3,3}) \]
\[ \leq 20 \]

(21) \[ 6x_{1,4,1} + 10x_{1,4,2} + 6x_{1,4,3} + 3x_{3,4,1} + 10x_{4,4,1} + 14x_{4,4,2} + 8x_{5,4,3} + 2x_{5,4,4} \]
\[ - 5(x_{4,4,1}x_{4,4,2}) - (x_{1,4,2}x_{5,4,3}) \]
\[ - 3(x_{3,4,1}x_{4,4,2}) - (x_{5,4,1}x_{5,4,3}) \]
\[ - 4(x_{4,4,1}x_{5,4,3}) - 5(x_{4,4,2}x_{5,4,3}) \]
\[ - 2(x_{4,4,2}x_{5,4,4}) + (x_{3,4,1}x_{4,4,2}x_{5,4,3}) \]
\[ + 4(x_{4,4,1}x_{4,4,2}x_{5,4,3}) \]
\[ \leq 35 \]

where, \( x_{i,j,k} = 0 \) or \( 1 \) for all \( i,j,k \).
Operation Allocation: Minimization of Transport Load

Objective function:

Minimize \( Z \)

\[
\begin{align*}
40(x_{1,1,1}) + 40(x_{1,1,2}) + 60(x_{1,2,1}) + 60(x_{1,2,2}) \\
+ 60(x_{1,2,3}) + 40(x_{1,3,2}) + 50(x_{1,4,1}) + 50(x_{1,4,2}) \\
+ 50(x_{1,4,3}) + 40(x_{3,1,3}) + 40(x_{3,1,4}) + 60(x_{3,2,2}) \\
+ 60(x_{3,2,3}) + 60(x_{3,2,4}) + 40(x_{3,3,1}) + 50(x_{3,4,1}) \\
+ 40(x_{4,1,1}) + 40(x_{4,1,2}) + 40(x_{4,1,3}) + 60(x_{4,2,1}) \\
+ 60(x_{4,2,2}) + 60(x_{4,2,3}) + 40(x_{4,3,1}) + 40(x_{4,3,2}) \\
+ 50(x_{4,4,1}) + 50(x_{4,4,2}) + 40(x_{5,1,1}) + 40(x_{5,1,2}) \\
+ 40(x_{5,1,3}) + 40(x_{5,1,4}) + 60(x_{5,2,1}) + 60(x_{5,2,2}) \\
+ 60(x_{5,2,3}) + 60(x_{5,2,4}) + 40(x_{5,3,1}) + 40(x_{5,3,4}) \\
+ 50(x_{5,4,1}) + 50(x_{5,4,2}) \\
- 60(x_{1,2,2} \cdot x_{1,2,3}) - 50(x_{1,4,2} \cdot x_{1,4,3}) \\
- 60(x_{3,2,2} \cdot x_{3,2,3}) - 60(x_{3,2,3} \cdot x_{3,2,4}) \\
- 40(x_{4,1,1} \cdot x_{4,1,2}) - 60(x_{4,2,1} \cdot x_{4,2,2}) \\
- 40(x_{5,1,1} \cdot x_{5,1,3}) - 60(x_{5,2,2} \cdot x_{5,2,3})
\end{align*}
\]

Subject to,

\begin{align*}
(1) & \quad x_{1,1,1} + x_{1,2,1} + x_{1,4,1} = 1 \\
(2) & \quad x_{1,1,2} + x_{1,2,2} + x_{1,3,2} + x_{1,4,2} = 1 \\
(3) & \quad x_{2,1,2} + x_{1,4,3} = 1 \\
(4) & \quad x_{2,3,1} + x_{2,4,1} = 1 \\
(5) & \quad x_{2,2,2} = 1 \\
(6) & \quad x_{2,1,3} + x_{2,2,3} = 1 \\
(7) & \quad x_{2,1,4} + x_{2,2,4} = 1 \\
(8) & \quad x_{3,1,1} + x_{3,2,1} + x_{3,3,1} + x_{5,4,1} = 1 \\
(9) & \quad x_{3,1,2} + x_{3,2,2} + x_{3,3,2} + x_{3,4,2} = 1 \\
(10) & \quad x_{3,1,3} + x_{3,2,3} = 1 \\
(11) & \quad x_{4,1,1} + x_{4,2,1} = 1 \\
(12) & \quad x_{4,1,2} + x_{4,2,2} = 1 \\
(13) & \quad x_{4,1,3} + x_{4,2,3} + x_{4,3,3} + x_{4,4,3} = 1 \\
(14) & \quad x_{4,1,4} + x_{4,2,4} + x_{4,3,4} + x_{4,4,4} = 1 \\
(15) & \quad (x_{3,1,4} \cdot x_{5,1,4}) + (x_{3,2,4} \cdot x_{5,2,4}) = 1 \\
(16) & \quad x_{1,1,2} + x_{1,2,2} + x_{1,3,2} + x_{1,4,2} + x_{3,1,3} + x_{3,2,2} + x_{3,2,3} - (x_{1,1,2} \cdot x_{3,1,3}) - (x_{1,2,2} \cdot x_{3,2,2}) \\
- (x_{1,2,2} \cdot x_{3,2,3}) - (x_{3,2,2} \cdot x_{3,2,3}) \\
+ (x_{1,2,2} \cdot x_{3,2,2} \cdot x_{3,2,3}) \leq 2 \\
(17) & \quad x_{1,1,1} + x_{1,2,1} + x_{1,4,1} + x_{3,1,4} + x_{3,2,4} + x_{5,1,2} + x_{5,2,2} - (x_{1,1,1} \cdot x_{3,1,4}) - (x_{1,1,1} \cdot x_{5,1,2}) \\
- (x_{1,1,1} \cdot x_{5,1,2}) + (x_{1,1,1} \cdot x_{3,1,4} \cdot x_{5,1,2}) \\
- (x_{1,2,2} \cdot x_{5,1,2}) \leq 2
\end{align*}
\[ 6x_{1,1,1} + 10x_{1,2,2} + 5x_{3,1,3} + 8x_{3,1,4} + 10x_{4,1,1} \\
+ 14x_{4,1,2} + 3x_{4,1,3} + 6x_{5,1,1} + 8x_{5,1,3} + 2x_{5,1,4} \\
+ 6x_{5,1,1} - (x_{4,1,2}x_{4,1,3} - 4(x_{1,2}x_{3,1,3}) \\
- 6(x_{1,1,1}x_{3,1,4}) - 6(x_{1,1,1}x_{5,1,2}) \\
- 6(x_{1,1,1}x_{5,1,2}) + 6x_{1,1,1}x_{3,1,4} + 4x_{5,1,2} \\
- (x_{1,1,2}x_{5,1,2}) - (x_{1,1,1}x_{5,1,3}) - 4(x_{4,1,2}-x_{5,1,3}) \\
- 4(x_{4,1,2}x_{5,1,1}) - 4x_{4,1,2}x_{5,1,1} \\
- 5(x_{4,1,2}x_{5,1,1}) - 2(x_{4,1,2}x_{5,1,1}) \\
- (x_{4,1,3}x_{5,1,2}) - 4x_{5,1,1}x_{5,1,3} \\
- (x_{5,1,2}x_{5,1,3}) + (x_{1,1,2}x_{5,1,1}) - x_{5,1,1} \\
+ 4(x_{4,1,1}x_{4,1,2}x_{5,1,1}) + 4(x_{4,1,1}x_{4,1,2}x_{5,1,1}) \\
+ 4(x_{4,1,1}x_{5,1,1}x_{5,1,1}) + (x_{4,1,2}x_{4,1,2}x_{5,1,2}) \\
+ 4x_{4,1,1}x_{4,1,2}x_{5,1,1} - 4x_{4,1,1}x_{4,1,2}x_{5,1,1} \leq 15 \]

\[ 6x_{1,2,1} + 10x_{1,2,2} + 6x_{1,2,3} + 9x_{3,2,2} + 5x_{3,2,3} \\
+ 8x_{3,2,4} + 10x_{4,2,1} + 14x_{4,2,2} + 3x_{4,2,3} + 6x_{5,2,1} \\
+ 8x_{5,2,2} + 8x_{5,2,3} + 2x_{5,2,4} \\
- (x_{4,2,2}x_{4,2,3}) - 4(x_{1,2,2}x_{5,2,2}) \\
- 4(x_{1,2,2}x_{3,2,3}) - 4(x_{3,2,2}x_{3,2,3}) \\
+ 4(x_{4,2,2}x_{3,2,2}x_{3,2,3}) - 6(x_{1,2,1}x_{3,2,4}) \\
- 6x_{1,2,1}x_{5,2,2} - 6x_{5,2,2}x_{5,2,2} \\
+ 6x_{1,2,1}x_{5,2,2} - 6x_{5,2,2}x_{5,2,2} - 6x_{1,2,1}x_{5,2,2} \\
- 5x_{1,2,1}x_{5,2,2} - 5x_{1,2,1}x_{5,2,2} \\
- 4(x_{4,2,1}x_{4,2,2}) - 4(x_{4,2,1}x_{5,2,2}) \\
- 4(x_{4,2,1}x_{5,2,2}) - 4(x_{4,2,2}x_{5,2,2}) \\
- (x_{4,2,2}x_{5,2,2}) - 4(x_{5,2,1}x_{5,2,2}) \\
- (x_{5,2,2}x_{5,2,2}) - 4(x_{5,2,1}x_{5,2,2}) \\
+ 4(x_{4,2,1}x_{5,2,2}x_{5,2,2}) + 4(x_{4,2,1}x_{5,2,2}x_{5,2,2}) \\
+ 4(x_{4,2,1}x_{5,2,2}x_{5,2,2}) + 4(x_{4,2,1}x_{5,2,2}x_{5,2,2}) \\
+ 4(x_{4,2,2}x_{5,2,2}x_{5,2,2}) \leq 28 \]

\[ 10x_{1,3,2} + 3x_{3,3,1} + 10x_{4,3,1} + 14x_{4,3,2} + 8x_{5,3,3} \\
+ 2x_{5,3,4} \\
- 5(x_{4,3,1}x_{4,3,2}) - (x_{4,3,1}x_{5,3,1}) \\
- 3(x_{5,3,1}x_{4,3,2}) - (x_{3,3,1}x_{5,3,1}) \\
- 4(x_{4,3,1}x_{5,3,3}) - 5(x_{3,3,2}x_{5,3,3}) \\
- 2(x_{4,3,2}x_{5,3,4}) + (x_{3,3,1}x_{4,3,2}) \\
+ 4(x_{4,3,1}x_{4,3,2}x_{5,3,3}) \leq 20 \]

\[ 6x_{1,4,1} + 10x_{1,4,2} + 6x_{1,4,3} + 3x_{3,4,1} + 10x_{4,4,1} \\
+ 14x_{4,4,2} + 8x_{5,4,3} + 2x_{5,4,4} \\
- 5(x_{4,4,1}x_{4,4,2}) - (x_{1,4,2}x_{5,4,3}) \\
- 3(x_{5,4,1}x_{4,4,2}) - (x_{3,4,1}x_{5,4,3}) \\
- 4(x_{4,4,1}x_{5,4,3}) - 5(x_{4,4,2}x_{5,4,3}) \\
- 2(x_{4,4,2}x_{5,4,3}) + (x_{3,4,1}x_{4,4,2}) \\
+ 4(x_{4,4,1}x_{4,4,2}x_{5,4,3}) \leq 35 \]

where \( x_{i,j,k} = 0 \) or 1 for all \( i,j,k \).
Part Mix Determination: Minimization of Workload Imbalances

Minimize \( Z = [2.5X_1 - 5.5X_2 - 3.75X_3 - 5.5X_4]^2 \)
\( + [-9.5X_1 + 20.5X_2 - 8.75X_3 + 28.5X_4]^2 \)
\( + [-9.5X_1 - 1.5X_2 + 21.25X_3 - 11.5X_4]^2 \)
\( + [16.5X_1 - 13.5X_2 - 8.75X_3 - 11.5X_4]^2 \)

Subject to,
\[
\begin{align*}
X_1 + X_2 + X_3 + X_4 & = 20 \\
X_1 + X_3 & \leq 10 \\
X_2 + X_4 & \leq 5 \\
2X_2 + X_3 + X_4 & \leq 15 \\
4 \leq X_1 & \leq 10 \\
2 \leq X_1 & \leq 5 \\
5 \leq X_1 & \leq 12 \\
1 \leq X_1 & \leq 4 \\
\end{align*}
\]

where \( X_i \) is integer for \( i = 1,2,3,4 \).

Part Mix Determination: Minimization of Transport Load

Minimize \( Z = 90X_1 + 140X_2 + 120X_3 + 160X_4 \)

Subject to,
\[
\begin{align*}
X_1 + X_2 + X_3 + X_4 & = 20 \\
X_1 + X_3 & \leq 10 \\
X_2 + X_4 & \leq 5 \\
2X_2 + X_3 + X_4 & \leq 15 \\
4 \leq X_1 & \leq 10 \\
2 \leq X_1 & \leq 5 \\
5 \leq X_1 & \leq 12 \\
1 \leq X_1 & \leq 4 \\
\end{align*}
\]

where \( X_i \) is integer for \( i = 1,2,3,4 \).

Part Mix Determination: Minimization of Refixturing Load

Minimize \( Z = 4X_1 + 13X_2 + 6X_3 + 15X_4 \)

Subject to,
\[
\begin{align*}
X_1 + X_2 + X_3 + X_4 & = 20 \\
X_1 + X_3 & \leq 10 \\
X_2 + X_4 & \leq 5 \\
2X_2 + X_3 + X_4 & \leq 15 \\
4 \leq X_1 & \leq 10 \\
2 \leq X_1 & \leq 5 \\
5 \leq X_1 & \leq 12 \\
1 \leq X_1 & \leq 4 \\
\end{align*}
\]

where \( X_i \) is integer for \( i = 1,2,3,4 \).
APPENDIX - D

COMPUTER PROGRAM LISTING
MATHEMATICAL MODELLING OF FLEXIBLE MANUFACTURING SYSTEMS

AUTHOR - ARVIND MADHUKAR PADHYE

GRADUATE STUDENT, DEPT. OF INDUSTRY ENGINEERING
UNIVERSITY OF WINDSOR, WINDSOR, ONTARIO N9B 3P4
APRIL 1986.

THIS PROGRAM IS WRITTEN TO PREPARE A COMPUTER FILE OF THE
FORMULATIONS OF THE OPERATION ALLOCATION PROBLEM IN FMS.

THIS FILE CAN BE USED AS AN INPUT FILE FOR THE LP ROUTINE OF THE
SAS/OR (VERSION 5) PACKAGE USING THE 'INFILE' STATEMENT.

INTEGER TYP, TOTIV, PTYPE, H,
* DPNI(100), NP(500), NJ(500), NK(500), NY(500), POS(10, 10, 10),
* MC(100), EMA, PB(25), PT(500, 50), RDW, ROWSUM, COLUMN,
* BF(25, 2), ROWNUM, TDB(10, 10, 100), T, P, TB(25, 2), RARETL,
* S1, S2, S3, S4, S5, S6, S7, S8, S9, S10, E1, E2, E3, E4, E5, E6,
* E7, E8, E9, E10, R, P, R(25), C1, C2, C3, TJ(10), SUM1, TS, PROD,
* E11, E12, E13, E14, E15, E16, E17, E18, E19, E20, S11, S12, S13, S14,
* S15, S16, S17, S18, S19, S20, ACTIV, PFLS, CACTS(10), SUM2, SUM3,
* F(10, 10, 10), FIXFLS, OBJL, PACT(10), UL, OBJECT,
* DLD(10), DUL(10), LSTOR(25), GSTOR(10, 25, 2), GAMMAS, ROWBUF,
* SUNS, DPNLIM, STRTGD, CSKIP, PCT, VERTAB, COLBFG, TAB(1500, 50, 2),
* ITAB(200), AMRES, FSEM, FCRSEI
REAL Y(200), PV8(500), YOUT(200), WET(10)
CHARACTER DESC120, DESC240, M11, CTYPE10

CALL TINIT
PRINT, TIME USED BY DIFFERENT SECTIONS IN MILISECONDS:

ARRAY INITIALIZE

DD : 100 I=1, 200
NY(I)=0
Y(I)=0.0
YOUT(I)=0.0
XTAB(I)=0
100 CONTINUE
DO 102 I=1, 10
MC(I)=0
DPNI(I)=0
TJ(I)=0

NCST(I)=0
FACT(I)=0
LDL(I)=0
ULL(I)=0
WET(I)=0.0

DO 111 J=1,25
SGSTOR(I, J, 1) = 0
SGSTOR(I, J, 2) = 0
111 CONTINUE

DO 104 J=1,10
DO 106 K=1,10
POS(I, J, K) = 0
DO 107 L=1,10
F(I, J, K, L) = 99
107 CONTINUE

106 CONTINUE

DO 112 K=1,100
TDB(I, J, K) = 0
112 CONTINUE

104 CONTINUE

DO 109 I=1,500
PB(I) = 0
NR(I) = 0
MR(I) = 0
DO 109 J=1,10
PT(I, J) = 0
TAB(I, J, 1) = 0
TAB(I, J, 2) = 0
109 CONTINUE

108 CONTINUE

DO 110 I=1,25
PB(I) = 0
PR(I) = 0
TB(I, 1) = 0
TB(I, 2) = 0
BF(I, 1) = 0
BF(I, 2) = 0
TLSTOR(I) = 0
110 CONTINUE

ROWNUM=0
ROWSUM=50
COEFF=1.0
NVAR1=0
NVAR2=0
NVAR3=0
OBJLB=0
LPB=0
WCMR=0.0
SUMS=0
C
C INITIALIZE THE FOLLOWING VARIABLES FOR EACH RUN OF
C THIS PROGRAM. THE LIMITS ON VALUES ARE,
C MACH : NUMBER OF MACHINES (1-9)
C MAIV : MAXIMUM NUMBER OF INTEGER VARIABLES ALLOWED.
        (1-100).
C MAXFV : MAXIMUM NUMBER OF FREE VARIABLES ALLOWED.
        (1-500).
C MAIOPN : MAXIMUM NUMBER OF OPERATIONS ALLOWED FOR
          ALLOCATION. (ALL PARTS? (1-25).
C MAXDB : MAXIMUM NUMBER OF TOOLS ALLOWED IN THE
          IN THE SYSTEM.
C MAXTL : MAXIMUM NUMBER OF TOOLS ALLOWED PER PART TYPE.
          (100).
C

SET OPT=9 IF NO OUTPUT FILE IS DESIRED.
OPT=0
C
C SET OBJECT=1 FOR MINIMIZATION OF REFITURING LOAD.
OBJECT=1
C SET OBJECT=2 FOR MINIMIZATION OF TRANSPORT LOAD.
OBJECT=1
C
C SET RMSR=1 IF DEGREE OF REFITURING COMPLEXITY IS TO BE USED.
RMESR=1
C SET MAIFV=NUMBER OF FREE LINEARIZATION VARIABLES ALLOWED.
MAXFV=100
C
C SET STRTG=1, 2, 3, 4, 5 TO CHOOSE A LINEARIZATION STRATEGY
STRTG=2
C
C SET MACH = NUMBER OF MACHINES
MACH=4
C
C SET CSKIP=0 IF TOOL MAGAZINE CONSTRAINTS ARE NOT NEEDED
CSKIP=1
C

IF(STRTG.LT.1.OR.STRTG.GT.5) THEN
  ERR=20
  EXECUTE ERPARA
ENDIF
IF(CSKIP.NE.0) CSKIP=1
IF(RMESR.NE.1) RMESR=0
MAXV=200
MAIOPN=20
MAXDB=55
MAXTL=20
ROWBUF=ROWSUM-1
COLBUF=MAXOPL
IM=1
EXECUTE MI1
READ AGGREGATE PART INFORMATION FROM THE CARD DECK

C

PTYPE=0
N1=1

WHILE(N1.NE.0)
READ 800; DESC,N1,N2,W
800
FORMAT(A20,1X,13,1X,13,1X,F4.2)
IF(N1.EQ.0) GOTO 510
PTYPE=PTYPE+1
PACT(PTYPE)=N1
QPM(PTYPE)=N2
SUMS=SUMS+QPM(PTYPE)
WCHK=WCHK+W
IF(W.LE.0.000001) W=1.00
WET(PTYPE)=W
ENDWHILE
510
CONTINUE
IF(WCHK.NE.0.0.AND.WCHK.LT.0.99) THEN
ERR=10
EXECUTE ERPARA
ENDIF
IF(SUMS.GT.QPM(LIM)) THEN
ERR=14
EXECUTE ERPARA
ENDIF

SET UP INTEGER VARIABLES BUFFER

L=0
DO 550 I=1,PTYPE
L:I=QPM(I)
DO 555 J=1,MACH
DO 560 K=1,LIM
L=L+1
N1(L)=FACT(I)
NK(L)=K
NJ(L)=J
560 CONTINUE
555 CONTINUE
550 CONTINUE
TOTIV=L
IF(TOTIV.GT.MAXIV) THEN
ERR=10
EXECUTE ERPARA
ENDIF
READ TOOL DATABASE

I=1
READ 570, DESC
FORMAT(A20)
WHILE (I.NE.0)
READ 572, I,K,(NY(T),T=1,MAXT)
FORMAT(12,2X,12,4X,20(I2,1X))
IF(I.EQ.0) GOTO 576
DO 574 T=1,MAXT
IF(NY(T).NE.0) THEN
TDB(I,K,MY(T))=1
XTAB(NY(T))=XTAB(NY(T))+1
ENDIF
574 CONTINUE
576 CONTINUE
ENDWHILE
READ, (NY(J),J=1,100)

READ TOOL SLOTS AVAILABLE ON EACH MACHINE

READ 727, DESC
READ 727, DESC
READ 727, DESC
J=0
ITL=1
WHILE (ITL.NE.0)
READ 1226, ITL,ICAP,UL,LD
FORMAT(12,2X,13,1X,13,1X,13)
IF(ITL.EQ.0) GOTO 1228
J=J+1
I2(J)=ICAP
DLD(J)=LD
DUL(J)=UL
1228 CONTINUE
ENDWHILE
IF(J.LT.MACH) THEN
ERR=16
EXECUTE ERERA
ENDIF

READ POSSIBILITIES MATRIX

I=1
READ 602, DESC
READ 602, DESC
FORMAT(A20)
WHILE (I.LE.0)
READ 605, I,K,1,(MC(J),J=1,MACH)
FORMAT(12,2X,12,2X,12,2X,12,2X,12,2X,12,2X)
IF(I.LE.0) GOTO 615
DO 610 J=1,MACH
POS(I,J,K)=MC(J)
610 CONTINUE
615 CONTINUE
ENDWHILE

C----------------------------------------
C READ FIXTURING INFORMATION
C----------------------------------------
READ 727,DESC
READ 727,DESC
READ 727,DESC
II=1
WHILE(II.NE.0)
READ 1290,II,II,II,II,II,IVAL
1290 FORMAT(II,II,II,II,II,II)
IF(II.EQ.0) GOTO 1292
IF(RMRES.NE.1) THEN
IF(IVAL.NE.0) IVAL=1
ENDIF
F(II,II,II,II)=IVAL
1292 CONTINUE
ENDWHILE
IH=2
EXECUTE MILI

C----------------------------------------
C OBJECTIVE FUNCTION FORMULATION
C----------------------------------------
IF(OBJECT.EQ.1) EXECUTE REFIX
IF(OBJECT.EQ.2) EXECUTE TRAVEL
IF(OBJECT.NE.1.AND.OBJECT.NE.2) THEN
ERR=18
EXECUTE_ERPARA
ENDIF
IH=3
EXECUTE MILI

C----------------------------------------
C UNIQUE ALLOCATION CONSTRAINTS GENERATION
C----------------------------------------
DO 700 I=1,PTYPE
LIM=OPN(I)
DO 705 K=1,LIM
C
DO 710 J=1,MACH
C1=FACT(I)
C2=J
C3=K
EXECUTE LOCY
IF(POS(FACT(I),J,K),NE.0) THEN
Y(LOCY=*LOCY+CCEFF*)
ELSE


Y(LGC) = Y(LGC) + 0.0
ENDIF

710 CONTINUE
CYTPE='EB'
RHS=1.0
EXECUTE PURE
NCST(1) = NCST(1) + 1

705 CONTINUE
700 CONTINUE
IH=4
EXECUTE MILI

C-----------------------------------------------------------------------------
C FORCED ASSIGNMENTS FOR GROUPS OF OPERATIONS
C-----------------------------------------------------------------------------

IG=0
READ 727, DESC
READ 727, DESC

727 FORMAT(A20)
735 IK=0
LPB=0

740 READ 725, DESC2, K1, K2
725 FORMAT(A6,2X, I2, 2X, I2)
IF(IDESC .EQ. 'ENDATA') GOTO 110
IF(IDESC .EQ. 'OPMSET') GOTO 737
IF(IDESC .EQ. 'ENDSET') GOTO 738
IF(IDESC .NE. '') THEN

ERA=12
EXECUTE ERFA
ENDIF

730 IK=IK+1
BF(IK, 1) = K1
BF(IK, 2) = K2
6RSTOR(16, IK, 1) = K1
6RSTOR(15, IK, 2) = K2
GOTO 740

C-----------------------------------------------------------------------------
C RESTART CYCLE
C-----------------------------------------------------------------------------

737 DO 736 IK=1, MATOPN
BF(IK, 1) = 0
BF(IK, 2) = 0
736 CONTINUE
IG=IG+1
GOTO 725

C-----------------------------------------------------------------------------
C PROCESSING THE SET OF OPERATIONS TO BE GROUPED
C-----------------------------------------------------------------------------

728 DO 742 J=1, NACH
DO 745 IH=1, IK
IF(JPST(BF(JH, J), BF(IH, J+1), ED)) GOTO 742
745 CONTINUE

C-----------------------------------------------------------------------------
DD 750 IH=1, IK
C1=BF(IH,1)
C2=1
C3=BF(IH,2)
EXECUTE LOADPB
750 CONTINUE
EXECUTE LOCPV
742 CONTINUE
CTYPE="EQ"
RHS=1.0
EXECUTE PURGE
NCST(2)=NCST(2)+1
DO 739 IH=1, IK
BF(IH,1)=0
BF(IH,2)=0
739 CONTINUE
GOTO 735
1110 IH=5
EXECUTE MILI
C QUITTING FORCED ASSIGNMENT SECTION
C-------------------------------------------------------------
C SCRAE TOOLS CONSTRAINTS GENERATION
C-------------------------------------------------------------
READ 570, DESC
READ 570, DESC
PROC=1
RARETL=1
ITL=0
WHILE(RARETL.NE.0)
925 READ 920, RARETL,P
920 FORMAT(12,2I1,2I2)
IF(RARETL.EQ.0) GOTO 922
IF(P.GE.MACH) GOTO 925
C EXECUTE LOCP
IF(INTB.LE.P) GOTO 925
ITL=ITL+1
ITLSTOR(ITL)=RARETL
DO 927 J=1, MACH
DO 930 R=1, NTB
COEFF=(-1.0)**(R+1)
EXECUTE MULTI
930 CONTINUE
927 CONTINUE
CTYPE="LE"
RHS=FLOAT(P)
EXECUTE PURGE
NCST(C)=NCST(C)+1
932 CONTINUE
ENDWHILE
IH=6
EXECUTE MILI

C-----------------------------------------------

C TOOL MAGAZINES CAPACITY CONSTRAINTS

C-----------------------------------------------

IF(CSCHIP.EQ.0) GOTO 1209
PRG=2
MOPN=0
NTB=0
DO 1200 I=1,PTYPE
   LIM=OPN(I)
   DO 1201 K=1,LIM
      NTB=NTB+1
      T(KNTB,1)=PACT(I)
      T(KNTB,2)=K
   1201 CONTINUE
1200 CONTINUE
EXECUTE CLIMPB
FFLAG=1
DO 1204 R=1,NTB
   IF(FFLAG.EQ.0) GOTO 1725
   FFLAG=0
   IF(R.EQ.2) ITAB=1
   EXECUTE MULTI
1204 CONTINUE
1725 IF( (R-1),GT,OPNLIM) THEN
      ERR=21
      EXECUTE ERPARA
      ENDFI
      LENTAB=ITAB
      DO 1650 J=1,MACH
      DO 1700 JTAB=1,NTB
         C1=TAB(1,JTAB,J)
         C2=J
         C3=TAB(1,JTAB,J)
         IF( POS(C1,C2,C3).NE.3) THEN
            EXECUTE LOCY
            EXECUTE SLOTAQ
            COEFF=TSUM
            Y(LOCY)=Y(LOCY)+COEFF
            COEFF=1.0
      1650 CONTINUE
      C
      DO 1705 JTAB=2,LENTAB
         VERTAB=TAB(1,JTAB,J)
      DO 1710 JTAB=1,VERTAB
         C1=TAB(1,JTAB,J)
         C2=J
         C3=TAB(1,JTAB,J)
      1700 CONTINUE
   )
IF( POS(C1,C2,C3).EQ.0) THEN
  EXECUTE CLINPB
  GOTO 1705
ENDIF.
EXECUTE LOADPB

1710 CONTINUE
  COEFF=FLOAT(TAB(ITAB,COLBUF,2) #
  ( (-1)**(TAB(ITAB,COLBUF,1)+1) ) )
  EXECUTE LOCPR
1705 CONTINUE
  CTYP="LE"
  RHS=FLOAT(IJ(J))
  EXECUTE PURGE
  NEST(4)=NEST(4)+1

1690 CONTINUE
1289 IH=7
  EXECUTE MILI-

C---------------------------------------------------------------
C  LINEARIZATION OF PRODUCT TERMS.
C---------------------------------------------------------------

IF(ROWNUM.EQ.0) GOTO 1251
  EXECUTE CLEAN
  EXECUTE CLINPB
  IF(STRAT.GE.EQ.1) THEN
    EXECUTE LIN11
    EXECUTE LIN21
  ENDIF
  IF(STRAT.GE.EQ.2) THEN
    EXECUTE LIN12
    EXECUTE LIN21
  ENDIF
  IF(STRAT.GE.EQ.3) THEN
    EXECUTE LIN11
    EXECUTE LINC2
  ENDIF
  IF(STRAT.GE.EQ.4) THEN
    EXECUTE LIN12
    EXECUTE LIN22
  ENDIF

C---------------------------------------------------------------
C  INTEGRALITY CONSTRAINTS
C---------------------------------------------------------------

1251 MVAR1=0
  MVAR2=0
  MVAR3=0
  DO 1250 I=1,NTYPE
  LIN=DPN(I)
  DO 1255 J=1,12
  SUM1=0
  DO 1255 J=1,MACH
  IF(POS(FACT(1),1,1,J,1,1,EQ,12)) SUM2=SUM2+1
1254 CONTINUE
IF (SUM2.LE.0 .OR. SUM2.GT.MACH) THEN
ERR=17
EXECUTE ERPARA
ENDIF
IF (SUM2.EQ.0) GOTO 1257
SUM3=0
DO 1256 J=1,MACH
IF (POS (FACT(I),J,K).NE.0 .AND. SUM3 .LT. (SUM2-1)) THEN
   CI=FACT(I)
   C2=J
   CJ=K
   EXECUTE LOCY
   Y (LOCY)=1.0
   SUM3=SUM3+1
ENDIF
1256 CONTINUE
NVAR1=NVAR1+SUM3
1257 CONTINUE
NVAR2=NVAR2+SUM2
1250 CONTINUE
CTYPE='INTEGER'
RHS=0.0
EXECUTE PURGE
NCST(B)=NCST(B)+1
IH=10
EXECUTE MILI
C-----------------------------------------------
C UPPERBOUNDS ON THE FREE AND INTEGER VARIABLES
C-----------------------------------------------
DO 1356 L=1,TOTIV
Y(L)=1.0
1356 CONTINUE
IF (MAXIV.EQ.0) GOTO 1261
DO 1260 L=1,MAXIV
PVY(L)=1.0
1260 CONTINUE
1261 CONTINUE
CTYPE='UPPERBO'
RHS=0.0
EXECUTE PURGE
NCST(?)=NCST(?)+1
IH=11
EXECUTE MILI
GOTO 9999
C-----------------------------------------------
REMOTE BLOCK LOCOP
T=ARETL
NTE=0
DG (*4)=1,FYPE
LIM=GMX(1)
DO 936 K=1,LIM
   IF(TDB(PACT(I),K,T).EQ.0) GOTO 936
   NTB=NTB+1
   TB(NTB,1)=PACT(I)
   TB(NTB,2)=K
936 CONTINUE
934 CONTINUE
ENDBLOCK

REMOTE BLOCK PRLOC
EXECUTE CLINPB
PR(1)=1
PR(2)=12
PR(3)=13
PR(4)=14
PR(5)=15
PR(6)=16
PR(7)=17
PR(8)=18
PR(9)=19
PR(10)=110
PR(11)=111
PR(12)=112
PR(13)=113
PR(14)=114
PR(15)=115
PR(16)=116
PR(17)=117
PR(18)=118
PR(19)=119
PR(20)=120

C
DO 940 IPR=1,MAXPN
   IF(PR(IPR).EQ.0) GOTO 943
   CI=TB(PR(IPR),1)
   C2=J
   CC=TB(PR(IPR),2)
   IF(pos(C1,C2,CC).EQ.0) GOTO 942
   EXECUTE LOADPB
940 CONTINUE
942 CONTINUE
EXECUTE WRITPB
EXECUTE LDCPV
942 CONTINUE
ENDBLOCK

REMOTE BLOCK LINLOC
PFLAG=1
   IF(PROC.EQ.2) THEN
      PCT=PCT+1
      TAB(:,PCT,1)=TB(:,PR,1)
TAB(1,FCT,2)=TB(IPR,2)
GOTO 1720
ENDIF
C1=TE(IPR,1)
C2=1
C3=TB(IPR,2)
EXECUTE LOCY
IF (POS(C1,C2,C3).NE.0) THEN
  Y(LOC)=Y(LOC)+COEFF*1.0
ENDIF
1720  COEFF=1.0
ENDBLOCK

REMOTE BLOCK MULTI
IF (NTB.LE.0.OR.NTB.GT.20.OR.R.LT.0.OR.R.GT.NTB) THEN
  ERR=14
  EXECUTE ERPARA
ENDIF

I1=0
I2=0
I3=0
I4=0
I5=0
I6=0
I7=0
I8=0
I9=0
I10=0
I11=0
I12=0
I13=0
I14=0
I15=0
I16=0
I17=0
I18=0
I19=0
I20=0
E1=NTB-R+1
E2=NTB-R+2
E3=NTB-R+3
E4=NTB-R+4
E5=NTB-R+5
E6=NTB-R+6
E7=NTB-R+7
E8=NTB-R+8
E9=NTB-R+9
E10=NTB-R+10
E11=NTB-R+11
E12=NTB-R+12
E13=MTB-R+13
E14=MTB-R+14
E15=MTB-R+15
E16=MTB-R+16
E17=MTB-R+17
E18=MTB-R+18
E19=MTB-R+19
E20=MTB-R+20

DO 1010 I1=1,E1
  IF (E1.EQ.NTB) THEN
    IRC=I1
  ENDIF
  EXECUTE LIMLOC
  GOTO 1010
ENDIF
S2=I1+1
DO 1020 I2=S2,E2
  IF (E2.EQ.NTB) THEN
    EXECUTE GRLOC
    GOTO 1020
  ENDIF
S3=I2+1
DO 1030 I3=S3,E3
  IF (E3.EQ.NTB) THEN
    EXECUTE GRLOC
    GOTO 1030
  ENDIF
S4=I3+1
DO 1040 I4=S4,E4
  IF (E4.EQ.NTB) THEN
    EXECUTE GRLOC
    GOTO 1040
  ENDIF
S5=I4+1
DO 1050 I5=S5,E5
  IF (E5.EQ.NTB) THEN
    EXECUTE GRLOC
    GOTO 1050
  ENDIF
S6=I5+1
DO 1060 I6=S6,E6
  IF (E6.EQ.NTB) THEN
    EXECUTE GRLOC
    GOTO 1060
  ENDIF
S7=I6+1
DO 1070 I7=S7,E7
  IF (I7.EQ.NTB) THEN
    EXECUTE GRLOC
    GOTO 1070
  ENDIF
S8=I7+1
DO 1080 I9=S8,E9
IF(E8.EQ.NTB) THEN
EXECUTE SRLGC
GOTO 1080
ENDIF
S9=I9+1
DO 1090 I9=S9,E9
IF(E9.EQ.NTB) THEN
EXECUTE SRLGC
GOTO 1090
ENDIF
S10=I9+1
DO 1100 I10=S10,E10
IF(E10.EQ.NTB) THEN
EXECUTE SRLGC
GOTO 1100
ENDIF
S11=I10+1
DO 1102 I11=S11,E11
IF(E11.EQ.NTB) THEN
EXECUTE SRLGC
GOTO 1102
ENDIF
S12=I11+1
DO 1104 I12=S12,E12
IF(E12.EQ.NTB) THEN
EXECUTE SRLGC
GOTO 1104
ENDIF
S13=I12+1
DO 1106 I13=S13,E13
IF(E13.EQ.NTB) THEN
EXECUTE SRLGC
GOTO 1106
ENDIF
S14=I13+1
DO 1108 I14=S14,E14
IF(E14.EQ.NTB) THEN
EXECUTE SRLGC
GOTO 1108
ENDIF
S15=I14+1
DO 1109 I15=S15,E15
IF(E15.EQ.NTB) THEN
EXECUTE SRLGC
GOTO 1109
ENDIF
S16=I15+1
DO 1110 I16=S16,E16
IF(E16.EQ.NTB) THEN
EXECUTE GRC
GOTO 1112
ENDIF
$17=116+1
DO 1116 117=$17,E17
IF(E17.EQ.NTB) THEN
EXECUTE GRC
GOTO 1116
ENDIF
$18=117+1
DO 1118 118=$18,E18
IF(E18.EQ.NTB) THEN
EXECUTE GRC
GOTO 1118
ENDIF
$19=118+1
DO 1120 119=$19,E19
IF(E19.EQ.NTB) THEN
EXECUTE GRC
GOTO 1120
ENDIF
$20=119+1
DO 1122 120=$20,E20
IF(E20.EQ.NTB) THEN
EXECUTE GRC
GOTO 1122
ENDIF
1122 CONTINUE
1120 CONTINUE
1118 CONTINUE
1116 CONTINUE
1112 CONTINUE
1109 CONTINUE
1108 CONTINUE
1106 CONTINUE
1104 CONTINUE
1102 CONTINUE
1100 CONTINUE
1099 CONTINUE
1090 CONTINUE
1080 CONTINUE
1070 CONTINUE
1060 CONTINUE
1050 CONTINUE
1040 CONTINUE
1030 CONTINUE
1020 CONTINUE
1010 CONTINUE
END BLOCK
REMOTE BLOCK CLINPB
DO 816 LB=1,MAXPN
PB(LB)=0
816 CONTINUE
LPB=0
ENDBLOCK

REMOTE BLOCK CONTL
TS=0
PR(1)=11
PR(2)=12
PR(3)=13
PR(4)=14
PR(5)=15
PR(6)=16
PR(7)=17
PR(8)=18
PR(9)=19
PR(10)=110
PR(11)=111
PR(12)=112
PR(13)=113
PR(14)=114
PR(15)=115
PR(16)=116
PR(17)=117
PR(18)=118
PR(19)=119
PR(20)=120
DO 1206 T=1,MAXTDB
IF(XTAB(T).LT.R) GOTO 1206
SUM1=0
DO 1208 ICT=1,R
IF(TAB(T,PR(ICT),1,TR(ICT),2).LT.R) SUM1=SUM1+1
1208 CONTINUE
IF(SUM1.EQ.R) TS=TS+4Y(T)
1206 CONTINUE
IF(TS.EQ.0) GOTO 1216
ITAB=ITAB+1
DO 1715 JTAB=1,R
TAB(ITAB,JTAB,1)=TB(PR(JTAB),1)
TAB(ITAB,JTAB,2)=TB(PR(JTAB),2)
1715 CONTINUE
TAB(ITAB,COLSUF,1)=R
TAB(ITAB,COLSUF,2)=TS
PFLAG=1
1216 CONTINUE
ENDBLOCK
REMOTE BLOCK FIXCHK
FCRAI=0
FSIM=0
DO 1405 IFP=1,MACH
DO 1407 IFG=1,MACH
IF(F(PACT(I1),1K,IFP,IFQ).EQ.0) THEN
FIXFLG=1
GOTO 1409
ENDIF
IF(F(SIM.EQ.0.AND.F(PACT(I1),1K,IFP,IFQ).NE.99)
FSIM=F(PACT(I1),1K,IFP,IFQ)
IF(SIM.NE.F(PACT(I1),1K,IFP,IFQ).AND.
F(PACT(I1),1K,IFP,IFQ).NE.99) FCRAI=1
1407 CONTINUE
1405 CONTINUE
IF(FCRAI.EQ.1) FIXFLG=1
1409 CONTINUE
ENDBLOCK
C~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
REMOTE BLOCK REFIX
C~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
FIXFLG=0
DO 1400 II=1,PRTYPE
LIM=OPN(III)
IF(LIM.LT.2) GOTO 1400
DO 1402 IK=2,LIM
EXECUTE FIXCHK
IF(FIXFLG.EQ.0) THEN
OBULE=OB3LB+FSIM
GOTO 1402
ELSE
FIXFLG=0
ENDIF
DO 1404 IP=1,MACH
EXECUTE CLINPB
C1=PACT(I1)
C2=IP
C3=IK+1
IF(POS(C1,C2,C3).EQ.0) THEN
GOTO 1404
ELSE
EXECUTE LOADPB
ENDIF
DO 1406 IQ=1,MACH
IF(F(PACT(I1),1K,IP,IQ).EQ.0.OR.F(PACT(I1),1K,IP,IQ).EQ.99)
GOTO 1406
C1=PACT(I1)
C2=IQ
C3=IK
IF(POS(C1,C2,C3).EQ.0) THEN
GOTO 1406
ELSE
  COEFF=F(PACT(II),IK,IP,IO)
  EXECUTE LOADP8
  EXECUTE LOCPU
  ENDF
  C1=PACT(II)
  C2=IP
  C3=IK-1
  EXECUTE LOADP8
  1406 CONTINUE
  1404 CONTINUE
  1402 CONTINUE
  1400 CONTINUE
  CTYPE='MIN'
  RHS=0.0
  EXECUTE PURGE
  NCST(7)=NCST(7)+1
ENDBLOCK

REMOTE BLOCK TRAVEL
DO 1500 IRF=1,PTYPE
  LIM=OPM(IRF)-1
  DO 1502 KRF=1,LIM
    DO 1504 JRF=1,MACH
      C1=PACT(IRF)
      C2=JRF
      C3=KRF
      EXECUTE LOCY
      IF(P(S(C1,C2,C3).NE.0) THEN
        COEFF=DUL(JRF)*MET(IRF)
        Y(LOC)=Y(LOC)+COEFF
      ELSE
        Y(LOC)=Y(LOC)+0.0
      ENDF
      C3=KRF+1
      EXECUTE LOCY
      IF(P(S(C1,C2,C3).NE.0) THEN
        COEFF=DUL(JRF)*MET(IRF)
        Y(LOC)=Y(LOC)+COEFF
      ELSE
        Y(LOC)=Y(LOC)+0.0
      ENDF
      COEFF=1.0
      C3=KRF+1
DC1=KRF
DC2=KRF+1
IF(P(S(C1,C2,DC1).NE.0.AND.P(S(C1,C2,DC2).NE.0) THEN
  C3=KRF
  EXECUTE LOADP8
  C3=KRF+1
EXECUTE LOADPB
IFIX=(FACT(IRF),CS,IRF,IRF)
IF(IFI.EQ.0.OR.IFII.EQ.99) THEN.
IFI=0
ELSE
IFI=1
ENDIF
COEFF=(ULJRF+OLDJRF)!!(IFI-1)!!WET(IRF)
IF(COEFF.ME.0) THEN
EXECUTE LDCPV
ELSE
EXECUTE CLIMB
ENDIF
COEFF=1.0
ENDIF

C
1504 CONTINUE
1502 CONTINUE
C
DO 1506 JRF=1,MACH
C1=FACT(IRF)
C2=IRF
C3=1
EXECUTE LOCY
IF(POS(C1,C2,C3).NE.0) THEN
COEFF=OLDJRF!!WET(IRF)
Y(LOC)=Y(LOC)+COEFF
ELSE
Y(LOC)=Y(LOC)+0.0
ENDIF
C
C
C3=OPM(IRF)
EXECUTE LOCY
IF(POS(C1,C2,C3).NE.0) THEN
COEFF=ULJRF!!WET(IRF)
Y(LOC)=Y(LOC)+COEFF
ELSE
Y(LOC)=Y(LOC)+0.0
ENDIF
COEFF=1.0
1506 CONTINUE
1500 CONTINUE
CTYPE='MIN'
RMS=0.0
EXECUTE PURGE
MCST(7)=MCST(7)+1
END9LOCK

C-------------------
REMOTE BLOCK M11
CALL TUSED(H)
IF(H.EQ.1) PRINT,'INITIALIATION TIME *=H
IF(H.EQ.2) PRINT,'DATA READING TIME *=H
IF(H.EQ.3) PRINT,'OBJECTIVE FUNCTION TIME *=H
IF(H.EQ.4) PRINT,'TYPE I CONSTRAINTS TIME *=H
IF(H.EQ.5) PRINT,'FORCED ASSIGNMENT TIME *=H
IF(H.EQ.6) PRINT,'SCARCE TOOLS TIME *=H
IF(H.EQ.7) PRINT,'TOOL MAGAZINES TIME *=H
IF(H.EQ.8) PRINT,'LINEARIZATION-I TIME *=H
IF(H.EQ.9) PRINT,'LINEARIZATION-II TIME *=H
IF(H.EQ.10) PRINT,'INTEGRALITY TIME *=H
IF(H.EQ.11) PRINT,'UPPER BOUND ROW TIME *=H
CALL TINIT
ENDBLOCK

REMOTE BLOCK PMAT
IF(ROWNUM.EQ.0) GOTO 907
PRINT,'TABLE OF PRODUCT TERMS :
DO 904 I=1,ROWNUM
PRINT 908,I,(PT(I,J),J=1,MAXPN),PT(I,ROWBUF),PT(I,ROWSUM)
908 FORMAT(21,50I3)
904 CONTINUE
907 CONTINUE
ENDBLOCK

REMOTE BLOCK PTAB
PRINT,'#### INTEGER VARIABLE TABLE ####
DO 910 L=1,TOTIV
PRINT,L,NI(L),NJ(L),NK(L)
910 CONTINUE
ENDBLOCK

REMOTE BLOCK LCDV
LOC=0
DO 1000 ILDC=1,TOTIV
  THEN
    LOC=ILDC
  ENDIF
1000 CONTINUE
IF(LOC.EQ.0) THEN
  ERR=11
  EXECUTE EPARA
ENDIF
ENDBLOCK

REMOTE BLOCK PURGE
IF(OPT.EQ.9) GOTO 5003
ACTIV=0
DO 1500 L=1,TOTIV
IF(PBNI(L),NI(L),NK(L)).EQ.0) GOTO 1300
ACTIV=ACTIV+1
YOUT(ACTIV)=Y(L)
1300 CONTINUE
IF(MAXVF.EQ.0) THEN
WRITE(8,5000) (YOUT(L),L=1,ACTIV)
ELSE
WRITE(8,5000) (YOUT(L),L=1,ACTIV),(PVB(L),L=1,MAIFV)
ENDIF
5000 FORMAT(16,X,B(6,1,2))
WRITE(8,5002) CTYP,RHS
5002 FORMAT(2x,A8,2x,F7.2)
EXECUTE CLEAN
5003 CONTINUE
ENDBLOCK

C-------------------------------------------------------------

REMOTE BLOCK CLEAN
DO 5010 L=1,TOTIV
Y(L)=0.0
YOUT(L)=0.0
5010 CONTINUE
IF(MAIFV.NE.0) THEN
DO 5012 L=1,MAIFV
PVB(L)=0.0
5012 CONTINUE
ENDIF
CTYP=''
RHS=0.0
COEFF=1.0
ENDBLOCK

C-------------------------------------------------------------

REMOTE BLOCK EPRARA
IF(ERR.EQ.10) THEN
PRINT,'MAXIMUM NUMBER OF INTEGER VARIABLES ALLOWED = ',MAXIV
PRINT,'THE PROBLEM HAS TOO MANY INTEGER VARIABLES = ',TOTIV
ENDIF
IF(ERR.EQ.11) THEN
PRINT,'UNABLE TO LOCATE VARIABLE IN RB LOCY.'
PRINT,'PART TYPE = ',C1,'OPN = ',C2,'MACHINE = ',C3
ENDIF
IF(ERR.EQ.12) THEN
PRINT,'***** E-R-R-R-R: GROUPED ASSIGNMENT DATA INCORRECT *****
PRINT,'***** CHECK DATA FOR E-R-R-R-RS IN FIELD 1-6 *****
ENDIF
C
IF(ERR.EQ.13) THEN
PRINT,'***** MAXIMUM FREE VARIABLES ALLOWED = ',MAIFV,'*****
PRINT,'***** MAXIMUM FREE VARIABLES LIMIT EXCEEDED *****
ENDIF
IF(ERR.EQ.14) THEN
IF(PB(L).EQ.PT(ROW,COLUMN)) FLAG=1
808 CONTINUE
IF(FLAG.EQ.0) GOTO 802
804 CONTINUE
IF(PT(ROW,ROWSUM).EQ.(L-1)) THEN
   Q=ROW
   EXECUTE LOADPV
   GOTO 814
ENDIF
802 CONTINUE
ERR=13
EXECUTE ERPARA
815 Q=ROW
EXECUTE ADDRROW
EXECUTE LOADPV
C
814 CONTINUE
EXECUTE CLNPB
ENDBLOCK
C-------------------------------------------------------------
REMOTE BLOCK LOADPV
'PVB(Q)=PVB(Q)+COEFF*1.0
ENDBLOCK
C-------------------------------------------------------------
REMOTE BLOCK ADDRROW
DO 812 LR=1,MAXPN
PT(Q,LR)=PB(LR)
IF(PB(LR).NE.0) PT(Q,ROWSUM)=PT(Q,ROWSUM)+1
812 CONTINUE
ROWNUM=ROWNUM+1
ENDBLOCK
C-------------------------------------------------------------
REMOTE BLOCK TOOLDB
PRINT,'***** TOOL DATABASE READ FROM THE CARDS *****
DO 578 I=1,PTYPE
   LIM=OPN(I)
DO 580 K=1,LIM
   PRINT 584,FACT(I),K,TDI(FACT(I)),K,T,I,MXTDB
584 FORMAT(12,2I4,12,4I1,100(I))
580 CONTINUE
578 CONTINUE
PRINT,'***** TOOL SLOT REQUIREMENTS *****
DO 582 I=1,MXTDB
   PRINT,'TOOL NUMBER ',I,' SLOTS REQUIRED = ',NY(I)
582 CONTINUE
ENDBLOCK
C-------------------------------------------------------------
REMOTE BLOCK GRLOC
IF(PROC.EQ.1) EXECUTE FALOC
IF(PROC.EQ.2) EXECUTE COMTL
ENDBLOCK
C-------------------------------------------------------------
REMOTE BLOCK WRITTB
DO 1220 IWB=1,MATE
1220 CONTINUE
ENDBLOCK

C----------------------------------------
REMOTE BLOCK SLOTRO
TSM=0
DO 1222 ISL=1,MAXTD
IF(TOD(C1,C3,ISL).NE.0) TSM=TSM+NY(ISL)
1222 CONTINUE
IF(TSUM.LE.0) THEN
  ERR=15
  EXECUTE ERPARA
ENDIF
ENDBLOCK

C----------------------------------------
REMOTE BLOCK LIN12
COEFF=1.0
DO 2100 INDEX=1,TOTIV
GAMMAS=0
DO 2110 I=1,ROWNUM
IF(PT(I,ROWBUF).EQ.999.0R,PT(I,ROWSUM).NE.2)
  GOTO 2110
IF(PT(I,1).NE.INDEX.AND.PT(I,2).NE.INDEX) GOTO 2110
Y(PT(I,1))=Y(PT(I,1))1.0
Y(PT(I,2))=Y(PT(I,2))1.0
COEFF=1.0
Q=1
EXECUTE LOADPV
PT(I,ROWBUF)=999
GAMMAS=GAMMAS1
2110 CONTINUE
IF(GAMMAS.EQ.0) GOTO 2100
CTYPE='LE'
RHS=FLOAT(GAMMAS)
EXECUTE PURGE
NCST(5)=NCST(5)1
2100 CONTINUE
C
COEFF=1.0
DO 1230 I=1,ROWNUM
IF(PT(I,ROWBUF).EQ.999) GOTO 1230
DO 1232 J=1,MAXPBN
IF(PT(I,J).EQ.0) GOTO 1234
Y(PT(I,J))=Y(PT(I,J))1.0
1232 CONTINUE
1234 Q=1
COEFF=1.0
EXECUTE LOADPV
C

REMOTE BLOCK LIN2
COEFF=1.0
DO 1240 LZ=1,TOTIV
    NG=0
    DO 1242 I=1,ROWNUM
       DO 1244 J=1,MAXPN
          IF(P1(J,J).EQ.0) GOTO 1242
          IF(P1(J,J).EQ.LZ) THEN
             NG=NG+1
             Q=I
          ENDIF
       END
       1244 CONTINUE
    END
1242 CONTINUE
IF(NG.NE.0) THEN
   Y(LZ)=Y(LZ)+(-1.0)*NG
   CTYPE='LE'
   RHS=0.0
   EXECUTE PURGE
   NCST(6)=NCST(6)+1
ENDIF
1240 CONTINUE
IH=9
EXECUTE MILI
ENDBLOCK

C

REMOTE BLOCK LIN11
COEFF=1.0
DO 1600 IL3=1,ROWNUM
   ILEN=PT(IL3,ROWSUM)
   DO 1602 IL4=1,ILEN
      Y(PT(IL3,IL4))=1.0
   END
   1602 CONTINUE
COEFF=-1.0
D=IL3
EXECUTE LOADPV
CTYPE='LE'
RHS=FLOAT(PT(IL3,ROWSUM)-1)
EXECUTE PURGE
NCST(5) = NCST(5) + 1
1600 CONTINUE
IH = 8
EXECUTE MILI
ENDBLOCK

C-------------------------
REMOTE BLOCK LIN21
COEFF = 1.0
DO 1610 IL3 = 1, ROWNUM
ILEN = PT(IL3, ROWSUM)
DO 1615 IL4 = 1, ILEN
Y(PT(IL3, IL4)) = -1.0
COEFF = 1.0
Q = IL3
EXECUTE LOADPY
CTYPE = 'LE'
RHS = 0.0
EXECUTE PURGE
NCST(6) = NCST(6) + 1
1615 CONTINUE
1610 CONTINUE
IH = 9
EXECUTE MILI
ENDBLOCK

C-------------------------
REMOTE BLOCK SUMMARY
WRITE(6, 1270)
1270 FORMAT(11H1, //, 20X,'*** OPERATION ALLOCATION PROBLEM IN FMS
***', //)
IF (OBJECT.EQ.1) THEN
PRINT, 'TYPE OF OBJECTIVE FUNCTION : MINIMIZE REFINISHING'
ELSE
PRINT, 'TYPE OF OBJECTIVE FUNCTION : MINIMIZE TRANSPORT'
ENDIF
PRINT, ''
PRINT, ''
PRINT, 'OBJECTIVE FUNCTION ROWS :
1 NCST(7)
PRINT, 'TYPE-I CONSTRAINTS: ONE MACHINE PER OPERATION :
1 NCST(1)
PRINT, 'TYPE-II CONSTRAINTS: FORCED ASSIGNMENTS :
1 NCST(2)
PRINT, 'TYPE-III CONSTRAINTS: SCARCE TOOLS :
1 NCST(3)
PRINT, 'TYPE-IV CONSTRAINTS: TOOL MAGAZINE CAPACITIES :
1 NCST(4)
PRINT, 'LINEARIZATION STRATEGY CHOSEN :
1 STRATEGY
PRINT, 'LINEARIZATION CONSTRAINTS TYPE-I :
1 NCST(5)
PRINT, 'LINEARIZATION CONSTRAINTS TYPE-II :
NCST(6)
PRINT,'INTEGRALITY CONSTRAINT',

NCST(8)
PRINT,'UPPERBOUNDS ON VARIABLES CONSTRAINT',

NCST(9)
PRINT,' ',
PRINT,' ',
PRINT,'ALLOCATION VARIABLES (INTEGER)',

NVAR1
NVAR2=NVAR2-NVAR1
PRINT,'ALLOCATION VARIABLES (FREE)',

NVAR3
PRINT,'LINEARIZATION VARIABLES (FREE)',

ROWNUM
IF(OBJECT.EQ.1) THEN
PRINT,'LB ON OBJECTIVE FUNCTION FROM FITURING INFO',
OBJLB
ENDIF
WRITE(6,1280)
1280 FORMAT(1H1,///)
ENDBLOCK

C-----------------------------------------------------

REMOTE BLOCK PRODESC
WRITE(6,1520)
1520 FORMAT(1H1,///)
   "DESCRIPTION OF THE PROBLEM FORMULATED : ',//)

C

PRINT,'PART TYPES CONSIDERED FOR FORMULATION ARE:
DO 1420 IS=1,PTYPE
PRINT,'PART TYPE=',PACT(IS),' NUMBER OF OPNS=',OPN(IS),
   'WEIGHTAGE=',WET(IS)
1420 CONTINUE
PRINT,' ',
PRINT,' ',

C

PRINT,'MACHINE INFORMATION :
PRINT,' ',
DO 1522 IS=1,MACH
WRITE(6,1524) IS,TJ(IS),OLD(IS),DUL(IS)
1524 FORMAT(1H1, 'MACHINE #',I3,' SLOTS = ',I3,' LOAD DIST. = ',I3,
   ' UNLOAD DIST. = ',I3)
1522 CONTINUE
IF(IS.GT.0) THEN
PRINT,'GROUPED OPERATIONS :'
PRINT,' ',
DO 1526 IS=1,IL6
DO 1528 JPR=1,MAXOPN
IF(GASTOR(IS,JPR,1).EQ.0) GOTO 1528
WRITE(6,1530) GASTOR(IS,JPR,1),GASTOR(IS,JPR,2)
1530 FORMAT(1I,'PART TYPE : ',I3,' OPERATION # ',I3)
CONTINUE
PRINT,
CONTINUE
PRINT,
PRINT,
ENDIF
IF(NCST(3).GT.0) THEN
PRINT,
PRINT,'SCRE6 TOOLS :' PRINT,
DO 1532 ISI=1,MAKPNP
IF(TLSTOR(IS).EQ.0) GOTO 1532
PRINT,'TOOl NUMBER :',TLSTOR(IS)
1532 CONTINUE
ENDIF
ENDBLOCK

REMOTE BLOCK VARTAB
WRITE(6,1546)
1546 FORMAT(1H1,//)
ISI=0
PRINT,'VARIABLE CORRESPONDENCE TABLE :
PRINT,
PRINT,'--Y----I----J----K----YOUT--' DO 1540 ISI=1,TOTIV
IF(POS(NI(IS),NJ(IS),NK(IS)).NE.0) THEN
ISI=ISI+1
PRINT 1542, ISI,NI(IS),NJ(IS),NK(IS),ISI ELSE
PRINT 1544, IS,NI(IS),NJ(IS),NK(IS)
1542 FORMAT(2I17,2X,3(I3,I1),I1,I3)
1544 FORMAT(2I15,2X,3(I3,I1),I1,'***')
ENDIF
1540 CONTINUE
ENDBLOCK

REMOTE BLOCK TPCHK
KU=0
KTUPLE=0
DO 1820 T=1,MAXTDB
'DO 1820 KUP=1,NTB
KU=KU+ TDB(TB1KUP,T),TB(KUP,2),T
1820 CONTINUE
IF(KU.GT.KTUPLE) KTUPLE=KU
KU=0
1810 CONTINUE
ENDBLOCK

CONTINUE
EXECUTE SUMMARY
EXECUTE PRDESC
EXECUTE PMAT
EXECUTE VARTAB
STOP
END

C THE INPUT DATA TO THE PROGRAM FOLLOWS.

ENTRY
PART TYPE 1 : 001 003 0.00
PART TYPE 3 : 003 004 0.00
PART TYPE 4 : 004 003 0.00
PART TYPE 5 : 005 004 0.00
----------------------------------------- 000 000 0.00

TOOL DATABASE:
01 01 41 12
01 02 01 02 13 14 23 28 48 49
01 03 36 37
02 01 16
02 02 18
02 03 02 04 25 29
02 04 03 13 14 24
03 01 04 16 32
03 02 13 14 17 41 43
03 03 13 14 44
03 04 03 11 12 24
04 01 05 15 18 20 30 38 50
04 02 04 06 16 19 20 31 32 39 40 50 51 52
04 03 04 07 53
05 01 18 19 20
05 02 04 11 12 48
05 03 16 19 20 42 48
05 04 39 40
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 0 0 1 1 1 1 1 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

MACHINE DATA
MD SLOT DL LD
01 015 020 020
02 020 020 020
03 020 020 020
04 025 025 025
05 030 025 025
06 070 020 020
00 000 000 000
POSSIBILITY MATRIX:

```
01 01 : 01 01 00 01 01 00
01 02 : 01 01 01 01 01 01
01 03 : 00 01 00 01 00 00
02 01 : 01 01 00 01 01 00
02 02 : 01 01 01 01 01 01
02 03 : 01 01 01 01 01 01
02 04 : 01 01 01 01 01 01
03 01 : 00 00 01 01 00 01
03 02 : 00 01 00 00 00 00
03 03 : 01 01 00 00 01 00
03 04 : 01 01 00 00 01 00
04 01 : 01 01 01 01 01 01
04 02 : 01 01 01 01 01 01
04 03 : 01 01 00 00 01 00
05 01 : 01 01 00 00 01 00
05 02 : 01 01 00 00 01 00
05 03 : 01 01 01 01 01 01
05 04 : 01 01 01 01 01 01
00 00 : 00 00 00 00 00 00
```

--------

FITURING INFORMATION

1 02 01 01 02
1 02 01 02 02
1 02 01 03 02
1 02 01 04 02
1 02 01 05 02
1 02 01 06 02
1 02 02 01 02
1 02 02 02 02
1 02 02 03 02
1 02 02 04 02
1 02 02 05 02
1 02 02 06 02
1 02 04 01 02
1 02 04 02 02
1 02 04 03 02
1 02 04 04 02
1 02 04 05 02
1 02 04 06 02
1 02 05 01 02
1 02 05 02 02
1 02 05 03 02
1 02 05 04 02
1 02 05 05 02
1 02 05 06 02
1 03 01 02 00
1 03 01 04 02
1 03 02 02 00
1 03 02 04 02
```
VITA AUCTORIS

1958
Born in Pune, India, on the 6th of January.

1974
Completed higher secondary education from New English School, Pune, India.

1980
Graduated from the University of Poona, Pune, India, with a Bachelor's degree in Mechanical Engineering.

1982
Graduated from the National Institute for Training in Industrial Engineering, Bombay, India, with a Post Graduate Diploma in Industrial Engineering.

1982-84
Worked as Planning Engineer in the Auto Planning division of the Tata Engineering and Locomotive Company, Pune, India.

1986
Currently a candidate for the M.A.Sc. degree in Industrial Engineering at the University of Windsor, Windsor, Ontario, Canada.