Cell formation in flexible manufacturing systems.

Gajanana. Nadoli
University of Windsor

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

Recommended Citation
https://scholar.uwindsor.ca/etd/1378

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 REÇUE
CELL FORMATION IN FLEXIBLE MANUFACTURING SYSTEMS

by

Gajanana Nadoli

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.

L'autorisation a été accordée à la Bibliothèque nationale du Canada de microfilm cette thèse et de prêter ou de vendre des exemplaires du film.

L'auteur (titulaire du droit d'auteur) se réserve les autres droits de publication; ni la thèse ni de longs extraits de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation écrite.

ISBN 0-315-31996-8
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.

Gajanana Nadoli

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.

Gajanana Nadoli
The University of Windsor requires the signature of all persons using or photocopying this thesis. Please sign below, and give address and date.
ABSTRACT

In recent years the concept of flexible manufacturing systems (FMS) has emerged as a viable answer to the problems of low volume, medium variety production. The technological sophistication and correspondingly high investment in these systems necessitate sufficient planning effort both in the implementation and the operation stages. This research deals with the initial specification decisions in the pre-production planning stage. The cellular configuration of FMS is considered, in which a group of machines is dedicated to the manufacture of a particular family of parts. Two of the problems in cell formation viz., part family formation and machine group allocation are formulated. A fractional programming model defined on zero-one integer variables has been proposed for the part family formation. The parts are grouped based on their processing similarity. The machine group allocation problem is formulated as a zero-one integer program, to maximize the routing diversity available for the parts in different families. The availability of alternative routings has been considered in cell formation. The application of the formulations has been illustrated through a number of examples using realistic data.
ACKNOWLEDGEMENTS

I would like to take this opportunity to express my gratitude towards Dr. S.P. Dutta and Dr. R.S. Lashkari for their guidance and support during the course of this research. I would like to thank Dr. Y. Aneja for sparing his time to give useful suggestions. A special note of thanks to Jacque Mummery and Tom Williams for their help from time to time. I would also like to thank the consultants of the University Computer Centre for their help.
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. TYPICAL ARRANGEMENT OF THE MACHINES IN MANUFACTURING SYSTEMS</td>
<td>6</td>
</tr>
<tr>
<td>2. SCHEMATIC DIAGRAM OF A CELLULAR FMS</td>
<td>11</td>
</tr>
<tr>
<td>3. MATRIX MANIPULATION METHODS</td>
<td>24</td>
</tr>
<tr>
<td>4. FLOW CHART OF THE ALGORITHM</td>
<td>54</td>
</tr>
</tbody>
</table>
# LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>ILLUSTRATION OF THE REGION (L,U) FOR P(R)</td>
<td>41</td>
</tr>
<tr>
<td>2.</td>
<td>PART PAIR (1,2)</td>
<td>44</td>
</tr>
<tr>
<td>3.</td>
<td>IP, THE MINIMUM NUMBER OF PART PAIRS</td>
<td>44</td>
</tr>
<tr>
<td>4.</td>
<td>PRODUCT TERMS INDICATING MULTIPLE ALLOCATIONS OF MACHINES</td>
<td>66</td>
</tr>
<tr>
<td>5.</td>
<td>PENALTY WEIGHTS TO MULTIPLE ALLOCATIONS OF THE MACHINES</td>
<td>68</td>
</tr>
<tr>
<td>6.</td>
<td>THE PROCESSING REQUIREMENTS FOR 15 PARTS</td>
<td>72</td>
</tr>
<tr>
<td>7.</td>
<td>BOUNDS ON THE OBJECTIVE FUNCTION FOR DIFFERENT VALUES OF R</td>
<td>76</td>
</tr>
<tr>
<td>8.</td>
<td>SUMMARY OF TRIALS WITH DIFFERENT STARTING CONFIGURATIONS FOR FIFTEEN PARTS EXAMPLE</td>
<td>78</td>
</tr>
<tr>
<td>9.</td>
<td>ITERATION LOG FOR THE APPROXIMATION PROCEDURE</td>
<td>79</td>
</tr>
<tr>
<td>10.</td>
<td>TYPICAL SOLUTION TIMES FOR THE APPROXIMATION PROCEDURE ITERATIONS</td>
<td>85</td>
</tr>
<tr>
<td>11.</td>
<td>MACHINE ROUTING DATA</td>
<td>89</td>
</tr>
<tr>
<td>12.</td>
<td>LIST OF MULTIPLICATION TERMS INDICATING THE ROUTINGS FOR PARTS</td>
<td>90</td>
</tr>
</tbody>
</table>
# TABLE OF CONTENTS

ABSTRACT .................................................. vi

ACKNOWLEDGEMENTS ............................................ vii

LIST OF FIGURES .............................................. viii

LIST OF TABLES ................................................ ix

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>1</td>
</tr>
<tr>
<td>II</td>
<td>4</td>
</tr>
<tr>
<td>2.1</td>
<td>4</td>
</tr>
<tr>
<td>2.1.1</td>
<td>4</td>
</tr>
<tr>
<td>2.1.2</td>
<td>5</td>
</tr>
<tr>
<td>2.1.3</td>
<td>8</td>
</tr>
<tr>
<td>2.1.4</td>
<td>9</td>
</tr>
<tr>
<td>2.1.5</td>
<td>10</td>
</tr>
<tr>
<td>2.2</td>
<td>12</td>
</tr>
<tr>
<td>2.2.1</td>
<td>12</td>
</tr>
<tr>
<td>2.2.2</td>
<td>14</td>
</tr>
<tr>
<td>2.2.2.1</td>
<td>15</td>
</tr>
<tr>
<td>III</td>
<td>18</td>
</tr>
<tr>
<td>IV</td>
<td>28</td>
</tr>
<tr>
<td>4.1</td>
<td>28</td>
</tr>
<tr>
<td>4.1.1</td>
<td>28</td>
</tr>
<tr>
<td>4.1.2</td>
<td>31</td>
</tr>
<tr>
<td>4.1.3</td>
<td>32</td>
</tr>
<tr>
<td>4.1.4</td>
<td>33</td>
</tr>
<tr>
<td>4.2</td>
<td>35</td>
</tr>
<tr>
<td>4.2.1</td>
<td>35</td>
</tr>
<tr>
<td>4.2.2</td>
<td>39</td>
</tr>
</tbody>
</table>

- x -
<table>
<thead>
<tr>
<th>Chapter</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.2.3 Establishing Lower Bound (LBR) and Upper Bound (UBR) for Z(R,X)</td>
<td>40</td>
</tr>
<tr>
<td>4.2.3.1 Constraints on the Function Z(R,X)</td>
<td>40</td>
</tr>
<tr>
<td>4.2.3.2 Coefficients of the Function Z(R,X)</td>
<td>45</td>
</tr>
<tr>
<td>4.2.3.3 Algorithm for Finding LBR and UBR</td>
<td>46</td>
</tr>
<tr>
<td>4.2.4 Summary of the Steps for Solving the Formulation</td>
<td>47</td>
</tr>
<tr>
<td>4.3 Approximation Procedure</td>
<td>47</td>
</tr>
<tr>
<td>4.3.1 Need for Finding a 'Good' Initial Solution</td>
<td>47</td>
</tr>
<tr>
<td>4.3.2 Principle</td>
<td>49</td>
</tr>
<tr>
<td>4.2.3.1 Single Reallocation</td>
<td>49</td>
</tr>
<tr>
<td>4.2.3.2 Multiple Reallocation</td>
<td>50</td>
</tr>
<tr>
<td>4.3.3 Algorithm</td>
<td>52</td>
</tr>
<tr>
<td>V MACHINE GROUP ALLOCATION</td>
<td>55</td>
</tr>
<tr>
<td>5.1 Formulation</td>
<td>55</td>
</tr>
<tr>
<td>5.1.1 Statement of the Problem</td>
<td>55</td>
</tr>
<tr>
<td>5.1.2 Objective</td>
<td>56</td>
</tr>
<tr>
<td>5.1.3 Concept of alternative Routings</td>
<td>56</td>
</tr>
<tr>
<td>5.1.4 Formulation</td>
<td>59</td>
</tr>
<tr>
<td>5.2 Solution Procedure</td>
<td>62</td>
</tr>
<tr>
<td>5.2.1 Linearization of Product Terms</td>
<td>62</td>
</tr>
<tr>
<td>5.2.2 Some Reductions in the Number of Product Terms</td>
<td>63</td>
</tr>
<tr>
<td>5.3 Infeasibility in the Machine Group Allocation</td>
<td>64</td>
</tr>
<tr>
<td>5.3.1 Multiple Family allocation of Some Machine(s)</td>
<td>64</td>
</tr>
<tr>
<td>5.3.2 Mathematical Model to Identify the machines Causing Infeasibility</td>
<td>65</td>
</tr>
<tr>
<td>5.4 Summary of steps for Solving the Formulation</td>
<td>69</td>
</tr>
<tr>
<td>VI APPLICATION OF THE FORMULATIONS</td>
<td>70</td>
</tr>
<tr>
<td>6.1 Problem data</td>
<td>70</td>
</tr>
<tr>
<td>6.1.1 Parts and Machines</td>
<td>70</td>
</tr>
<tr>
<td>Chapter</td>
<td>Page</td>
</tr>
<tr>
<td>---------</td>
<td>------</td>
</tr>
<tr>
<td>6.1.1.1 Parts Spectrum</td>
<td>71</td>
</tr>
<tr>
<td>6.1.1.2 Machines</td>
<td>71</td>
</tr>
<tr>
<td>6.1.2 Generation of Problem Input for an IP Routine</td>
<td>73</td>
</tr>
<tr>
<td>6.2 Part Family Formation - An Example</td>
<td>73</td>
</tr>
<tr>
<td>6.2.1 Finding the Interval ((L,U))</td>
<td>75</td>
</tr>
<tr>
<td>6.2.2 Initial Solution through Approximation Procedure</td>
<td>77</td>
</tr>
<tr>
<td>6.2.3 Search Log</td>
<td>82</td>
</tr>
<tr>
<td>6.2.4 Some Computational Considerations</td>
<td>84</td>
</tr>
<tr>
<td>6.3 Machine Group Allocation - An Example</td>
<td>88</td>
</tr>
<tr>
<td>6.3.1 Routing Information</td>
<td>88</td>
</tr>
<tr>
<td>6.3.2 Solution Procedure</td>
<td>88</td>
</tr>
<tr>
<td>6.4 Discussion of Results</td>
<td>91</td>
</tr>
</tbody>
</table>

VII SUMMARY | 96 |

REFERENCES |

APPENDICES | 101 |

A. ALTERNATIVE DEFINITIONS FOR THE OVERALL DISSIMILARITY COEFFICIENT | 101 |

B. PART SKETCHES AND PROCESS DETAILS | 104 |

C. COMPUTER PROGRAM LISTINGS | 135 |

D. ITERATION LOGS FOR DIFFERENT TRIALS OF APPROXIMATION PROCEDURE | 178 |

VITA AUCTORIS | 188 |
Chapter I

INTRODUCTION

In recent years the concept of Flexible Manufacturing systems (FMS) has emerged as a viable answer to the problems of low volume, medium variety production. These systems offer automated and flexible operation, coupled with the optimum exploitation of resources. It is acknowledged that an integrated approach to parts manufacture from design conceptualization to operation stage is the pre-condition for the success of such systems.

The technological sophistication and the correspondingly high investment in these systems necessitate sufficient planning efforts both in the implementation and operation stages.

The efficient system design to facilitate the gradual implementation is very important. It can be achieved by conceiving the FMS to be made up of different groups of machines. In Group Technology terms these groups are known as cells. Chapter 2 briefly explains the flexible manufacturing systems, the types of arrangements of the manufacturing set ups, the advantages of cellular
arrangement and the system configuration under consideration.

The objective of this research is to model two of the problems related to the cell formation in FMS.

1) The part family formation

ii) The machine group allocation.

A review of the previous research is given in Chapter 3.

The formulations of the above two problems are explained in Chapters 4 and 5 respectively. A fractional programming model for minimizing the processing dissimilarities between different part types has been proposed for the part family formation. A solution procedure is developed for this model taking into consideration the nature of the objective function. The procedure suitably adopts a general principle of search for finding the optimal solution. The infeasibility in allocation caused by restricting each of the machines to only one family (unique allocation) has been resolved. A simple mathematical model identifies the machines causing the infeasibility and the unique allocation constraint is relaxed for these machines.

Realistic data representing typical part and machine varieties have been considered in solving a number of problems to illustrate the formulations. The results are explained in Chapter 6. A summary of the research findings
has been presented in Chapter 7.
Chapter II

SYSTEM DESCRIPTION AND OBJECTIVES

2.1 Flexible manufacturing systems

2.1.1 Definitions

An FMS is an automated, batch manufacturing system consisting of a set of numerically controlled machine tools with automatic tool changing capabilities. A computer controlled material handling system transports the parts from machine to machine.

These systems have been given a variety of names — Computerized Manufacturing Systems (CMS) and Variable Manufacturing Systems (VMS), for example, and have in fact been designed in a variety of configurations.

The cellular configuration of FMS is considered in this research. The definitions of the terminology [22], with reference to this type of configuration are given below.

Flexible Manufacturing Module (FMM): An FMM is defined as a Numerically Controlled Machine augmented by a
part buffer, a tool changer, a pallet changer etc. An FMM will be referred to as a machine throughout this report.

Flexible Manufacturing Cell (FMC): An FMC consists of several machines, capable of producing a range of parts. Each of these FMCs are organized as independent facility set-ups. The term cell has been borrowed from cellular manufacturing in the conventional systems. The FMCs are referred to as cells in the discussions to follow. An FMS can be considered to be consisting of cells. Many a times individual cells themselves are considered as systems, indicating the independent nature of these cells.

2.1.2 Cellular Configuration of FMS

There are various approaches to the arrangement of machines in a manufacturing system. In all the cases, it is necessary to conceive the system as a whole from design to installation.

The typical arrangements of the machines in the manufacturing systems are (Fig. 1):

i) Random: A number of machines are arranged in a rectangular shop. The disadvantage of this lay-out is that with larger number of machines, transfer paths are complicated and are likely to be longer than necessary.

ii) Functional: The machines are arranged according to function, such as turning, milling, boring and grinding, so
Figure 1. TYPICAL ARRANGEMENTS OF MACHINES IN MANUFACTURING SYSTEMS

Random Layout

Functional Layout

Modular Layout

Cellular Layout
that the workpieces flow through the shop from one section to another. The workpieces have to be moved many times between the sections, and material handling paths in this type of arrangement may be excessively long.

iii) Modular: Here identical modules perform similar processes in parallel. This layout is likely to result in some redundant capacity, but can be an alternative to the functional layout, while redundancy may make it easy to cope with critical jobs or unexpected problems.

iv) Cellular: In this arrangement each cell is dedicated to a certain group of parts. It is an extension of the Group Technology (GT) concept. The cellular system is likely to give the best match of machining capability to the processing of various workpieces [18].

Group technology is a manufacturing philosophy that seeks to rationalize small and medium sized batch production by capitalizing on the similarity between the parts. GT is applied with respect to two aspects of part characteristics viz., geometric features and processing requirements.

The geometric feature based grouping has been mainly a part of design standardization effort for the various shapes of the parts. The concept has recently been considered in the computer aided process planning area, where an attempt to relate the processing steps to the geometric features is made to develop computerized systems.
for generating the process plans [9].

The grouping of parts with respect to the processing requirements forms the basis of cellular arrangement of the machines. A manufacturing cell is designed to produce the parts with similar machining requirements. Due to similarity of the parts, change over of processing from one part type to another on the machines causes minimal disruptions in terms of tooling requirements. Section 2.2.1 describes this issue. The cellular arrangement is an attempt to achieve the advantages of mass production in small batch production. Several conventional systems have been installed based on this principle [17].

2.1.3 Operation of the system

The FMSs can be viewed as highly automated job shops. A typical sequence of events involved in processing of a part in an FMS is as follows:

When a part is scheduled for an operation on one of the available alternative machines, the part is fixtured on a pallet and transported to the machine. The machine on which this part is to be processed receives the necessary part programs. If certain tools are not available on the machine, the handling system transports those tools also to the machine [18]. Once the machining for that operation is finished, the part is moved for its next operation.
Since different parts are in production simultaneously, conflicts in the requirements arise. Among other things, the automated control has to consider the important issues of scheduling of parts, queues for the machines and machine break downs.

The automated operation requires the proper operation logic to be programmed into the system prior to the start of production. The precise anticipation of all the operational exigencies is necessary in such an operation mode.

2.1.4 Advantages of Cellular Configuration

Dividing the system into smaller sub-systems (cells) is essential due to the complexity of operation as indicated in section 2.1.3. Such a division can be viewed as a method of aggregation leading towards a reduction in the size of the planning and scheduling problems [17]. It is a normal practice to install a small system first and then to build up the complete system in due course. Since apart from the machines, the peripheral equipment themselves constitute a large investment, a phased plan is necessary to implement these systems.

An arrangement of machines in the form of a single system has the disadvantages of increased control problems, difficulty in keeping track of parts, increased part
movement distances and complex scheduling requirements.

The cellular arrangement of the machines in FMS has
the following advantages [17]:

i) Implied reduction in control.

ii) Reduced material handling.

iii) Quick change over of part types within a range of
parts.

iv) Better tooling control.

v) Reduced in process inventory.

vi) Reduced expediting.

A schematic diagram of the cellular FMS is shown in
Fig. 2.

2.1.5 **Design and operation problems in flexible
manufacturing systems**

The design and operation of any FMS involves a
variety of problems. Many of these problems are typical of
any manufacturing system. The classification and
description of these problems have been given in [29,35]:

i) Strategic Decision Problems

ii) Facility Planning Problems

iii) Intermediate decisions

iv) Dynamic Operations

The strategic decisions are concerned with such
problems as the financial and policy decisions in
Figure 2. SCHEMATIC DIAGRAM OF CELLULAR FMS

M - MACHINES
L - LOADING STATIONS
S -storages
P - MATERIAL HANDLING PATHS
implementing FMS.

The facility planning problems are concerned with decisions about initial specification and implementation of the production system. The initial specification decisions include the selection of the parts to be produced, machines & other peripherals and material handling system. The subsequent implementation decisions include layout of the machines, software development and the design of fixtures.

The intermediate range problems are the pre-production decisions on operation allocation, part mix ratio and allocation of other resources.

The dynamic operations refer to the control problems due to conflicts in the production requirements. These are the in-production decisions on part release rules into the system, scheduling, sequencing, etc.

2.2 Objectives of the Research

2.2.1 Statement of Objectives

The objective of this research is to solve two problems related to the cell formation in flexible manufacturing systems:

- Grouping of the parts into families.
- Allocation of machine groups to families.

These are the initial specification issues in the
pre-production planning stage of FMS.

It is important to consider the relevant criterion for grouping the parts into families. The criterion for organizing the cells for manufacturing the parts is based on the processing similarity of the parts. The lack of such similarity has an adverse effect on the operation of the cellular system.

The tools have to be changed intermittently in the tool magazines if parts with different processing requirements are manufactured in the cells. Each tool change puts a certain demand on the system resources for the following activities:

The measurement of cutter compensation and tool offsets (to be supplied to the machines) may have to be carried out when a tool is loaded on to or transferred between the machines. This puts a load on the metrology facilities in the system.

The tool loading is usually done manually in the present systems (although there is an attempt to make this automatic, the use of such automation is not yet widespread [18]) and the frequent tool changing interrupts the operation of the machines.

Frequent tool changing also results in a constant flow of tools within the shop competing with the parts for the resources (trolley, scheduling time on computer etc.). It has been found in some cases that the flow of tools
through the shop caused more problems than the flow of workpieces [18].

Hence the processing similarity is considered for grouping parts first to form the part families.

Once the parts constituting different families are determined, the machine group allocation problem will be solved. The objective of such allocation is to provide maximum number of alternative routings for the parts. The diversity in part routing is known to be a very helpful strategy in the operation of the system. It is possible to divert the parts to different machines when the designated machines break down or are busy serving some other parts.

The part family formation and machine group allocation problems are formulated in Chapters 4 and 5.

2.2.2 Typical Problem Situation

Most of the modern manufacturing plants have NC/CNC machines located randomly within the factory. Even though many such machines may be in operation, the net effect on production may not be as significant as can be expected with these versatile machines. The individual machines are really very efficient, but the way in which they are placed in the system may result in low utilization levels. They may be restricted by limitations such as production bottlenecks at other machines and material handling delays.
In this situation, since the NC/CNC machines, which are the major components of FMS are already available, there is an opportunity to reorganize the system into an independent cellular FMS. The machines in the cells can be linked together through a material handling system. Once such a strategic decision is taken, it becomes essential to analyse the part range under consideration to form families and then allot the available machines to each of the families. As mentioned earlier, the implementation can be done in phases, organizing one cell at a time.

2.2.2.1 Part range manufactured in FMS

When the FMS capacity augments the conventional capacity, the part variety chosen for manufacturing in FMS is restricted keeping in mind the need to utilize other high cost plant and auxiliary equipment. The parts chosen are high value, critical components required in the downstream production facilities. A fabrication shop supplying the finished parts to an assembly section is an example of such a situation, where certain parts in the final assembly are invariably in short supply due to the difficulties encountered in manufacturing them in the conventional shops.

This is clearly illustrated by the reports on the existing systems and the restricted component variety they
encompass [2].

The literature on FMS [2,18] and experience in a light engineering industry indicates that the parts selected for manufacturing on CNC machines (and hence in FMS) have, in general, the following characteristics:

1) The parts require a large number of processing steps. If loaded in a conventional machine shop these parts have to visit several machines, in most cases one machine carrying out one processing step. This results in a tremendous amount of handling and subsequently a tardy output from the shop. These parts are the right candidates to be manufactured in an FMS, since the CNC machines allow for a number of processing steps to be completed in one visit to the machine.

2) Heavy emphasis on the milling, drilling, boring and tapping. The existing systems indicate their strength in these processes basically due to the corresponding capabilities offered by the machining centres. "Difficult" processes such as grinding and honing, mass production oriented processes such as broaching and not-so-common production processes such as planing and shaping (shaper), if required on a part, are usually carried out on the facilities operating in tandem with, but outside, the FMS.
iii) Apart from the problem of excessive handling, the sheer difficulty involved in achieving the complicated process requirements of some parts (in conventional shops) makes them the automatic choice for manufacturing in FMS.

iv) The parts are mostly finished from raw casting state.
Chapter III

LITERATURE SURVEY

The problems of FMS design and operation have been considered using different Operations Research approaches. The major approaches used in the literature are Networks of Queues, Simulation, and Mathematical Programming.

The facilities design problem has two issues as mentioned earlier, the initial specification decisions and the subsequent implementation decisions. These decisions are generally one time decisions, especially the ones concerning the machines constituting the FMS cells. The implementation decisions about the number of pallets and the number of fixtures can be spread over the time of operation of the system.

The queueing network models provide some aggregate results and are perhaps helpful in the decision issues such as the number of pallets and the number of fixtures required in the system. The aggregation may not be acceptable for more specific decisions such as sequencing and scheduling of the parts and the number of buffer spaces required. Simulation is the approach for such problems.
The mathematical models are appropriate for the static decision issues of facility design, operation allocation in the planning stage and fixture & pallet allocation. In such pre-production planning decisions some criteria are used which have been proved to be effective either by experience or by theoretical research in the operation of the system. Providing alternative routings for parts, balancing workloads between machines, minimizing part handling distances, launching similar parts for production, etc., are some examples of such criteria. These would be basically indirect measures, which are recommended as static problem objectives.

Wilhelm and Sarin [35] provide a review about the issue of suitability and limitation of different modelling approaches.

In this research mathematical modelling has been adopted. The criteria adopted in this research are the processing similarity concept for part family formation and routing diversity concept for machine group allocation.

Buzacott and Shantikumar [5] have reported some simple models for the understanding of the FMS. Their approach is to consider the system as an automated job shop. The models are simple and aggregate in nature, but they demonstrate amongst other aspects the importance of diversity in job routing.

Chatterjee et. al [10] have developed a general
framework for manufacturing system specification. They present some scheme for manufacturing systems to identify critical distinctions between various types of manufacturing capabilities. They define manufacturing flexibility and identify the number of routings available for a part within a system as the routing flexibility.

Stecke [30] gives an analysis of FMS cell using the queueing network theory. It has been shown that the pooling of machines in FMS cells improves the output of the system. Under a separate study of a real system through simulation [31], the same result was obtained. The system showed maximum output through the pooling in combination with some scheduling rule. The pooling of machines with reference to an operation means that there is more than one machine available for that operation and the part routing can be through one of the available machines depending on the scheduling decisions in real time.

Thus, providing maximum number of alternative routings has been proved to be a good strategy in operating the system.

One of the principles in Group Technology is to restrict a machine to only one part family (unique allocation). Thus, a certain machine group is made available to the parts in a particular family. However, in practice, some exceptions do exist. The scarcity of certain machines may force the sharing of those machines by
more than one family of parts. Certain overlapping
referred to as 'cascading' is allowed in these situations.
This possibility has been incorporated in the formulation
of machine group allocation.

The literature on the grouping procedures is mostly
limited to the conventional systems.

There are two issues in the grouping; part
representation and grouping procedure based on this
representation of the part. However, as pointed out by
King and Nakornchai [20], in the past decade the emphasis
has slowly shifted from classification schemes per se to
the problem of developing methods for grouping. This has
happened mainly due to the realization that most of the
classification schemes have to be industry-specific anyway.

A review of the various grouping procedures is given
by King and Nakornchai [20]. Recent work in this area
includes [6], [8], [21] and [33]. The classification of the
available techniques is as follows:

1) Similarity Coefficient methods
2) Set theoretic methods
3) Evaluative methods
4) Other analytical methods.

Similarity coefficient is an approach drawn from
numerical taxonomy, and first suggested by McAuley [24].
The basis of the method is to measure the similarity
between each pair of machines and then to group the
machines based on their similarity measurement.

These methods are called "hierarchical clustering methods" and are based on some "threshold value" of coefficients. If a coefficient is less than a predetermined value, the coefficient will be ignored in the next stage of the algorithm. The selection of the threshold values is arbitrary. Rajagopalan and Batra [27] suggest a more systematic method of finding the threshold value; however, the arbitrary nature of the procedure still persists. The hierarchical grouping methods can be explained as follows:

First two parts are selected which have the greatest similarity to form the nucleus of the first group. A third part is added which has the most similarity with the first two. The fourth is added which has the most similarity with the first three and so on. At any stage, if there is no part which has a similarity above a particular level with the parts in the first cluster, a new cluster is formed with the remaining parts in the same manner.

Set theoretic method has been developed by Purcheck [25]. This method considers the lists of machines required for the parts as sets and does set union operations on them. This is a heuristic method for grouping the machines and parts.

Evaluative methods are based on the Production Flow Analysis [4], and basically use the judgement of the
analyst. The main feature of the evaluative approach is that it involves listing of components in different ways in the expectation that the groups can be found by careful inspection. This requires manual intervention to identify groups at each stage.

The other analytical methods are based on machine component matrix manipulation. King and Nakornchai [20] and Chan and Milner [7] reported algorithms using this approach. The procedure developed in [33] for finding the bottleneck machines also is based on the matrix representation. Some criticism about King and Nakornchai's algorithm is given in [33]. The principle used is to improve a criterion starting from initial grouping, through some manipulations in the grouping using graph theory. Figure 3 (a) illustrates the typical machine-component matrix used by these methods. In this example, the machines are labelled from A to E and the parts from 1 to 6. An entry of 1 in cell (i, j) indicates that some operation of part j requires processing on machine i, whereas a blank entry means that it does not. The cell entries of 1 are spread around the matrix in a random fashion, so that no particular pattern of machine component grouping is apparent.

Figure 3 (b) shows the same matrix, but after several exchanges of the relative positions of both rows and columns. It will be seen that the original cell entries of...
**MATRIX MANIPULATION METHODS**

**Figure 3(a)**

<table>
<thead>
<tr>
<th>MACHINES</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>

**Figure 3(b)**

<table>
<thead>
<tr>
<th>MACHINES</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

**PARTS**

1 2 3 4 5 6

6 5 3 2 4 1
Fig. 3 (a) are preserved unchanged, but now, two machine component groupings (B,D,6,5,3) and (E,A,C,2,4,1) emerge naturally along the diagonal of the matrix as a result of a particular arrangement of the rows and columns of the original matrix. In some cases for geometric feature based grouping instead of having machine-component matrix, a code-component matrix is formed with the same basic idea of grouping the components and features together. The matrix manipulation methods are not mathematically rigorous [23].

Clustering is basically a yes/no type decision of allotting a part to a cluster. A 0-1 integer programming approach for the grouping of the parts has been reported by Kusiak [23], which employs a statistical clustering method [1]. This method considers the "distance" between the parts and then considers each part as the "median" of the cluster in the formulation. These concepts of "distance" and "median" are vaguely defined. The integer programming approach has also been used for grouping based on the geometric features.

The literature survey indicates that:

1) In the grouping methods reported it is assumed that the operations are restricted to one machine. Based on this single and fixed machine allocation information for the operations a one to one relationship between the parts and machines is defined in the form of a matrix. This leads to a
simultaneous grouping of parts and the machines.

ii) The (dis)similarity coefficients and measures of (dis)similarity between two parts, have been adopted with the arbitrary specification of some "cut off" values as the basis for grouping.

In conventional systems assumption (i) can be justified by noting that each operation on a part is generally restricted to one machine. The machine assignment for different operations of the parts acts as the basis for simultaneous grouping of parts and machines.

This assumption is not applicable to flexible manufacturing systems since each operation on a particular part can be performed on alternative machines. In this case, the processing similarity between the parts should be determined by using the basic information about the processing steps required to manufacture the parts.

The necessity of achieving homogeneity amongst the parts to be produced in an FMS cell, as explained in Chapter 2 is incorporated in the model by defining a dissimilarity coefficient. This coefficient is defined using 0-1 decision variables and is used as the objective function in a fractional programming model.

The procedure developed for cell formation groups the parts first based on the similarity of the processes and subsequently allots the machines to each of the part groups (families).
In the problem of allocation of machines to the part families the concept of providing routing diversity for the parts has been used as the objective function.
Chapter IV

PART FAMILY FORMATION

The mathematical formulation of the part family formation problem is discussed in this Chapter. First, the criterion for grouping based on the manufacturing attributes is explained. The objective function of the formulation is fractional defined on zero-one integer variables. A solution procedure for this situation is outlined. Due to the computational difficulty in solving this model for larger problems, an approximation procedure that yields a good initial solution is developed.

4.1 Formulation

4.1.1 Statement of the Problem

The part family formation is considered with respect to manufacturing attributes for eventually forming the cells.

The objective is to group the parts into part families based on their processing similarities. The
similarity based grouping is done to achieve minimum disruptions in the production within the cells as the batches of different parts are launched into production.

Notations Used

N  - Number of parts to be grouped
K  - Number of families
ij  - Index for the part pair (i, j) (i=1,2...N-1 ; j=i+1,i+2...N)
dij  - Number of dissimilar processes between part i and part j
sij  - Number of similar processes between part i and part j
DISij  - Dissimilarity coefficient between part i and part j
DCk  - Dissimilarity coefficient for family k
Dk  - Contribution of family k to the value of CDC
MF  - Family having highest value of Dk
CDC  - Dissimilarity coefficient for a configuration of part families
A(X)  - Linearized numerator of the coefficient CDC
B(X)  - Linearized denominator of the coefficient CDC
P(.)  - Minimization problem of CDC
R  - Parameter in the search procedure for optimal solution to P(.)
- 30 -

\( P(R) \) = Transformed minimization problem with parametric
objective function

\( Z(R,X) \) = Objective function of the problem \( P(R) \)

\( C \) = Constraint set of the problems \( P(\cdot) \) and \( P(R) \)

\( C_1 \) = Reduced constraint set for problem \( P(R) \)

\( Z(R,X) \) = Minimum of the objective function \( Z(R,X) \) subject
to \( C \)

\( \bar{Z}(R,X) \) = Maximum of the objective function \( Z(R,X) \) subject
to \( C \)

\( L_B \) = Bound on function \( Z(R,X) \) for minimization

\( U_B \) = Bound on function \( Z(R,X) \) for maximization

\( (L,U) \) = A range of values of \( R \) established such that
\[ L < R^* < U \]

\( C_{ij} \) = Coefficients of \( M_{ijk} \) variables in function
\( Z(R,X) \). (For convenience the superscript is
dropped and the coefficient is denoted by \( C_{ij} \)).

\( J \) = Set of \( C_{ij} \) s for all \((i,j)\)

\( N_P \) = Number of positive \( C_{ij} \) s in the set \( J \)

\( N_N \) = Number of negative \( C_{ij} \) s in the set \( J \)

Decision variables:

\( X_{ik} \) =
\[
= \begin{cases} 
1 & \text{if the part } i \text{ is included in family } k \\
0 & \text{if the part } i \text{ is not included in family } k 
\end{cases}
\]

These variables for all \((i,k)\) are denoted by \((X)\).

\( M_{ijk} \) = Linearization variable introduced to replace the
product term \( X_{ik} \cdot X_{jk} \)
4.1.2 **Criterion for grouping**

As stated earlier, the objective of the part family formation problem is to group the parts with similar processing requirements. For a part pair \((i, j)\) in a particular family, we would like to have a low ratio of \(d_{ij}/s_{ij}\), indicating that the parts \(i\) and \(j\) have more operations in common than dissimilar operations.

**An Example:**

Consider the part pair \((i, j)\) having the processing requirements as shown:

<table>
<thead>
<tr>
<th>Processes</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>part (i)</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>part (j)</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

For this part pair, \(s_{ij} = 3\) (processes 1, 4 and 7) and \(d_{ij} = 4\) (processes 2, 3, 5 and 6).

The dissimilarity between two parts is relevant only when they are grouped together into the same part family. The dissimilarity of two parts in different families is of no concern, since these parts are manufactured in different cells. The grouping should be done such that within the families formed, parts have the minimum dissimilarities and the maximum similarities in terms of processing requirements.

Based on this concept, the coefficient of dissimilarity between part \(i\) and part \(j\) is defined as:
\[ \text{DIS}_{ij} = \sum_{k=1}^{K} \left( \frac{d_{ij}}{s_{ij}} \right) \cdot X_{ik} \cdot X_{jk} \]  \hspace{1cm} (1)

This coefficient is used as the basis for defining the objective function for grouping the parts. The value of \( \text{DIS}_{ij} \) would be \( \frac{d_{ij}}{s_{ij}} \) or 0 respectively, depending on whether parts \( i \) and \( j \) are grouped in same family \( k \) or not.

4.1.3 Definition of Dissimilarity Coefficients

The objective function for the part family formation would be the minimization of an overall measure of processing dissimilarity between the parts. The definition of such a measure considered in this research is explained next. It represents the overall average of the pairwise dissimilarity coefficients. This is similar to the coefficient considered in [12]. Alternative representations for the overall measure of processing dissimilarity are indicated in Appendix A.

The dissimilarity coefficients are defined for each of the families and for the overall partitioning of the parts into families.

1) Dissimilarity coefficient for the family:

The average of the pairwise dissimilarity coefficients of all the parts in family \( k \) is given in Eqn.
(2). A high value of this coefficient indicates that the family \( k \) contains parts which are highly dissimilar to each other.

\[
\text{DC}_k = \frac{1}{\sum_{i=1}^{N-1} \sum_{j=i+1}^{N} d_{ij} \cdot X_{ik} \cdot X_{jk}}
\]

\[
\text{DC}_k = \frac{1}{\sum_{i=1}^{N-1} \sum_{j=i+1}^{N} s_{ij} \cdot X_{ik} \cdot X_{jk}}
\]

(2)

ii) Dissimilarity coefficient for the configuration:

The average of the dissimilarity coefficient of all the part pairs in the configuration can be expressed as:

\[
\text{CDC} = \frac{1}{\sum_{k=1}^{K} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} d_{ij} \cdot X_{ik} \cdot X_{jk}}
\]

\[
\text{CDC} = \frac{1}{\sum_{k=1}^{K} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} s_{ij} \cdot X_{ik} \cdot X_{jk}}
\]

(3)

This coefficient is taken as a measure of the overall dissimilarity between the parts in different families in a particular grouping. A high value may indicate the possibility of decreasing the dissimilarity by reallocating some parts from the present configuration. This idea is used in the approximation procedure for allocating the parts between the families to get a good initial solution.

4.1.4 Formulation

The problem could be formulated as follows:
Minimize the overall dissimilarity coefficient:

Minimize $Z_1 = CDC$ \hfill (4)

Subject to the following constraints:

1) Each part is allocated to only one family:

$$\sum_{k=1}^{K} x_{ik} = 1 \text{ for } i=1,2,3,...,N \hfill (5)$$

2) Each part family should at least have some specified number of parts say, $L$. This constraint may or may not be specified.

$$\sum_{i=1}^{N} x_{ik} \geq L \hfill (6)$$

for $k=1,2,...,K$

3) $x_{ik} = 0 \text{ or } 1 \text{ for } i=1,2,...,N \text{ and } k=1,2,...,K$

Let constraints (i), (ii) and (iii) be denoted by $C_1$.

The objective function in (4) is a ratio of two non-linear functions. As a first step in solving the problem, the numerator and denominator of the objective functions are linearized. The linearization scheme [14] is explained next.

Consider the term $x_{ik} \cdot x_{jk}$; both $x_{ik}$ and $x_{jk}$ are 0-1 integer variables.

Each of the terms $x_{ik} \cdot x_{jk}$ can be replaced by $m_{ijk}$ with the addition of the following constraints:

3(i) $x_{ik} + x_{jk} - m_{ijk} \leq 1 \hfill (7)$

3(iv) $m_{ijk} \leq x_{ik} \hfill (8)$
v) \( M_{ijk} \leq X_{jk} \)  \hspace{1cm} (9)

The above constraints force the variable \( M_{ijk} \) to assume the values 0-1. Let the set of constraints (iii), (iv) and (v) for all \( i, j \) and \( k \) be denoted by \( C_2 \).

With the linearized numerator and denominator, the formulation can be written as follows:

\[
\begin{align*}
\text{Minimize } Z_2 &= \frac{\sum_{k=1}^{K} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} d_{ij} \cdot M_{ijk}}{\sum_{k=1}^{K} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} s_{ij} \cdot M_{ijk}} \\
\text{Subject to: } C_1 \text{ and } C_2
\end{align*}
\]

Let \( C \) denote the constraint sets \( C_1 \) and \( C_2 \).

4.2 Solution Procedure

The objective function in (10) is a ratio of two linear integer functions \( A(X) \) and \( B(X) \). This type of problem is referred to as fractional programming in the literature. Methods have been reported for solving the fractional programming models with continuous decision variables [11, 32, 34]. The objective function in (10) being defined on zero-one integer variables, does not lend itself to these methods. Hence, in this case, a general search principle [9] has been adopted which involves solving a series of linear/non-linear problems to arrive at the optimal solution.
to a fractional programming problem. A description of this principle is given in section 4.2.2.

Section 4.2.2 also describes a method developed for this model to restrict the search area and subsequently to reduce the search time. This method identifies a range of values in which the optimal value for the ratio \( A(X)/B(X) \) exists.

### 4.2.1 Parametric Search Principle

The objective function (10) can be expressed as:

\[
P(\cdot): \min_{X \in C} \frac{A(X)}{B(X)}
\]

Let \((X^*)\) be the optimal solution to \(P(\cdot)\).

Then,

\[
\min_{X \in C} \frac{A(X)}{B(X)} = \frac{A(X^*)}{B(X^*)} = R^*
\]

and

\[
A(X^*) - R^* \cdot B(X^*) = 0
\]

Consider the following problem:

\[
P(R): \min_{X \in C} Z_3 = \min_{X \in C} [A(X) - R \cdot B(X)] = \min_{X \in C} Z(R,X)
\]

The function \(Z(R,X)\) for a particular \((X)\) decreases with increasing values of \(R\), since, both the functions \(A(X)\) and \(B(X)\) have only positive coefficients (\(d_{ij}\) and \(s_{ij}\) respectively) and are defined over the same set of
non-negative variables \((M_{ijk})\). It follows that the optimal value of \(Z(R,X)\) will also behave in a similar manner with respect to changes in \(R\). This characteristic of \(Z(R,X)\) helps in deciding the direction of search for the optimal ratio \(A(X)/B(X)\). The value of the parameter \(R\) which gives a value of \(Z(R) = 0\) is the optimal ratio \(A(X)/B(X)\). This will be clear from the following:

a) **Suppose \(R = R_0\) and Optimal \((X) = (X_0)\)**

Then, \(Z(R_0,X_0) = A(X_0) - R_0 \cdot B(X_0) = 0 \) (say)

Since \(\min_{X \in C} A(X) - R \cdot B(X) = A(X_0) - R_0 \cdot B(X_0) = 0\)

\[ \Rightarrow A(X_0)/B(X_0) = R_0 \]

Now, \(A(X) - R_0 \cdot B(X) \geq 0\) for all \(X \in C\)

\[ \Rightarrow A(X)/B(X) \geq R_0 \]

\[ \Rightarrow R^* = R_0 = A(X^*)/B(X^*) \] (13)

b) **Suppose \(R = R_1\) and Optimal \((X) = (X_1)\)**

Then, \(Z(R_1,X_1) = A(X_1) - R_1 \cdot B(X_1) > 0 \) (say)

Since \(\min_{X \in C} A(X) - R \cdot B(X) = A(X_1) - R_1 \cdot B(X_1) > 0\),

\[ A(X) - R_1 \cdot B(X) > 0 \] for all \(X \in C\)

\[ \Rightarrow A(X)/B(X) > R_1 \]

\[ \Rightarrow R^* = A(X^*)/B(X^*) > R_1 \] (14)

c) **Suppose \(R = R_2\) and Optimal \((X) = (X_2)\)**

Then, \(Z(R_2,X_2) = A(X_2) - R_1 \cdot B(X_2) < 0 \) (say)

\[ \Rightarrow A(X_2)/B(X_2) < R_2 \]
Since \( \frac{A(X_2)}{B(X_2)} < R_2 \),

The optimal solution,

\[
\min_{X \in C} A(X)/B(X) = A(X^*)/B(X^*) = R^* < R_2
\]

i.e.

\[
\frac{A(X^*)}{B(X^*)} < \frac{A(X_2)}{B(X_2)} < R_2
\]

Hence from (b) and (c),

\[
R_1 < R^* < R_2
\]

(15)

(16)

Now, consider \( R_3 = \frac{R_1 + R_2}{2} \)

\[
R_1 \quad R_3 \quad R_4 \quad R_2
\]

If \( Z(R_3) = \min_{X \in C} A(X) - R_3B(X) > 0 \)

then, \( R_3 < R^* < R_2 \)

(from similar arguments in (b) and (c))

Now consider \( R_{14} = \frac{R_2 + R_3}{2} \) and continue the search.

If \( Z(R_3) = \min_{X \in C} A(X) - R_3B(X) < 0 \)

then, \( R_1 < R^* < R_3 \)

(from similar argument in (b) and (c))

Now consider \( R_{14} = \frac{R_1 + R_3}{2} \) and continue the search.

The solution for problem \( P(\cdot) \) is obtained from a binary search for the parameter \( R \) which gives \( Z(R) = 0 \).

In other words, the search for \( R^* \) can be carried out by solving a series of problems \( P(R) \) with different values of \( R \), each time selecting the value of \( R \) depending on the optimal solution of the previous problem.
4.2.2 Finding an Interval \((L, U)\) Such that \(L < R^* < U\)

By initially choosing a value of \(R\) too far away from \(R^*\), a considerable amount of computation will be required to converge on \(R^*\). Hence it is necessary to identify a range of \(R\) in which \(R^*\) lies. This can be done by finding the upper bound and lower bound for the function \(Z(R, X)\) at different values of \(R\) with some constraints relaxed. If for a particular \(R\), both these bounds are positive, the problem \(P(R)\) need not be solved, since it is known beforehand that the optimal solution to \(P(R)\) cannot be zero. A similar argument holds for the case when both the upper bound and the lower bound are negative.

Consider \(P(R) : \min_{X \in C} Z(R, X) = Z(R)\)

and \(\max_{X \in C} Z(R, X) = \bar{Z}(R)\)

Let \(C_1\) be any subset of set \(C\) (Constraint set \(C\) has been defined earlier). Consider the minimization and the maximization of \(Z(R, X)\) under \(C_1\) (i.e., fewer number of constraints).

Let \(\min_{X \in C_1} Z(R, X) = L_{BR}\), the lower bound.

\(\max_{X \in C_1} Z(R, X) = U_{BR}\), the upperbound.

Now, for all \(R\),

\(L_{BR} \leq Z(R) \leq U_{BR}\) \hspace{1cm} (17)

\(L_{BR} \leq \bar{Z}(R) \leq U_{BR}\) \hspace{1cm} (18)
It is evident that only those values of $R$ which give a negative $\text{LB}_R$ and a positive $\text{UB}_R$ have to be considered in the search for $R^*$. The changes in values of $\text{LB}_R$ and $\text{UB}_R$ with respect to the changes in $R$ are indicated in Table 1. $R^*$ lies in the region $(L,U)$. In this region the binary search principle outlined in Section 4.2.1 can be applied with $R$ as the parameter.

Another point to be noted here is that, although a strict binary search plan requires the whole region $(L,U)$ to be searched, actually it is possible to restrict to the lower end of the region $(L,U)$. The basic strategy of the search is to solve the problems with different values of $R$, looking for a value of $R$ that gives the value of $Z(R)$ equal to zero. Since $\text{LB}_R$ is a relaxed solution to the minimization of $Z(R,X)$, it can be expected that this will occur (i.e., $Z(R) = 0$) at those values of $R$ giving $\text{LB}_R$ values closer to zero on the negative side.

Establishing the interval $(L,U)$ will reduce guesswork and the computational requirements of the search.

4.2.3 Establishing Lower Bound($\text{LB}_R$) And Upper Bound($\text{UB}_R$) for $Z(R,X)$

4.2.3.1 Constraints on the function $Z(R,X)$

Consider $P(R) : \min_{X \in \mathcal{C}} A(X) - R B(X)$
TABLE 1

Illustration of the Region \((L, U)\) for \(P(R)\)

<table>
<thead>
<tr>
<th>Value of (R)</th>
<th>(\text{Sign of } LBR)</th>
<th>(\text{Sign of } UBR)</th>
<th>Case</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>+</td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>(0+s)</td>
<td>+</td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>(0+2s)</td>
<td>+</td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>(0+k_1s)</td>
<td>-</td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>(0+(k_1+1)s)</td>
<td>-</td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>(0+k_2s)</td>
<td>-</td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>(0+(k_2+1)s)</td>
<td>-</td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>(0+k_3s)</td>
<td>-</td>
<td>-</td>
<td></td>
</tr>
</tbody>
</table>

\(s\) is a small increment.
\[
\text{Min } Z(R,X) \\
\text{ } \\
_k N-1 \sum \sum d_{ij} M_{ijk} \text{ } R. [ \text{ } _k N-1 \sum \sum s_{ij} M_{ijk} ] \\
_X \& C \text{ } \\
\text{ } \\
= \text{ } \sum \sum \sum (d_{ij} - R.s_{ij}) M_{ijk} \\
_X \& C \text{ } \text{ (19)} \\
\]

Let \((d_{ij} - R.s_{ij}) = C_{ij}\). In the discussions to follow the superscript \(R\) has been dropped for convenience.

The following conditions are implied by the constraint set \(C\).

(i) \(M_{ijk}\)'s are zero-one variables (forced to assume values of \(0/1\)).

(ii) Let \(S_{ij} = [M_{ij1}, M_{ij2}, M_{ij3}, \ldots M_{ijk}]\).
     Since each of the parts \(i\) and \(j\) can be allotted to only one family, at most one variable in the set \(S_{ij}\) can assume a value \(1\).
     This can be illustrated by Table 2.

(iii) At least IP number of variables \(M_{ijk}\) should have a value \(1\) where IP is given by the following expression:
     \[\text{IP} = [N/K].([N/K]-1)/2 \times K + (N - K.[N/K]) \times [N/K]\]
     where, \([N/K]\) is the largest integer less than or equal to \(N/K\).
This expression represents the minimum number of part pairs (which in turn corresponds to the minimum number of $M_{ijk}$ variables taking the value 1) that have to be formed while grouping $N$ parts into $K$ families. It is impossible to form $K$ families out of $N$ parts without forming at least IP part pairs. The expression for IP has been derived by trial and error.

This can be illustrated by the number of distinct possible groupings possible for different values of $N$ given in Table 3:

For $N=2$ and $K=3$ \( IP=0 \)
For $N=3$ and $K=3$ \( IP=0 \)
For $N=4$ and $K=3$ \( IP=1 \)

Since this is true for all $N$, it follows that at least a minimum number of $M_{ijk}$'s must take the value of 1 in any feasible solution to $P(R)$.

(iv) Some combinations of $M_{ijk}$'s cannot take the value of 1 in the same solution.

For example, consider the part pair $(1,2)$ has been in family 1.

Then $M_{121} = X_{11} \cdot X_{21} = 1 \cdot 1 = 1$

In this case, the variable $M_{132} = X_{32} \cdot X_{12} = 0$

Since, $X_{11} + X_{12} = 1 \& X_{11}=1 \implies X_{12}=0$

i.e., $M_{121}$ and $M_{132}$ cannot take a value 1 at the same time.
TABLE 2

Part pair (1,2)

K, the number of families = 2

<table>
<thead>
<tr>
<th>No.</th>
<th>Allocation</th>
<th>Value of</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>F1 F2</td>
<td>M_{121} M_{122}</td>
</tr>
<tr>
<td>1</td>
<td>1 2</td>
<td>0 0</td>
</tr>
<tr>
<td>2</td>
<td>2 1</td>
<td>0 0</td>
</tr>
<tr>
<td>3</td>
<td>1,2 1,2</td>
<td>1 0</td>
</tr>
<tr>
<td>4</td>
<td>1,2</td>
<td>0 1</td>
</tr>
</tbody>
</table>

Note: M_{121} = X_{11} \cdot X_{21} and
M_{122} = X_{12} \cdot X_{22}

It is clear that at most one value in the set S_{12} takes a value of 1

TABLE 3

IP, The Minimum Number Of Part Pairs

K, the number of families = 3

<table>
<thead>
<tr>
<th>N</th>
<th>Distinct groupings</th>
<th>F1</th>
<th>F2</th>
<th>F3</th>
<th>No of part pairs</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td></td>
<td>0 *</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1,2</td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>61</td>
<td>2</td>
<td>3</td>
<td>0 *</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1</td>
<td>2,3</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>1,2,3</td>
<td>-</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3,4</td>
<td>1 *</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1,2</td>
<td>3,4</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>1</td>
<td>2,3,4</td>
<td>-</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>1,2,3,4</td>
<td>-</td>
<td></td>
<td>3</td>
</tr>
</tbody>
</table>
If any one of these conditions are violated by a variable $M_{ijk}$, some of the constraints in the set $C$ will be violated. The constraint set $C$ minus the violated constraints is denoted as $C^1$.

The procedure explained in the next section neglects condition (iv) and finds the maximum ($UB_R$) and the minimum ($LB_R$) of the function $P(R)$ under a reduced set of constraints $C^1$. As shown in Section 4.2.2, $UB_R$ and $LB_R$ will be the bounds on the objective function $P(R)$ for maximization and minimization respectively subject to the constraint set $C$.

4.2.3.2 **Coefficients of the function $Z(R,X)$**

The $M_{ijk}$'s are 0-1 variables. Hence the objective function $Z(R,X)$, is the sum of all $C_{ij}$'s corresponding to the $M_{ijk}$'s taking a value of 1. From conditions (i), (ii), (iii) and (iv) it follows that:

1) If $C_{ij}$'s are positive for all $(i,j)$, then both the upper bound and lower bound would be positive (Case I).

2) If $C_{ij}$'s are negative for all $(i,j)$, then both the upper bound and lower bound would be negative (Case II).

3) If some $C_{ij}$'s are positive and some $C_{ij}$'s are negative, then the value of $R$ may correspond to
Case III. It is necessary to find the \( L_R \) and \( U_R \) in this situation.

4.2.3.3 **Algorithm for finding \( L_R \) and \( U_R \)**

Although each \( C_{ij} \) is a coefficient for \( K \) linearization variables, from condition (ii) in Section 4.2.3.1 it can be counted in only once for any solution.

The lower limit (upper limit) of \( Z(R,X) \) can simply be found by counting in all the negative (positive) \( C_{ij} \)'s. However if the condition (iii) is not satisfied some positive (negative) terms should be counted in.

**Algorithm**

1. Sort the set \( J \) in ascending order.

2.0 \( \text{Sum}_L = 0 \)

2.1 Add all the negative \( C_{ij} \)'s to \( \text{Sum}_L \)

2.2 If \( NN > IP \) go to step 3.0

2.3 If \( NN < IP \) then add to \( \text{Sum}_L \) the first \( (IP-NN) \) positive terms in set \( J \)

3.0 \( \text{Sum}_U = 0 \)

3.1 Add all the positive \( C_{ij} \)'s to \( \text{Sum}_U \)

3.2 If \( NP > IP \) go to step 4.0

3.3 If \( NN < IP \) then add to \( \text{Sum}_U \) the last \( (IP-NN) \) negative terms in set \( J \)
4.0 \( LBR = \text{Sum}_L \) and \( UBR = \text{Sum}_U \)

As indicated earlier, this algorithm neglects condition (iv) implied by the constraint set C.

4.2.4 Summary of the steps for solving the formulation

In brief, the steps in solving problem P(.) are:
1. Set up the objective function P(R).
2. For different values of R starting with 0 find \( LBR \) and \( UBR \).
3. Establish the interval (L,U) by the method explained in Section 4.2.2.
4. Carry out the binary search for the value of \( R^* \) in this interval (L,U).
   Or
   Choose a smaller interval \((R_1, R_2)\) in the lower end of the range (L,U) and carry out the search for \( R^* \) (since \( R^* \) is expected to be at the lower end of the region (L,U)).
5. Stop when a value of R yields \( Z(R, X) = 0 \)

4.3 Approximation Procedure

4.3.1 Need for Finding a "Good" Initial Solution

The grouping problem is combinatorial in nature.
Each of the $N$ parts can be allocated to one of the $K$ families independently. Hence the number of feasible solutions to the family formation problem is $K^N$, which becomes too large with increasing values of $N$. The solution time required for the problem $P(R)$ will also increase rapidly.

It is clear that the time required for the search as outlined in section 4.2.1 is dependent on the number of problems $P(R)$ solved in the process. The number of problems $P(R)$ required to be solved depends on the interval $(R_1, R_2)$ chosen initially.

If the time required for each problem $P(R)$ is high, it is desirable to limit the number of such problems solved to as few as possible. This means that a tight interval $(R_1, R_2)$ has to be selected.

The approach suggested in this case is to find a "good" solution and choose the corresponding CDC as the upper limit on the value of $R$. Any procedure for finding such an initial solution should be expected to satisfy the following requirements:

a. The solution should be "good". An initial solution is considered to be better than others if the corresponding CDC is nearer to the value of $L$ ($L$ is the lower limit on the value of $R$). This results in a shorter search interval.

b. The time required to arrive at that solution
should be justifiable.

An approximation procedure has been developed for this purpose. Both the above requirements have been found to be satisfied by the procedure in the several problems solved.

4.3.2 Principle

The approximation procedure also uses the IP formulation described in section 4.1.4. In this case however, a number of smaller problems are solved instead of a single large problem. The procedure is based on a method of clustering first reported by Friedman and Rubin [13]. Whereas the principle in [13] is single reallocation based, the procedure developed in this section is multiple reallocation based.

A random partitioning of N parts into K families is considered initially to start off the approximation procedure. Let $n_1, n_2, \ldots, n_K$ be the number of parts in families 1, 2, ..., K, respectively.

4.3.2.1 Single Reallocation

The principle as applied to part grouping problem is given below:

Start with a random partitioning of parts into K
families. The parts are considered in a particular order for moving into other families. The part selected is moved to some other family such that it brings about the maximum favourable change in the objective function. This reallocation generates a new configuration, and causes the coefficients to assume new values. The procedure restarts each time a reallocation move is made. If the reallocation fails to bring about a favourable change, the part is retained in its present family and the next part in the order is selected. This continues until no part can be moved from its present family to another.

This approach to part grouping has been applied by Dutta et al. [12], for the part family formation. Different trials were conducted with varying starting partitions. The final objective function values were very close to each other irrespective of the starting configurations.

4.3.2.2 Multiple Reallocation

It can be noted that in the single move algorithm, each time a part is considered for reallocation, a decision is taken with respect to each family about moving the part to that family. The value of the objective function is calculated for all the possible reallocations. This, in effect, means that a problem with $1 \times K$ integer variables (0-1) is solved each time by complete enumeration.
Extending the same principle, all the parts of a particular family "s" (to be chosen based on some criterion) can be considered for reallocation.

If there are $n_s$ parts in family "s", the number of feasible reallocations is $n_s \times K$, which is quite large even for small values of $n_s$, if a complete enumeration has to be attempted. However, the reallocation of these $n_s$ parts can be considered using the formulation in section 4.1.4. The allocation of all the other parts in other families is fixed and the reallocation of the $n_s$ parts into $K$ families is considered.

A series of smaller sub-problems are solved, until a stopping criterion is reached. The criterion for choosing the family to be considered for reallocation is the value of $D_k$ for different part families ($k=1,2,\ldots,K$).

We define for a family $k$,

$$D_k = \frac{\sum_{i=1}^{N-1} \sum_{j=i+1}^{N} d_{ij} x_{ik} - x_{jk}}{B(X)} \quad (20)$$

A high value of $D_k$ indicates that the parts in family $k$ are such that the contribution from the part family $k$ to the value of CDC is very high, which suggests the presence of highly dissimilar parts in that family. This means that the parts from this family are the candidates for reallocation. The family with highest $D_k$ (family, MF) is considered for reallocation of parts at an iteration of the algorithm.
As indicated, a sub-problem of the form \( P(\cdot) \) with \( M \times K \) integer variables is solved in an iteration. After each successful iteration (iteration causing an improvement in the objective function), the algorithm returns to a stage similar to the initial configuration with an improved bound on the value of the objective function. The algorithm terminates when the reallocation of parts from any family fails to bring about an improvement in the objective function.

4.3.3 Algorithm

The flowchart of the algorithm is shown in Fig. 1. A brief explanation of the flowchart blocks follows:

- Block (a) initialize the algorithm by computing the values of the objective function \( Z_1 \) and \( D_k \) for all the families.
- An iteration of the procedure involves reallocation consideration of all the parts in a family for improving the value of the objective function \( Z_1 \).
- Blocks (b), (c) and (d) represent the main steps in an iteration. The reallocation subproblem is solved as an IP of the same form as formulated in section 4.1.4.
Decision block (e) indicates whether a reallocation of the parts has to be made or not.

The loop (e)-(f)-(g)-(h)-(a) represents the steps involved when a decision to reallocate the parts from family "MF" is made.

When it is impossible to reallocate the parts in family "MF" under consideration, the loop (e)-(i)-(j)-(b) represents the choice of another family for considering reallocation of parts.

Block (i) identifies the families from which the parts could not be reallocated within an iteration.

The procedure terminates when the decision block (j) returns a result YES.
Start

Read initial configuration

Compute $D_k$ for all $k$
Compute $z_k$ for the configuration

Among families under consideration, identify family 'MF' having the max $D_k$

Consider all the parts in the family 'MF' for reallocation, keeping the parts in other families in their present allocation. Formulate this as a sub-problem of allocating $n_{MF}$ parts in the family 'MF'.

Solve the problem and find $z_k(NEW)$

If $z_k(NEW) < z_k$

Yes

Reallocate the parts from family 'MF' as per IP-subproblem solution
The current configuration is considered as initial configuration
Remove all flags from the families

The current family configuration is optimal
Stop

No

Flag family 'MF', i.e., remove family 'MF' from further consideration

Are all families considered?

Yes

No
Chapter V

MACHINE GROUP ALLOCATION

The assignment of machine groups for the production of the parts segregated into part families is discussed in this chapter. The availability of alternative machines for each of the operations on the parts is considered. Our objective is to maximize the number of available alternative routings for the parts within the cellular system. The formulation allows for the individual machines to be allocated to only one part family. When such allocation is infeasible, the machine(s) causing this infeasibility is(are) identified through a mathematical model. The condition of allocation to only one family is relaxed for these machines.

5.1 Formulation

5.1.1 Statement of the Problem

All the parts of a particular family have to be processed completely within the corresponding machine

- 55 -
group. These machine groups constitute the FMCs or cells. The aim of this problem is to allocate a group of machines to each of the part families.

5.1.2 Objective

The objective is to provide the maximum possible number of alternative routings for the parts within their respective machine groups. The availability of alternative routings is known as the routing flexibility. Routing flexibility is maximized taking into account the operation requirement of the parts in different families. The routing for a part is defined as a sequence of machine visits needed to complete the operations required. Consider a part with three operations. A sequence of visits to the machines 3, 2 and 5 represents a routing for that part.

5.1.3 Concept of Alternative Routings

Notations

\[ N \quad \text{Total number of parts} \]
\[ K \quad \text{Number of part families} \]
\[ n_k \quad \text{Number of machine groups to be formed} \]
\[ n_k \quad \text{Number of parts in family } k. \]

Therefore, \[ \sum_{k=1}^{K} n_k = N \]
jk = Index for part j in family k (j=1,2,3...nk)
O(jk) = Number of operations on a part indexed by
       the part identity jk
M = Number of available machines
PR_{jk} = Maximum possible number of routings for
          part jk
FP_{jk} = Number of alternative machines
          available for operation p of part jk.
NR_{jk} = Number of alternative routings available
          for part jk
S_{k} = Product terms of the decision variables
       (to be defined later) to indicate the
       allocation of individual machines to k
       families (k=2,3...K). P_{i,k} indicates the
       product term i in set S_{k}.
W_{k} = Penalty weight to k family allocation of a
       machine in the model to identify the
       machines causing infeasibility in the
       machine group allocation formulation.

The feasible routing for the operations is
represented in the form a matrix as explained below:

Consider a part jk with O(jk) = 3. Assume M = 9.
The matrix A_{jk}, with elements a(jk)_{pm} indicates
the feasibility of operation p of part jk on
different available machines.
Op M  1  2  3  4  5  6  7  8  9
1  1  0  1  0  1  0  0  0  1  4 feasible machines
2  0  1  0  0  0  0  1  0  1  3 feasible machines
3  0  0  0  1  1  1  0  1  0  4 feasible machines

For example, operation 1 can be done on machines 1, 3, 5, or 9.

This information is assumed to be available based on the technological capabilities of the available M machines, which can be expressed by a matrix for each part jk.

**Decision variables**

\[ I_{mk} = \begin{cases} 
1 & \text{if machine } m \text{ is allocated to cell } k, \\
0 & \text{otherwise.}
\end{cases} \]

Referring to the previous example, if all the machines 1, 2, ...., 9 are assigned to family k, all the possible alternative routings, \( PR_{jk} \) will be available for the part jk.

\[
PR_{jk} = \begin{bmatrix} \sum_{m=1}^{M} a(jk)1m \cdot \sum_{m=1}^{M} a(jk)2m \cdot \sum_{m=1}^{M} a(jk)3m \end{bmatrix}
\]

\[
= \begin{bmatrix} 4 \cdot 3 \cdot 4 \end{bmatrix} = 48
\]

This may be the most desirable situation as far as
the production of part \( jk \) is concerned. But, all the
machines cannot be allocated to family \( k \), since there will
be a requirement of these machines for parts in other
families. Due to these requirements, allocation of the
machine groups has to be done for each of the families with
the objective of maximizing the number of routings
available.

5.1.4 **Formulation**

Consider the decision variable \( I_{mk} \) which represents
the allocation of machine \( m \) to family \( k \). Each machine can
be allocated to only one family.

If a machine \( m \) is allocated to family \( k \), it offers a
routing possibility for all the operations in that family
that can be done on machine \( m \).

A formulation with the objective function to maximize
the number of alternative routings available will try to
allocate those machines to a particular family, which offer
routing to a large number of operations.

For operation \( p \) on part \( jk \) the number of alternative
machines available is given by:

\[
FP_{jk} = \sum_{m=1}^{M} a(jk)pm \cdot I_{mk} \quad (22)
\]
The number of available routings for part $jk$ is then given by:

$$NR_{jk} = F^1_{jk} \cdot F^2_{jk} \cdot F^3_{jk} \cdot F^4_{jk} \cdots F^0_{jk}$$

(23)

The total number of alternative routings over all the parts in all the families is given by:

$$Z = \sum_{k=1}^{K} \sum_{j=1}^{n_k} [F^1_{jk} \cdot F^2_{jk} \cdot F^3_{jk} \cdot F^4_{jk} \cdots F^0_{jk}]$$

(24)

The objective function can be stated as:

$$P_{1} \quad \text{Maximize} \quad Z$$

(25)

Subject to the following constraints:

1. Each operation $p$ on the part $jk$ must have at least one feasible machine in the corresponding group. This ensures that for operation $p$ at least one routing is provided in the corresponding group of machines,

$$\sum_{m=1}^{M} a(jk)_{pm} \cdot I_{mk} \geq 1 \quad \text{for} \quad k=1,2,3,\ldots,K.$$  

(26)

$$j=1,2,3,\ldots,n_k \quad p=1,2,3,\ldots,O_{jk}$$

2. Each of the available machines can be allotted to one cell only.

$$\sum_{k=1}^{K} I_{mk} = 1 \quad \text{for} \quad m=1,2,3,\ldots,M$$

(27)
3. Integrality constraints:

\[ \text{Imk} = 0 \text{ or } 1 \]  \hspace{1cm} (28)

for \( m = 1, 2, 3 \ldots \ldots M \)

\[ k = 1, 2, 3 \ldots \ldots K \]

**Example of the expression for NR_{jk}**

The objective function of this formulation is the sum of alternative routings available for all the parts.

The expression for the number of alternative routings available for part \( jk \) is developed as follows:

\[ F_{jk} = \sum_{m=1}^{M} a(\text{jk})p_m \cdot \text{Imk} \]

\[ = 1 \cdot I_{1k} + 0 \cdot I_{21k} + 1 \cdot I_{3k} + 0 \cdot I_{4k} + 1 \cdot I_{5k} \]

\[ + 0 \cdot I_{6k} + 0 \cdot I_{7k} + 0 \cdot I_{8k} + 1 \cdot I_{9k} \]

\[ = I_{1k} + I_{3k} + I_{5k} + I_{9k} \]

Similarly,

\[ F_{2jk} = I_{1k} + I_{7k} + I_{9k} \]

\[ F_{3jk} = I_{1k} + I_{5k} + I_{6k} + I_{8k} \]

\[ \text{NR}_{jk}, \text{ the number of alternative routings available for the part } jk = \]

\[ (I_{1k}+I_{3k}+I_{5k}+I_{9k}) \cdot (I_{2k}+I_{7k}+I_{9k}) \cdot (I_{4k}+I_{5k}+I_{6k}+I_{8k}) \]
5.2 Solution Procedure

The objective function \( Z \) is non-linear in integer variables. The non-linear terms, each of which represents the possibility of a routing are the product terms of the decision variables \( I_{mk} \).

These terms are linearized by introducing additional variables using a scheme suggested in [12,13]. This is similar to the method adopted in the part family formation problem, where product terms of two integer variables were considered. In this case however, each term corresponding to a routing for a part \( jk \) will be a multiplication of \( \delta(jk) \) integer variables.

5.2.1 Linearization of Product Terms

The scheme for linearizing the product terms of zero-one variables is [12,13]:

Let \( Q \) be the index set of the variables in a particular product term.

- Replace each of the product terms of the type \( (x_j)^k \) by \( x_j^k \).
- Replace each of the product terms of the type \( \prod_{j \in Q} x_j \) by \( x_Q \) and add the constraints.
\[ \sum_{j} x_{j} - x_{Q} \leq q - 1 \]

and,

\[ x_{Q} \leq x_{j} \]

where \( q \) is the number of elements in \( Q \).

The linearization strategy adopted for the problem of part family formation is a specific case of this with \( q = 2 \).

This formulation is straightforward once the product terms are linearized.

5.2.2 Some Reductions in the Number of Product Terms

The number of variables in the formulation is problem specific, depending on the number of operations for the parts, number of possible machines for each operation and the number of machines. It was mentioned that each routing possibility for a part is denoted by a product term of \( 0(jk) \) decision variables. However, a careful consideration while generating the problem can result in a reduction of the actual number of terms.

For example, a routing for a particular part in the family \( k \) may be through machines 1-2-3. This routing will be identified by the product term \( I_{1k} \cdot I_{2k} \cdot I_{3k} \). It can be noted that, this product term will also represent the routings 1-3-2 and 3-2-1 for any other operation for
the parts in that family. Taking care of these situations while generating the problem for input to an IP routine would be helpful.

5.3 **Infeasibility in Machine Group Allocation**

It is assumed in Section 5.2 that one or more allocations of the machine groups to the families exist, such that each machine is allocated to only one family.

The condition that one machine should be allocated to only one family may not be possible sometimes due to the problem data.

The reason for infeasibility is the absolute necessity of some machine(s) to be in more than one family. The infeasibility can be removed from the problem by relaxing the assignment constraint (27) on the machine(s). These machines are allowed to be allocated to more than one family.

5.3.1 **Multiple Family Allocation of Some Machine(s)**

As mentioned earlier, the requirement of some machines in more than one family causes the infeasibility in the machine group allocation problem, Pl. The possible cases are the requirement of some machine(s) in two, three, ..., or K families.
The problem P1 can be made feasible by relaxing the allocation constraints on the machine(s) as follows:

Allow for some machine(s) to be allocated to two families and check for the feasibility of problem P1. If the problem is not feasible, then allow for some machine(s) to be allocated to three families and check for the feasibility of problem P1, and so on.

The rationale of the above strategy is to allow for sharing of some machine(s) by the least possible number of part families to make problem P1 feasible.

An objective function is defined in the next Section that implements this strategy and identifies the machine(s) for which the assignment constraints have to be relaxed.

Consider the constraints of problem P1, with constraint (27) modified as follows:

\[ \sum_{k=1}^{K} I_{mk} \leq K \quad (27-a) \]

Let \( S_1 \) denote the decision variables \( I_{mk} \) (Number of decision variables = \( M \cdot K \)). The possible multiple allocation variables defined by the original decision variables are listed in Table 4.

5.3.2 Mathematical Model to Identify the Machines Causing Infeasibility

All the product terms in \( S_2, S_3, \ldots, S_K \) take the
## TABLE 4

**Product Terms Indicating Multiple Allocation Of Machines**

<table>
<thead>
<tr>
<th>Description of Allocation</th>
<th>Product Terms</th>
<th>Example of a product term</th>
<th>Number of product terms</th>
</tr>
</thead>
<tbody>
<tr>
<td>Two families</td>
<td>$s_2$</td>
<td>$I_{mk} \cdot I_{ml}$</td>
<td>$M \cdot \binom{K}{2}$</td>
</tr>
<tr>
<td>Three families</td>
<td>$s_3$</td>
<td>$I_{mk} \cdot I_{ml} \cdot I_{mj}$</td>
<td>$M \cdot \binom{K}{3}$</td>
</tr>
<tr>
<td>K families</td>
<td>$s_K$</td>
<td>$I_{mk} \cdot I_{ml} \cdot I_{mK}$</td>
<td>$M \cdot \binom{K}{K}$</td>
</tr>
</tbody>
</table>
values 0-1 depending on the value of the decision variables in each of these product terms. Consider the minimization of an objective function $Y$ involving the above product terms.

Based on the strategy explained earlier, the coefficients of the product terms in $Y$ should be such that:

- No term in $S_2$ takes a value of 1, if the constraints can be satisfied by having the decision variables $I_{mk}$ to assume the value of 1 without any multiple assignments.

- No term in $S_3$ takes a value of 1 if the constraints can be satisfied by having the terms in $S_1$ (without any multiple assignments) and $S_2$ to take a value of 1.

- No term in $S_k$ takes a value of 1, if the constraints can be satisfied by having the terms in $S_1$ (without any multiple assignments), $S_2, \ldots, S(k-1)$ to take a value of 1.

Consider a weightage of zero for the terms in $S_1$, indicating no penalty to the objective function value for any single family allocation of a machine.

The terms in $S_1, S_2, \ldots, S_k$ are given increasing values of penalty weightages. An example is given in Table 5, which satisfies the conditions listed above. Any non-negative value for $D$ will give the same solution to the
### TABLE 5

**Penalty Weights to Multiple Allocation Of Machines**

<table>
<thead>
<tr>
<th>Product Terms</th>
<th>Weightage Equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td>0</td>
</tr>
<tr>
<td>S2</td>
<td>( W_2 = 0 \cdot M \cdot K + D )</td>
</tr>
<tr>
<td>S3</td>
<td>( W_3 = 0 \cdot M \cdot K + W_2 \cdot M \cdot (K_{2}) + D )</td>
</tr>
<tr>
<td>S4</td>
<td>( W_4 = 0 \cdot M \cdot K + W_2 \cdot M \cdot (K_{2}) + W_3 \cdot M \cdot (K_{3}) + D )</td>
</tr>
</tbody>
</table>

...
minimization of $Y$ i.e., the formulation is independent of the value of $D$ chosen.

The mathematical model to identify the machines causing the infeasibility can now be written as follows:

$$\begin{align*}
\text{Minimize} & \quad Y = \sum_{k=1}^{K} \sum_{i,k \in S_k} W_k \cdot P_{i,k} \\
\text{Subject to} & \quad \text{constraints (26), (27-a) and (28).}
\end{align*}$$

5.4 **Summary of steps for solving the formulation**

**Allocation Problem**

In brief the steps to be followed are:

i) Solve the problem **INF** to identify the machines which have to be allocated to more than one family.

ii) Relax the assignment constraints in the formulation $P_1$ for the machine(s) identified in (i).

iii) Solve problem $P_1$ for the allocation of machines to maximize the number of routings available for the parts.
Chapter VI

APPLICATION OF THE FORMULATIONS

The application of the formulations of part family formation and machine group allocation is illustrated in this chapter. Section 6.1 gives a description of the problem data. Details about the software written for generating the input problem matrix for the integer programming routine of SAS/OR (Version 5) are provided in Section 6.1.2. Solution procedures for the two problems are discussed in detail in Sections 6.2 and 6.3. A discussion about the application of the formulations and the scope for future work has been included in Section 6.4.

6.1 The Problem Data

6.1.1 Parts and Machines

The problem data considered represent the typical part spectrum characteristics and the machine tool variety in the Flexible Manufacturing Systems.
6.1.1.1 Parts Spectrum

A set of fifteen parts suitable for manufacturing on CNC machines and hence the natural choice for manufacturing in an FMS are considered. Sketches of these parts are given in Appendix B.

Process details required for the part family formation have been written for these parts and are also provided in Appendix B. A summary of the process requirements is given in Table 6.

When a part visits a machine, a number of processing steps can be carried out and this set of processing steps constitutes an operation. Referring to the process details for Part 81 (HOUSING), a processing step, e.g., (1) which is a combination of nine processing steps, is an operation.

These parts have the characteristics explained in Section 2.2.2.1.

6.1.1.2 Machines

The machines assumed to be available for allocation to part families are basically the variety of machining centres found in FMSs. The two major types of machining centres are Horizontal Spindle and Vertical Spindle. Heavy boring operations are done on designated
### TABLE 6

**SUMMARY OF PROCESS REQUIREMENTS**

<table>
<thead>
<tr>
<th>PART</th>
<th>NUMBER OF TOOLS AND THE TOOL REQUIREMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(TOOL CODES ASSUMED)</td>
</tr>
<tr>
<td>2</td>
<td>M101 M103 M104 M105 M901 M602 M501 M506 M508 M402 D201 D125 R126 S104 B101 B109 B105 B104</td>
</tr>
<tr>
<td>6</td>
<td>M501 M504 M502 M415 M412 M416 M901 M902 M401 M507 M509 D201 D118 D130 B108 B112 B109 S102</td>
</tr>
<tr>
<td>7</td>
<td>M102 M401 M103 M105 M404 M511 M402 M512 M407 M408 M502 M701 M704 M702 M413 M506 M508 M601 D201 D108 D120 D115 D202 D105 S102 S101 T108 R120</td>
</tr>
<tr>
<td>11</td>
<td>M112 M401 M104 M412 M403 M402 M410 D201 D130 D202 D116 B101 B109 B105 T116</td>
</tr>
<tr>
<td>12</td>
<td>M112 M401 M104 M105 M101 M106 M402 M505 M710 M107 M702 M704 M502 D201 D130 D202 D125 B108 B109 R130 S102</td>
</tr>
</tbody>
</table>
machines, with sturdier structure. The light operations of drilling and tapping are done on special CNC drilling machines when necessary. In total, twelve machines of these different types are assumed to be available for allocation to the part families.

The physical dimensions of the problem under consideration can be summarized as follows:

15 parts

12 Machines

4 Horizontal Spindle Machining Centers
3 Vertical Spindle Machining Centers
3 Boring Centers (Heavy Machining)
2 NC Drilling and Tapping Machines

Three cells

6.1.2 Generation of Problem Input to An IP Routine

The problems are solved using the integer programming routine of the SAS/OR package (Version 5) on an IBM 4381 computer.

The input problem matrix has to be generated through a program for each of the problems, since the problem sizes are too large for manual input.

A series of program modules in Fortran have been written for the generation of the problem matrix in SAS/OR
format for different problems listed below.

a. $P(R)$, the parametric objective function problem in
   the part part family formation.

b. Finding upperbound and the lowerbound for the
   problem $P(R)$ for varying values of $R$ (to find the
   interval $(L,U)$) by the algorithm in Section
   4.2.3.3

c. Subproblems of type $P(R)$ to be solved in the
   iterations of Approximation Procedure.

d. Finding the upperbound and lowerbound for the
   subproblems $P(R)$ in the Approximation Procedure
   for varying values of $R$, to find the interval
   $(L,U)$.

e. Problem $P_1$, the machine group allocation
   formulation.

f. Problem $INF$, for identifying the machines causing
   infeasibility in the machine group allocation
   problem.

The program listings are given in Appendix C.

6.2 Part Family Formation - An Example

The solution procedure for the fractional programming
formulation of the problem involves a search procedure as
indicated in section 4.2. The problem size for the data in
Appendix B is as follows.
N=15 and K=3.

Problem size:

\# of integer $X_{i,k}$ variables: 30
\# of continuous $X_{i,k}$ variables: 15
\# of $M_{i,j,k}$ variables: 315

TOTAL \# OF VARIABLES: 360

\# of type (i) constraints: 15
\# of type (ii) constraints: NIL
\# of type (iii) constraints: 315
\# of type (iv) and (v) constraints: 630

TOTAL \# OF CONSTRAINTS: 960

The computations involved in solving the problem are indicated Sections to follow.

6.2.1 Finding the interval $(L, U)$

The values of LB$_R$ and UB$_R$ were calculated using the algorithm in section 4.2.3.3 for the values of R from 0.05 to 10.50 in steps of 0.05. The partial listing of the values is tabulated in Table 7.

From the table:

$L = 2.45$ ; $LB_{2.45} = -5.10$ ; $UB_{2.45} = 1041.59$

$U = 6.10$ ; $LB_{6.10} = -2272.2$ ; $UB_{6.10} = 0.20$

Based on the proof in section 4.2.2 we have,

$2.45 \leq R^* \leq 6.10$

The argument about restricting to the lower end of the
### TABLE 7

**BOUNDS ON THE OBJECTIVE FUNCTION FOR DIFFERENT VALUES OF R**

**TOTAL NUMBER OF CIJ's = 315**

<table>
<thead>
<tr>
<th>R</th>
<th># OF NEGATIVE CIJ's</th>
<th>UPPER BOUND</th>
<th>LOWER BOUND</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.05</td>
<td>0</td>
<td>734.45</td>
<td>3076.54</td>
</tr>
<tr>
<td>0.10</td>
<td>0</td>
<td>720.90</td>
<td>3033.09</td>
</tr>
<tr>
<td>0.15</td>
<td>0</td>
<td>707.35</td>
<td>2989.64</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.45</td>
<td>27</td>
<td>-5.10</td>
<td>1041.59</td>
</tr>
<tr>
<td>2.50</td>
<td>33</td>
<td>117.50</td>
<td>1005.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6.10</td>
<td>282</td>
<td>-2272.29</td>
<td>0.20</td>
</tr>
<tr>
<td>6.15</td>
<td>282</td>
<td>-2313.44</td>
<td>-7.70</td>
</tr>
<tr>
<td>6.20</td>
<td>282</td>
<td>-2354.59</td>
<td>-26.80</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9.95</td>
<td>309</td>
<td>-5536.75</td>
<td>-605.15</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
region \((L, U)\) in the search for \(R^*\) is evident from the values of \(UB_R\) and \(UB_R\) in Table 7.

6.2.2 Initial Solution through Approximation Procedure

The total solution time requirement for solving the part family formation problem is dictated by the number of problems \(P(R)\) solved during the search. Each of the problems \(P(R)\) to be solved in this case is of the same size indicated earlier.

Considering the problem size and the solution time required, the importance of starting with an initial solution nearer to the optimal solution is evident.

The approximation procedure as outlined in the section is applied in this case to find an initial solution. The result from a single trial of the approximation procedure with some random starting configuration solution is sufficient to get an initial solution.

As indicated earlier, the requirements of such an approximation procedure are arriving at a "good" solution (near to optimal) and doing so in a reasonable amount of time (time comparable to the solution time of one problem \(P(R)\)). With a view to test the procedure, trials are carried out with different starting configurations. Table 8 provides a summary of these trials. The typical
### TABLE 8

**Summary of Trials with Different Starting Configurations for Fifteen Parts Example.**

<table>
<thead>
<tr>
<th>#</th>
<th>Starting Config.</th>
<th>Initial ODC</th>
<th># of Iter</th>
<th>Final Config.</th>
<th>Final ODC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>[1,6,7,12,13]</td>
<td>3.6653</td>
<td>7</td>
<td>[1,3,4,8,15,13]</td>
<td>2.8158</td>
</tr>
<tr>
<td></td>
<td>[2,5,8,11,14]</td>
<td></td>
<td></td>
<td>[2,6,11,12]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>[3,4,9,10,15]</td>
<td></td>
<td></td>
<td>[5,7,9,10,14]</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>[1,6,7,12,13]</td>
<td>3.6971</td>
<td>8</td>
<td>[1,3,4,8,15,13]</td>
<td>2.8158</td>
</tr>
<tr>
<td></td>
<td>[2,5,8,11,14]</td>
<td></td>
<td></td>
<td>[2,6,11,12]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>[3,4,9,10,15]</td>
<td></td>
<td></td>
<td>[5,7,9,10,14]</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>[1,5,7,9,13,14]</td>
<td>3.1573</td>
<td>8</td>
<td>[1,3,4,5,6,13]</td>
<td>2.79999</td>
</tr>
<tr>
<td></td>
<td>[3,6,11,15]</td>
<td></td>
<td></td>
<td>[6,11,12,15]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>[2,4,8,10,12]</td>
<td></td>
<td></td>
<td>[2,7,9,10,14]</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>[1,5,8,10,15]</td>
<td>3.44</td>
<td>5</td>
<td>[1,3,4,8,15,13]</td>
<td>2.8158</td>
</tr>
<tr>
<td></td>
<td>[3,9,7,13,14]</td>
<td></td>
<td></td>
<td>[2,6,11,12]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>[2,4,6,11,12]</td>
<td></td>
<td></td>
<td>[5,7,9,10,14]</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>[1,3,9,12,13]</td>
<td>3.66</td>
<td>7</td>
<td>[1,3,4,5,8,13]</td>
<td>2.79999</td>
</tr>
<tr>
<td></td>
<td>[4,5,7,10,11]</td>
<td></td>
<td></td>
<td>[6,11,12,15]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>[2,6,8,14,15]</td>
<td></td>
<td></td>
<td>[2,7,9,10,14]</td>
<td></td>
</tr>
</tbody>
</table>
### TABLE 9

**Iteration Log for the Approximation Procedure**

**Number of Parts = 15**

**Random Starting Partition = 1**

<table>
<thead>
<tr>
<th>ITR INITIAL ALLOCATION</th>
<th>INTERMEDIATE OBJ R AND FINAL FN. NEW ALLOCATIONS Z(R,X) CDC Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 [1, 6, 7, 12, 13]</td>
<td>* Based on range analysis choose R=3.55</td>
</tr>
<tr>
<td></td>
<td>[2, 5, 8, 11, 14]</td>
</tr>
<tr>
<td></td>
<td>[3, 4, 9, 10, 14]</td>
</tr>
<tr>
<td>MF=1</td>
<td>3.55 [2, 5, 8, 11, 14, -73 3.349 High, 12] Choose [3, 4, 9, 10, 15, 7, 13] [6]</td>
</tr>
<tr>
<td>CDC=3.665</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3.349 Same as above  0  3.349 OK</td>
</tr>
<tr>
<td>1 [6]</td>
<td>* Based on range analysis choose R=3.349</td>
</tr>
<tr>
<td>[2, 5, 8, 11, 14, 1, 12]</td>
<td></td>
</tr>
<tr>
<td>[3, 4, 9, 10, 15, 7, 13]</td>
<td></td>
</tr>
<tr>
<td>MF=3</td>
<td>3.20 [6, 3, 15] 0  3.201 OK</td>
</tr>
<tr>
<td>CDC=3.349</td>
<td>[1, 2, 5, 8, 11, 12, 14]</td>
</tr>
<tr>
<td></td>
<td>[4, 7, 9, 10, 13]</td>
</tr>
<tr>
<td>2 [3, 6, 15]</td>
<td>* Based on the range analysis choose R = 2.90</td>
</tr>
<tr>
<td>[1, 2, 5, 8, 11, 12, 14]</td>
<td></td>
</tr>
<tr>
<td>[4, 7, 9, 10, 13]</td>
<td>2.90 [3, 6, 15, 8] -1.6 2.895 High [2, 11, 12] Choose [4, 7, 9, 10, 13, 1, 5, 14]</td>
</tr>
<tr>
<td>MF=2</td>
<td>2.895 Same as above  0  2.895 OK</td>
</tr>
<tr>
<td>CDC=3.201</td>
<td></td>
</tr>
</tbody>
</table>

(Contd.)
<table>
<thead>
<tr>
<th>Itr No.</th>
<th>INITIAL ALLOCATION</th>
<th>INTERMEDIATE AND FINAL ALLOCATIONS</th>
<th>OBJ Z(R,X)</th>
<th>NEW CDC Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>[3,6,8,15] [2,11,12] [1,4,5,9,10,13, 14]</td>
<td>Based on range analysis, choose R=2.70</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2.70 [3,6,8,15, 1,4]</td>
<td>Choose 2.869</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>[2,11,12]</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>[5,7,9,10, 13,14]</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.869 Same as above -0.01 2.869 OK</td>
</tr>
<tr>
<td></td>
<td>[3,6,8,15, 1,4] [2,11,12] [5,7,9,10,13, 14]</td>
<td>* Based on range analysis choose R = 2.85</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>MF=3 CDC=2.895</td>
<td></td>
<td>2.85 Same as 6.45 2.869 Low, Initial Choose 2.869</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.869 Same as -0.01 2.869 OK initial</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Family configuration not changed</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Choose the family with next highest Dk. i.e MF=1 R = 2.85</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2.85 [1,3,4,8, 15]</td>
<td>Choose 2.845 OK</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>[2,6,11,12]</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>[7,9,10,5,14]</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>13,1,5,14]</td>
<td></td>
</tr>
</tbody>
</table>

(Contd.)
### TABLE 9 (Continued)

| Itr No. | INITIAL ALLOCATION | R
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>[1, 3, 4, 8, 15]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>[2, 11, 12, 6]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>[5, 7, 9, 10, 14]</td>
<td></td>
</tr>
<tr>
<td>MF=3</td>
<td>CDC=2.845</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>R=2.845</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.845</td>
<td></td>
</tr>
<tr>
<td></td>
<td>[1, 3, 4, 8, 15]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>[2, 11, 12, 6]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>[5, 7, 9, 10, 14]</td>
<td></td>
</tr>
</tbody>
</table>

* Based on range analysis choose

R=2.845

-9.3  2.815  Hogh,

Choose

2.815  [13, 15]

2.815  [2, 11, 12, 6]

-14]

2.815 Same as above -0.01  2.815 OK above

| Itr No. | INITIAL ALLOCATION | R
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>[1, 3, 4, 8, 15]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>[2, 11, 12, 6]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>[5, 7, 9, 10, 14]</td>
<td></td>
</tr>
<tr>
<td>MF=1</td>
<td>CDC=2.815</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>R=2.815</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.815</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.815</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Same as Initial 0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Same as Initial 0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.815 OK</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Family configuration has not changed.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Choose the family with next highest</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D_k. i.e., MF=3 Based on the range</td>
<td></td>
</tr>
<tr>
<td></td>
<td>analysis choose R =2.815</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.815 Same as initial 0 2.815 OK</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Family configuration has not changed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Choose the next family i.e., MF=1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Based on the range analysis, choose</td>
<td></td>
</tr>
<tr>
<td></td>
<td>R=2.815</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.815 Same as Initial 0 2.815 OK</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Family configuration is not changed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>All the families are considered.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>STOP</td>
<td></td>
</tr>
</tbody>
</table>

---
iteration log maintained for the first trial is given in Table 9. The iteration logs for other trials are listed in Appendix D.

From Table 8, two points are evident:

i) The final CDG's obtained by the procedure are very close to each other (similar result has been reported in [10]).

ii) The solution obtained is close to the optimal solution ($R^*$ should be in the range $2.45 - 2.7999$, since $L = 2.45$).

The solution times required for different subproblems in each of the iterations are given in Table 10. The total times required are:

<table>
<thead>
<tr>
<th>Starting Config</th>
<th>Approximate Total Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.48 Min</td>
</tr>
<tr>
<td>2</td>
<td>8.03 Min</td>
</tr>
<tr>
<td>3</td>
<td>2.31 Min</td>
</tr>
<tr>
<td>4</td>
<td>3.90 Min</td>
</tr>
<tr>
<td>5</td>
<td>1.53 Min</td>
</tr>
</tbody>
</table>

6.2.3 Search Log

The problem is then solved using $R = 2.7999$ (the value obtained through the approximate solution procedure) as the upper limit on the value of $R^*$. i.e., the search range chosen is $[2.45, 2.799]$.

First the problem is solved at the midpoint of this
range, i.e., at 2.625.

The optimal solution to the problem \( P(2.625) = 173.25 \)
\( \Rightarrow R^* > 2.625 \)

Hence, the value 2.7999 is chosen as the \( R \) for next \( P(R) \).

The problem \( P(2.7999) \) gives the optimal value as 0.

\( \Rightarrow R^* = 2.7999 \)

The solution times required for the above two
problems \( P(R) \) are:
- \( P(2.625) \) ----- 21.28 Mins
- \( P(2.799) \) ----- 22.29 Mins.

It also turns out that the optimal solution is
obtained in some of the trials of the approximation
procedure, and close to optimal solution in other trials.

Thus the solution obtained through the approximation
procedure is "good" (in fact optimal in this case. It can
not be guaranteed to be optimal, however). Also, the
solution time required for the problems \( P(R) \) confirms the
importance of starting the search at a "good" solution. If
the problem were to be attempted with some other initial
value as the upper limit, say with \( R=3.00 \), then the number
of problems \( P(R) \) solved would have been more (each of them
taking a time of about 20 Minutes) resulting in a larger
solution time.
6.2.4 Some computational considerations

The solution times (CPU) for the series of problems solved in the course of approximation procedure iterations are listed in Table 10 along with the number of integer variables in each of the problems.

Whilst these times should be strictly associated with a specific Package-Computer combination (SAS/OR and IBM 4381), they are indicative of the computational behaviour of these problems viz.,

i) The solution times increase even for a small increase in the number of integer variables.

ii) For the same number of integer variables, different problems require different solution times, sometimes varying widely from each other.

Several problems have been solved using variations of the original data. It is observed that in all the cases the solution procedure converges to the exact solution. And, it appears that for the values of R close to the value of \( R^* \), the optimal allocations obtained by \( P(R) \) would also be the optimal allocation corresponding to \( R^* \).

The problems \( P(R) \) are solved steps, halting the IP routine intermittently to check the sign of the objective function any intermediate solution that might have been found. If the sign turns out to be negative, problem \( P(R) \)
<table>
<thead>
<tr>
<th>Starting Config</th>
<th>Iteration</th>
<th>Problem No</th>
<th># of Integer Variables</th>
<th>Solution time Min:Sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>10</td>
<td>0</td>
<td>3.38</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>10</td>
<td>0</td>
<td>3.38</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>14</td>
<td>0</td>
<td>26.43</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>14</td>
<td>0</td>
<td>7.52</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>16</td>
<td>0</td>
<td>36.27</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>16</td>
<td>0</td>
<td>36.27</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>12</td>
<td>0</td>
<td>5.10</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>12</td>
<td>0</td>
<td>5.09</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>12</td>
<td>0</td>
<td>4.89</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>12</td>
<td>0</td>
<td>4.91</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>12</td>
<td>0</td>
<td>4.89</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>12</td>
<td>0</td>
<td>5.20</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>10</td>
<td>0</td>
<td>3.38</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>8</td>
<td>0</td>
<td>2.12</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td></td>
<td></td>
<td></td>
<td><strong>2.48 Min</strong></td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>12</td>
<td>0</td>
<td>4.89</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>12</td>
<td>0</td>
<td>4.79</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>16</td>
<td>0</td>
<td>36.27</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>16</td>
<td>0</td>
<td>36.27</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>20</td>
<td>1</td>
<td>32.22</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>20</td>
<td>1</td>
<td>35.41</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>20</td>
<td>1</td>
<td>37.85</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>16</td>
<td>0</td>
<td>21.77</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>16</td>
<td>0</td>
<td>18.80</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>10</td>
<td>0</td>
<td>3.09</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>10</td>
<td>0</td>
<td>3.09</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>16</td>
<td>0</td>
<td>17.13</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>16</td>
<td>0</td>
<td>14.61</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>12</td>
<td>0</td>
<td>5.10</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>12</td>
<td>0</td>
<td>5.09</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>12</td>
<td>0</td>
<td>4.89</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>12</td>
<td>0</td>
<td>4.91</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>12</td>
<td>0</td>
<td>4.89</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>12</td>
<td>0</td>
<td>5.20</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>10</td>
<td>0</td>
<td>3.38</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>8</td>
<td>0</td>
<td>2.12</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td></td>
<td></td>
<td></td>
<td><strong>8.0295 Min</strong></td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>12</td>
<td>0</td>
<td>5.61</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>12</td>
<td>0</td>
<td>5.58</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>16</td>
<td>0</td>
<td>25.87</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>16</td>
<td>0</td>
<td>20.05</td>
</tr>
<tr>
<td>Starting Config</td>
<td>Iteration</td>
<td>Problem No.</td>
<td># of Integer Variables</td>
<td>Solution time Min : Sec</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------</td>
<td>-------------</td>
<td>------------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>16</td>
<td>0 : 16.14</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>16</td>
<td>0 : 26.26</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>12</td>
<td>0 : 5.15</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>10</td>
<td>0 : 3.38</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>10</td>
<td>0 : 3.38</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>12</td>
<td>0 : 5.58</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>10</td>
<td>0 : 3.50</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>10</td>
<td>0 : 3.50</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>14</td>
<td>0 : 7.75</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>14</td>
<td>0 : 7.75</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>12</td>
<td>0 : 5.05</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>10</td>
<td>0 : 3.50</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>8</td>
<td>0 : 1.91</td>
<td></td>
</tr>
<tr>
<td>TOTAL</td>
<td></td>
<td></td>
<td></td>
<td>2.314 Min</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>10</td>
<td>0 : 2.98</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>10</td>
<td>0 : 2.92</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>18</td>
<td>1 : 19.61</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>18</td>
<td>1 : 57.18</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>12</td>
<td>0 : 5.04</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>10</td>
<td>0 : 3.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>10</td>
<td>0 : 2.81</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>12</td>
<td>0 : 4.91</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>12</td>
<td>0 : 4.89</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>12</td>
<td>0 : 5.20</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>10</td>
<td>0 : 3.38</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>8</td>
<td>0 : 2.12</td>
<td></td>
</tr>
<tr>
<td>TOTAL</td>
<td></td>
<td></td>
<td></td>
<td>3.900 Min</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>10</td>
<td>0 : 3.25</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>10</td>
<td>0 : 3.38</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>10</td>
<td>0 : 3.29</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>14</td>
<td>0 : 8.03</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>14</td>
<td>0 : 8.21</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>12</td>
<td>0 : 6.72</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>10</td>
<td>0 : 2.88</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>14</td>
<td>0 : 8.14</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>14</td>
<td>0 : 8.69</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>10</td>
<td>0 : 3.03</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>10</td>
<td>0 : 2.90</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>14</td>
<td>0 : 7.45</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>14</td>
<td>0 : 7.19</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>12</td>
<td>0 : 5.28</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>10</td>
<td>0 : 2.85</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>12</td>
<td>0 : 3.05</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>10</td>
<td>0 : 3.50</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>8</td>
<td>0 : 1.91</td>
<td></td>
</tr>
<tr>
<td>TOTAL</td>
<td></td>
<td></td>
<td></td>
<td>1.529 Min</td>
</tr>
</tbody>
</table>
is terminated since the optimal solution must be negative. This approach is helpful in cutting down the actual time required to solve the problem. For this reason, intuitively it is better to approach the value of \( R \) from the negative side.

The solutions obtained by solving the problems \( P(R) \) can be used to get a short-cut in search procedure compared to binary principle. At each value of \( R \), at least one feasible solution is found before deciding on the next course of action. Each feasible solution gives a particular allocation of parts to the families. For this allocation, the value of \( A(X)/B(X) \) can be calculated. The value of \( R^* \) should be less than or equal to the calculated value for this feasible solution. The calculated value \( A(X)/B(X) \) thus is helpful in the search for \( R^* \).

6.3 **Machine Group Allocation—An Example**

The machine group allocation problem is illustrated by an example. The infeasibility occurring in the problem is resolved using the formulation in Section 5.3.2.

6.3.1 **Routing Information**

The routing information in the form of matrix \( A_{jk} \) as described in section 5.1.3 is given in Table 11.
this type of information about the feasibility of certain operations on the available machines is assumed to be available.

6.3.2 Solution Procedure

The machine group allocation problem has been solved for the part families formed with fifteen parts.

The family configuration obtained by the part grouping in Section 6.2 is given below:

Family 1 : parts 1, 3, 4, 5, 8, 13
Family 2 : parts 6, 11, 12, 15
Family 3 : parts 2, 7, 9, 10, 14

From the routing matrix for these parts (Table 11) a list of multiplication terms for each family is generated as explained in Section 5.1.4. For example, the operation 1 and 2 of the part 6 can be done on machines 1 and 4 respectively, and the product term $I_{12}I_{142}$ represents this routing. The sample of such a list is given in Table 12.

The size of the problem is:

\[ M=12 \quad K=3 \]

- Number of integer variables = $12 \times 2 = 24$
- Number of free variables = 12
- Number of product term variables representing the routings for the parts = 117
<table>
<thead>
<tr>
<th>Part No.</th>
<th># Of Opns</th>
<th>Opns</th>
<th>H1</th>
<th>H2</th>
<th>H3</th>
<th>H4</th>
<th>V1</th>
<th>V2</th>
<th>V3</th>
<th>B1</th>
<th>B2</th>
<th>B3</th>
<th>D1</th>
<th>D2</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td></td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td></td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td></td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td></td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>3</td>
<td></td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>3</td>
<td></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td></td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>3</td>
<td></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>12</td>
<td>3</td>
<td></td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>13</td>
<td>3</td>
<td></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>14</td>
<td>4</td>
<td></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>15</td>
<td>3</td>
<td></td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>No.</td>
<td>Product Term</td>
<td>6</td>
<td>11</td>
<td>12</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<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>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>I₁₂ I₄₂</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>I₁₂ I₅₂</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>I₁₂ I₆₂</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>I₁₂ I₉₂</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>I₁₁₂ I₄₂</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>I₁₁₂ I₅₂</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>I₁₁₂ I₆₂</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>I₁₁₂ I₉₂</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>I₁₂ I₄₂ I₉₂</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>I₃₂ I₄₂ I₁₂</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>I₃₂ I₄₂ I₉₂</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>I₃₂ I₉₂ I₁₂</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>I₃₂ I₉₂</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>I₁₂ I₁₂₂</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>I₁₂ I₁₂₂ I₉₂</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>I₄₂ I₉₂</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>I₄₂ I₁₂₂ I₁₂</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>I₄₂ I₁₂₂ I₉₂</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>I₁₂ I₁₂₂ I₃₂</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>I₁₂ I₁₂₂ I₁₁₂</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>I₁₂ I₅₂ I₃₂</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>I₁₂ I₅₂ I₁₁₂</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>I₃₂ I₁₂₂</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>I₃₂ I₁₂₂ I₁₁₂</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>I₃₂ I₅₂</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>I₃₂ I₅₂ I₁₁₂</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The problem is infeasible with the constraint of allocating one machine to one family only. The mathematical model developed in Section 5.3.2 is applied to find out the machine causing infeasibility. The value of the optional weightage $D$ is chosen to be 1. The optimal value of the objective function is 2. Two of the machines are identified to be the ones causing infeasibility.

Machine 2 is absolutely essential for families 2 and 3. Machine 6 is absolutely essential for families 1 and 3. The allocation constraint for these two machines was relaxed by giving a RHS value of 2 in the corresponding constraints of type (27). Problem $P_1$ is then solved to find the optimal solution to the allocation problem. The optimal machine group allocation (after the relaxation for machines 2 and 3) to the families is given below.

Maximum number of routings achieved = 38

Machines allotted to family 1: 3, 5, 6, 8, 9, 12
Machines allotted to family 2: 2, 1, 4, 11
Machines allotted to family 3: 2, 6, 7, 10

6.4 Discussion of Results

The cell formation is an initial specification problem in the pre-production planning stage. This research presents new formulations for this problem.

The examples illustrating the formulations involve
grouping of fifteen parts into three families and then allocating twelve machines to these families. The problems are solved on an IBM 4381 computer using the integer programming (IP) routine of SAS/OR package.

The part family formation example for the above data has $15 \times 3 = 45$ decision variables. In the input to the IP routine $15 \times 2 = 30$ variables are explicitly stated to be of 0-1 integer type. The remaining 15 are forced to take 0-1 values due to constraints of type (i) in the formulation. The solution time for the problems solved during the search procedure (Problem P(R)) is about 20-25 Minutes.

The solution times for several problems of smaller size are listed in Table 10.

An example of twenty-five parts using variations in the data given in Appendix B has been attempted. The solution time for the problems solved during the search procedure has been found to be in excess of 150 Min. The experimentation with larger problems has been limited. However, based on the results it appears that with a comparable computer-package combination, problems with 40-50 variables could be solved in a "reasonable" time. In physical terms the corresponding problem size is 20-25 parts to be grouped into 2-3 families.

Further computational experimentation is necessary for the larger problems considering the following issues.
1) **Branch and Bound Strategy:** Several search heuristic options are available for the rules of selecting branching nodes and branching variable at the nodes. The experience with the smaller problems can not be directly extrapolated to larger problems, since the strategy that works well for a particular problem size may not work well as the problem size increases [29].

II) **Linearization Strategy:** Some variations involving reductions in the number of extra constraints generated have been suggested by Glover and Woolsey [15]. A brief discussion of these is provided by Stecke [30]. The experimentation with these different strategies could be a possibility for the larger problems. (The program written to generate the input for the problem P(R) provides options for implementing different strategies as indicated in [30]).

The bounds established on the objective function could be used to determine the maximum possible variation of the objective function value for any feasible solution from the optimal value. This would be useful especially for large problems.

The approximate solution procedure developed is an extension of the "single move" heuristic to a "multiple
move case. The results from the procedure have been found to be near-optimal in the problems solved. Also, starting with different configurations the procedure converged to solutions with objective function values very close to each other. This result is comparable to the "single move" implementation of the heuristic in [12].

The solution obtained through the approximation procedure provides an upper limit on the objective function value.

The formulation of machine group allocation has $12 \cdot 3 = 36$ decision variables. In the input to the LP routine $12 \cdot 2 = 24$ variables are explicitly stated to be of 0-1 type. The remaining 12 are forced to take 0-1 values due to the assignment constraints. The solution time for the problem is about 2-3 Min. The formulation for identifying the machines causing infeasibility in the problem required less than 1 Min.

The implementation of the formulations gives the system specification in terms of the cell configuration and the parts manufactured in the cells. The contributions of this research are:

- Defining a dissimilarity coefficient as the objective function of the part grouping formulation.
- Developing an algorithm for finding the bounds on the above objective function.
Extension of a clustering method of "single move" type to a "multiple move" case.

Consideration of the availability of alternative machines and the routing diversity in machine grouping problem.

Developing a mathematical model to identify the machines causing infeasibility in the machine group allocation problem.

The computational aspects for larger problems have to be further tested for larger problems as explained earlier. Also a matter of consideration could be to impose other constraints on grouping. For example, one of the constraints could be to consider the quantities of the parts to be manufactured with a view to balance the work load in the cells.

Some of the direct consequences of grouping are reduction in the work in progress, reduced lead times and reduced scheduling complexity. A study by Purcheck [26] confirms that the cellular systems have better "operating characteristics" as measured by these factors. The reduced work in progress and reduced lead times result in financial gains. An approach for analysing such gains due to grouped system has been suggested by Boucher [3]. Such an analysis could be done after the cell formation problem has been solved using the formulations presented.
CHAPTER VII

SUMMARY

This research deals with the initial specification decisions in the pre-production planning stage for Flexible Manufacturing Systems. The problems of part family formation and machine group allocation have been formulated as 0-1 integer programming models.

The formulation of part family formation is a fractional program. The dissimilarity between the parts in terms of processing requirements has been represented by a coefficient and is defined as a function of 0-1 variables. By identifying the specific nature of the objective function a general search principle has been suitably adopted for solving the formulation.

As a method for providing a starting solution to the search procedure, an extension to a clustering principle reported in the literature has been developed. This extension is based on the fractional model.

The concept of routing flexibility, or the number of available routes for the parts within the cellular system has been adopted in the machine group allocation.
This aspect has not been considered by the Group Technology researchers in conventional systems.

The formulations have been applied to a set of realistic problem data. Several problems have been solved.

The computational experience with these problems indicates that the formulations are applicable to FMS installations manufacturing low or medium variety of parts.

As indicated in Chapter III, many of the systems which operate in tandem with conventional facilities have been, in general, used for the manufacture of critical, high value parts. Many of the FMSs reported in the literature fall into this category. The proposed procedure is applicable for these systems.

The solution procedure developed for the fractional programming model is also applicable in other clustering applications where the pairwise ratio criteria could be used.

In summary, the main contributions of this thesis are the development of a new formulation for part family formation, extension of a heuristic procedure in clustering and adopting the availability of alternative routings for the parts as the criterion in machine grouping.
REFERENCES


APPENDIX A

ALTERNATIVE DEFINITIONS FOR THE OVERALL DISSIMILARITY COEFFICIENT
ALTERNATIVE DEFINITIONS FOR THE OVERALL DISSIMILARITY COEFFICIENT

The overall dissimilarity coefficient defined as the objective function in the part family formation problem is an average of all the pairwise dissimilarity coefficients. The overall measure also can also be defined as follows:

\[ a) \quad CDC_1 = \frac{1}{N.(N-1)} \sum_{k=1}^{K} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \left( \frac{d_{ij}/s_{ij}}{X_{ik} \cdot X_{jk}} \right) \]

\[ b) \quad CDC_2 = \frac{1}{K} \left[ \sum_{k=1}^{K} \left( \frac{\sum_{i=1}^{N-1} \sum_{j=i+1}^{N} d_{ij} \cdot X_{ik} \cdot X_{jk}}{\sum_{i=1}^{N-1} \sum_{j=i+1}^{N} s_{ij} \cdot X_{ik} \cdot X_{jk}} \right) \right] \]

\[ c) \quad CDC_3 = \frac{1}{K} \sum_{k=1}^{K} \left[ \frac{\sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \left( d_{ij}/s_{ij} \right) \cdot X_{ik} \cdot X_{jk}}{n_k} \right] \]

where,

\[ n_k = \sum_{i=1}^{N} X_{ik} \]

\( CDC_1, CDC_2 \) and \( CDC_3 \) are the other possible definitions of the objective of the minimization of dissimilarities and maximization of similarities between the parts.
A formulation with CDC$_1$ as the objective function would be similar to the problem P(R).

The formulations CDC$_2$ and CDC$_3$ as objective functions would be more complicated to solve. Both these functions can be simplified into a single ratio of non-linear integer functions. The difference would be that in these formulations, the functions would have polynomial terms of higher degree (unlike the formulation for CDC which has only the product terms of degree two). Hence, the method developed for establishing the region (L, U) would not be applicable to these formulations. The general search principle however, still holds in these cases.

The reasons for adopting CDC as the objective function are:

- Using as objective function a similar expression as reported in [12].
- The expression CDC incorporates a weighted average of all $d_{ij}/s_{ij}$ values.
APPENDIX B

PART SKETCHES AND PROCESS DETAILS
PART 01  HOUSING

Scale 1:6
### PROCESS DETAILS

**PART NO.: 1**

**PART NAME:** HOUSING

<table>
<thead>
<tr>
<th>OPERATION NO.</th>
<th>DESCRIPTION OF THE OPERATION</th>
<th>TOOL(S) REQUIRED</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rough Mill Surface (A)</td>
<td>M501</td>
<td></td>
</tr>
<tr>
<td>Finish Mill Surface (A)</td>
<td>M502, M701</td>
<td></td>
</tr>
<tr>
<td>Center Hole (1)</td>
<td>D201</td>
<td></td>
</tr>
<tr>
<td>Drill 42 Dia Hole Thro&quot; (1)</td>
<td>D142</td>
<td></td>
</tr>
<tr>
<td>Ream 42 Dia Hole Thro&quot;</td>
<td>R142</td>
<td></td>
</tr>
<tr>
<td>Chamfer</td>
<td>D109, M602</td>
<td></td>
</tr>
<tr>
<td>Rough Mill Edge (B)</td>
<td>M101, M401</td>
<td></td>
</tr>
<tr>
<td>Rough Mill Edge (C)</td>
<td>M101, M401</td>
<td></td>
</tr>
<tr>
<td>Rough Mill Edge (D)</td>
<td>M101, M401</td>
<td></td>
</tr>
</tbody>
</table>

(1)

| Rough Mill Base Projections | M102 | |
| Finish Mill Base Projections | M103 | |
| Center Holes (4) | D202 | |
| Drill 30 Dia Holes Thro" (4) | D130 | |
| Ream 30 Dia Holes Thro" (4) | R130 | |
| Chamfer (4) | D109 | |
| Peripheral Mill Surface (J) | M301, M101 | |
| Finish Mill Surface (J) | M302, M102 | |
| Rough Mill Face (C) | M301, M102 | |
| Finish Mill Face (C) | M301, M102 | |
| Bore 42 Dia Hole Thro" | B108 | |
| Finish Bore 42 Dia Hole Thro" | B109 | |
| Counter Bore 72 Dia 36 Deep | B101 | |
| Chamfer | M702 | |

(2)

| Contour Mill (F) | M603, M108 | |
| Face Mill Surface (H) | M503 | |
| Finish Mill Surface (H) | M504 | |
| Rough Mill Surface (I) | M603, M108 | |
| Finish Mill Surface (I) | M503 | |
| Bore 48 Dia Holes Thro" (2) | B108 | |
| Finish Bore 48 Dia Holes Thro" (2) | B115 | |
| Ream 48 Dia Holes Thro" (2) | R148 | |
| Counter Bore 78 Dia Inside | B101 | |
| Counter Bore 78 Dia Outside | B106 | |

(3)
### PROCESS DETAILS

**PART NO.:** 2  
**PART NAME:** BASE BLOCK

<table>
<thead>
<tr>
<th>OPERATION NO.</th>
<th>DESCRIPTION OF THE OPERATION</th>
<th>TOOL(S) REQUIRED</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>Rough Mill Bottom Face (A)</td>
<td>M101</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Bottom Face (A)</td>
<td>M103</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Sides</td>
<td>M104</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Sides</td>
<td>M105</td>
</tr>
<tr>
<td></td>
<td>Mill Contours (4)</td>
<td>M901, M602</td>
</tr>
<tr>
<td></td>
<td>Face Mill Surface (D)</td>
<td>M501</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Surface (D)</td>
<td>M501</td>
</tr>
<tr>
<td></td>
<td>Face Mill Surface (B)</td>
<td>M506, M508</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Surface (B)</td>
<td>M506, M508</td>
</tr>
<tr>
<td></td>
<td>Center Holes (4)</td>
<td>D201</td>
</tr>
<tr>
<td>(2)</td>
<td>Drill Dia 28 Holes Thru (4)</td>
<td>D125</td>
</tr>
<tr>
<td></td>
<td>Ream Dia 28 Holes Thru (4)</td>
<td>R128</td>
</tr>
<tr>
<td></td>
<td>Deburr</td>
<td>S104</td>
</tr>
<tr>
<td></td>
<td>Face Mill Boss</td>
<td>M101, M402</td>
</tr>
<tr>
<td></td>
<td>Rough Bore 34 Dia Hole Thru</td>
<td>B101</td>
</tr>
<tr>
<td></td>
<td>Finish Bore 34 Dia Hole Thru</td>
<td>B109</td>
</tr>
<tr>
<td></td>
<td>Step Bore Outside Step (Rough)</td>
<td>B105</td>
</tr>
<tr>
<td></td>
<td>Step Bore Inside Step (Rough)</td>
<td>B105</td>
</tr>
<tr>
<td></td>
<td>Finish Bore Outside Step</td>
<td>B104</td>
</tr>
<tr>
<td></td>
<td>Finish Bore Inside Step</td>
<td>B104</td>
</tr>
</tbody>
</table>
## PROCESS DETAILS

**PART NO.** 3  
**PART NAME:** PULLEY BLOCK

<table>
<thead>
<tr>
<th>OPERATION NO.</th>
<th>DESCRIPTION OF THE OPERATION</th>
<th>TOOL(S) REQUIRED</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Face Mill Periphery (B)</td>
<td>M403</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Periphery (B)</td>
<td>M404</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Face (C)</td>
<td>M701</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Face (C)</td>
<td>M702</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Grove (D)</td>
<td>M401</td>
</tr>
<tr>
<td></td>
<td>Mill Peripheral Edges</td>
<td>M412</td>
</tr>
<tr>
<td>(1)</td>
<td>Bore 50 Dia Hole Throat</td>
<td>B108</td>
</tr>
<tr>
<td></td>
<td>Finish Bore 50 Dia Throat</td>
<td>B109</td>
</tr>
<tr>
<td></td>
<td>Counter Bore 35 Dia 8 Deep</td>
<td>B101</td>
</tr>
<tr>
<td></td>
<td>Finish Counter Bore</td>
<td>B102</td>
</tr>
<tr>
<td></td>
<td>Enlarge Bore Dia Throat from step</td>
<td>B115</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Side A</td>
<td>M405</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Side A</td>
<td>M406</td>
</tr>
<tr>
<td></td>
<td>Center Holes (4)</td>
<td>D203</td>
</tr>
<tr>
<td>(2)</td>
<td>Drill Holes 30 Dia Throat (4)</td>
<td>D128</td>
</tr>
<tr>
<td></td>
<td>Ream Holes 30 Dia Throat (4)</td>
<td>R130</td>
</tr>
<tr>
<td></td>
<td>Tap Holes 30 Dia Throat</td>
<td>T130</td>
</tr>
<tr>
<td></td>
<td>Finish Bore 80 Dia</td>
<td>B106</td>
</tr>
<tr>
<td></td>
<td>Counter Bore Dia 105</td>
<td>B112</td>
</tr>
<tr>
<td></td>
<td>Chamfer edge</td>
<td>M702,M712</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Periphery</td>
<td>M101</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Outside Periphery</td>
<td>M102</td>
</tr>
</tbody>
</table>
### PROCESS DETAILS

**PART NO.:** 4  
**PART NAME:** FLANGE

<table>
<thead>
<tr>
<th>OPERATION NO.</th>
<th>DESCRIPTION OF THE OPERATION</th>
<th>TOOL(S) REQUIRED</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>Rough Mill Bottom Face</td>
<td>M501, M518</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Bottom Face</td>
<td>M502</td>
</tr>
<tr>
<td></td>
<td>Center Hole (1)</td>
<td>D202</td>
</tr>
<tr>
<td></td>
<td>Drill 50 Dia Hole Thru</td>
<td>D150</td>
</tr>
<tr>
<td></td>
<td>Rough Bore 110 Dia Hole Thru</td>
<td>B108</td>
</tr>
<tr>
<td></td>
<td>Finish Bore 110 Dia Hole Thru</td>
<td>B112</td>
</tr>
<tr>
<td></td>
<td>Counter Bore 175 Dia Deep</td>
<td>B112</td>
</tr>
<tr>
<td></td>
<td>Finish 175 Dia Bore Deep</td>
<td>B112</td>
</tr>
<tr>
<td></td>
<td>Chamfer</td>
<td>M701, M712</td>
</tr>
<tr>
<td>(2)</td>
<td>Rough Mill Sides</td>
<td>M211, M101</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Sides</td>
<td>M212, M102</td>
</tr>
<tr>
<td></td>
<td>Face Mill Top Surface</td>
<td>M501, M502</td>
</tr>
<tr>
<td></td>
<td>Counter Bore 175 Dia Deep</td>
<td>B112</td>
</tr>
<tr>
<td></td>
<td>Finish Bore 175 Dia Deep</td>
<td>B112</td>
</tr>
<tr>
<td></td>
<td>Center Holes (4)</td>
<td>D202</td>
</tr>
<tr>
<td></td>
<td>Drill 20 Dia Holes Thru (4)</td>
<td>D120</td>
</tr>
<tr>
<td></td>
<td>Deburr</td>
<td>S109</td>
</tr>
<tr>
<td></td>
<td>Tap 20 Dia Holes Thru (4)</td>
<td>T120</td>
</tr>
<tr>
<td></td>
<td>Face Mill Bosses</td>
<td>M503</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Bosses</td>
<td>M503</td>
</tr>
<tr>
<td></td>
<td>Center Holes (2)</td>
<td>D201</td>
</tr>
<tr>
<td></td>
<td>Drill 20 Dia Holes Thru (2)</td>
<td>D120</td>
</tr>
<tr>
<td>(3)</td>
<td>Mill Periphery of Projections</td>
<td>M301</td>
</tr>
<tr>
<td></td>
<td>Finish Periphery of Projections</td>
<td>M302</td>
</tr>
<tr>
<td></td>
<td>Face Mill Top Surface(Rough)</td>
<td>M501</td>
</tr>
<tr>
<td></td>
<td>Mill Periphery Of Bosses</td>
<td>M401, M405</td>
</tr>
</tbody>
</table>
PART #5                BASE

Scale 1:8
## PROCESS DETAILS

**PART NO:** 5  
**PART NAME:** BASE

<table>
<thead>
<tr>
<th>OPERATION NO.</th>
<th>DESCRIPTION OF THE OPERATION</th>
<th>TOOL(S) REQUIRED</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>Face Mill Side (A)</td>
<td>M102, M401</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Side (A)</td>
<td>M103, M105</td>
</tr>
<tr>
<td></td>
<td>Angle Mill Side (B)</td>
<td>M602</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Side (B)</td>
<td>M603</td>
</tr>
<tr>
<td>(2)</td>
<td>Rough Mill Bottom Face (C)</td>
<td>M501, M513</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Bottom Face(C)</td>
<td>M503, M502</td>
</tr>
<tr>
<td></td>
<td>Face Mill Top Edge</td>
<td>M101, M701</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Top Edge</td>
<td>M103, M702</td>
</tr>
<tr>
<td></td>
<td>Peripheral Mill Boss</td>
<td>M401</td>
</tr>
<tr>
<td></td>
<td>Face Mill Boss</td>
<td>M401</td>
</tr>
<tr>
<td></td>
<td>Center Holes (15)</td>
<td>D201</td>
</tr>
<tr>
<td></td>
<td>Drill 8 Dia Holes (15)</td>
<td>D108</td>
</tr>
<tr>
<td></td>
<td>Deburr</td>
<td>S101</td>
</tr>
<tr>
<td>(3)</td>
<td>Ream 8 Dia Holes (15)</td>
<td>R108</td>
</tr>
<tr>
<td></td>
<td>Center Hole (1)</td>
<td>D201</td>
</tr>
<tr>
<td></td>
<td>Drill 50 Dia Hole Thro'</td>
<td>D150</td>
</tr>
<tr>
<td></td>
<td>Bore 56 Dia Hole</td>
<td>B106</td>
</tr>
<tr>
<td></td>
<td>Finish Bore 56 Dia Hole</td>
<td>B109</td>
</tr>
<tr>
<td></td>
<td>End Mill Pocket (rough)</td>
<td>M501, M402</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Pocket</td>
<td>M508, M402</td>
</tr>
<tr>
<td></td>
<td>Peripheral Mill Sides</td>
<td>M406</td>
</tr>
<tr>
<td></td>
<td>Contour Mill Projections</td>
<td>M610, M602</td>
</tr>
<tr>
<td></td>
<td>Center Hole (1)</td>
<td>D203</td>
</tr>
<tr>
<td></td>
<td>Drill 50 Dia Hole (1)</td>
<td>D150</td>
</tr>
<tr>
<td></td>
<td>Drill 70 Dia Hole</td>
<td>D170</td>
</tr>
<tr>
<td></td>
<td>Bore 120 Dia Hole</td>
<td>B108</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Sides B,C and D</td>
<td>M301, M305</td>
</tr>
</tbody>
</table>
### PROCESS DETAILS

**PART NO.: 6**

**PART NAME: CAP**

<table>
<thead>
<tr>
<th>OPERATION NO.</th>
<th>DESCRIPTION OF THE OPERATION</th>
<th>TOOL(S) REQUIRED</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>Rough Mill Bottom Face (A)</td>
<td>M501, M504</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Bottom Face (A)</td>
<td>M502</td>
</tr>
<tr>
<td></td>
<td>End Mill Grove</td>
<td>M415, M412</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Grove</td>
<td>M416</td>
</tr>
<tr>
<td></td>
<td>Center Holes (4)</td>
<td>D201</td>
</tr>
<tr>
<td></td>
<td>Drill 18 Dia Holes Thru (4)</td>
<td>D118</td>
</tr>
<tr>
<td></td>
<td>Deburr</td>
<td>S102</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Top Face</td>
<td>M501</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Top Face</td>
<td>M504</td>
</tr>
<tr>
<td></td>
<td>Bore 96 Dia Hole</td>
<td>B108, B112</td>
</tr>
<tr>
<td></td>
<td>Finish Bore 96 Dia Hole</td>
<td>B109</td>
</tr>
<tr>
<td></td>
<td>End Mill Base Edges</td>
<td>M401, M412</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Face C</td>
<td>M507</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Face c</td>
<td>M509</td>
</tr>
<tr>
<td></td>
<td>Center Hole</td>
<td>D201</td>
</tr>
<tr>
<td></td>
<td>Drill Dia 30 Hole Thru Wall</td>
<td>D130</td>
</tr>
</tbody>
</table>
### PROCESS DETAILS

**PART NO.: 7**  
**PART NAME:** PANEL SIDE COVER

<table>
<thead>
<tr>
<th>OPERATION NO.</th>
<th>DESCRIPTION OF THE OPERATION</th>
<th>TOOL(S) REQUIRED</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>Rough Mill Bottom Face (A)</td>
<td>M102,M401</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Bottom Face (A)</td>
<td>M103,M105</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Top Edges (B)</td>
<td>M404,M511</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Top Edges (B)</td>
<td>M402,M512</td>
</tr>
<tr>
<td></td>
<td>Center Holes (5)</td>
<td>D201</td>
</tr>
<tr>
<td></td>
<td>Drill 8 Dia Holes 25 Deep (5)</td>
<td>D108</td>
</tr>
<tr>
<td></td>
<td>Deburr (5)</td>
<td>S102</td>
</tr>
<tr>
<td></td>
<td>Tap 8 Dia Holes 25 Deep (5)</td>
<td>T108</td>
</tr>
<tr>
<td></td>
<td>Box Mill Inside Edges (Rough) (C)</td>
<td>M401,M407</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Inside Edges (C)</td>
<td>M402,M408</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Sides Of projection (D)</td>
<td>M404</td>
</tr>
<tr>
<td>(2)</td>
<td>Face Mill Boss (E)</td>
<td>M401,M408</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Boss (E)</td>
<td>M402</td>
</tr>
<tr>
<td></td>
<td>Center Hole (1)</td>
<td>D201</td>
</tr>
<tr>
<td></td>
<td>Drill 20 dia Hole Thro (1)</td>
<td>D120</td>
</tr>
<tr>
<td></td>
<td>Ream 20 Dia Hole Thro (1)</td>
<td>R120</td>
</tr>
<tr>
<td></td>
<td>Deburr (1)</td>
<td>S102</td>
</tr>
<tr>
<td></td>
<td>Side Mill ridge (Rough) (F)</td>
<td>M502</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Ridge (F)</td>
<td>M502,M701</td>
</tr>
<tr>
<td>(3)</td>
<td>Face Mill Surface (C)</td>
<td>M701,M704</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Surface (G)</td>
<td>M702</td>
</tr>
<tr>
<td></td>
<td>Center Holes (3)</td>
<td>D201</td>
</tr>
<tr>
<td></td>
<td>Drill 15 Dia Holes 10 Deep (3)</td>
<td>D115</td>
</tr>
<tr>
<td></td>
<td>Drill 15 Dia Hole 25 Deep (for notch H)</td>
<td>D115</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Notch (H)</td>
<td>M413</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Pocket (I)</td>
<td>M506</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Pocket (I)</td>
<td>M508</td>
</tr>
<tr>
<td></td>
<td>Center Hole (I)</td>
<td>D201</td>
</tr>
<tr>
<td></td>
<td>Drill 8 Dia Hole 10 Deep (1)</td>
<td>D108</td>
</tr>
<tr>
<td></td>
<td>Tap 8 dia Holes 10 Deep</td>
<td>T108</td>
</tr>
<tr>
<td>(4)</td>
<td>Step Mill Edge (Rough) (J)</td>
<td>M506</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Edge (J)</td>
<td>M508</td>
</tr>
<tr>
<td></td>
<td>Contour Mill to Finish (J)</td>
<td>M601,M402</td>
</tr>
<tr>
<td></td>
<td>Center Holes (2)</td>
<td>D202</td>
</tr>
<tr>
<td></td>
<td>Drill 5 Dia Hole 5 deep</td>
<td>D105</td>
</tr>
<tr>
<td></td>
<td>Deburr</td>
<td>S101</td>
</tr>
<tr>
<td>OPERATION NO.</td>
<td>DESCRIPTION OF THE OPERATION</td>
<td>TOOL(S) REQUIRED</td>
</tr>
<tr>
<td>--------------</td>
<td>------------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>(1)</td>
<td>Rough Mill Bottom Face (A)</td>
<td>M101, M108, M501</td>
</tr>
<tr>
<td></td>
<td>Rough Mill sides (C)</td>
<td>M401</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Bottom Face (A)</td>
<td>M102, M502</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Sides (C)</td>
<td>M401</td>
</tr>
<tr>
<td>(2)</td>
<td>Rough Mill Top Face (B)</td>
<td>M101</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Top face (B)</td>
<td>M102</td>
</tr>
<tr>
<td></td>
<td>Center Holes (2)</td>
<td>D201</td>
</tr>
<tr>
<td></td>
<td>Drill 25 Dia Hole Thru&quot;</td>
<td>D125</td>
</tr>
<tr>
<td></td>
<td>Drill 40 Dia Hole Thru&quot;</td>
<td>D140</td>
</tr>
<tr>
<td></td>
<td>Counter Bore</td>
<td>B105</td>
</tr>
<tr>
<td></td>
<td>Deburr</td>
<td>S102</td>
</tr>
<tr>
<td>(3)</td>
<td>Rough Mill Face (D)</td>
<td>M702, M301</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Face (D)</td>
<td>M703</td>
</tr>
<tr>
<td></td>
<td>Centering Holes (3)</td>
<td>D203</td>
</tr>
<tr>
<td></td>
<td>Drill Dia 25 Hole (3)</td>
<td>D125</td>
</tr>
<tr>
<td></td>
<td>Tap Dia 25 Hole (3)</td>
<td>T125</td>
</tr>
<tr>
<td></td>
<td>Rough Bore Thru&quot;</td>
<td>B108</td>
</tr>
<tr>
<td></td>
<td>Enlarge Bore (Rough)</td>
<td>B109</td>
</tr>
<tr>
<td></td>
<td>Chamfer</td>
<td>B110</td>
</tr>
<tr>
<td></td>
<td>Finish Bore Thru&quot;</td>
<td>B112</td>
</tr>
</tbody>
</table>
## PROCESS DETAILS

**PART NO.**: 9  
**PART NAME**: COVER

<table>
<thead>
<tr>
<th>OPERATION NO.</th>
<th>DESCRIPTION OF THE OPERATION</th>
<th>TOOL(S) REQUIRED</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) Rough Mill Side (A)</td>
<td>M102, M401</td>
<td></td>
</tr>
<tr>
<td>Finish Mill Side (A)</td>
<td>M103, M105, M402</td>
<td></td>
</tr>
<tr>
<td>Rough Mill Top Face</td>
<td>M101</td>
<td></td>
</tr>
<tr>
<td>Finish Mill Top Face</td>
<td>M101</td>
<td></td>
</tr>
<tr>
<td>Center Holes (9)</td>
<td>D203</td>
<td></td>
</tr>
<tr>
<td>Drill 15 Dia Holes (9)</td>
<td>D115</td>
<td></td>
</tr>
<tr>
<td>Center Holes (2)</td>
<td>D203</td>
<td></td>
</tr>
<tr>
<td>Drill Dia 30 Holes (2) Thru</td>
<td>D130</td>
<td></td>
</tr>
<tr>
<td>Debur</td>
<td>S101</td>
<td></td>
</tr>
<tr>
<td>Tap 15 Dia Holes (9)</td>
<td>T115</td>
<td></td>
</tr>
<tr>
<td>Circular Milling (Rough)</td>
<td>M501, M506</td>
<td></td>
</tr>
<tr>
<td>Circular Milling (Finish)</td>
<td>M502</td>
<td></td>
</tr>
<tr>
<td>Thro Bore 80 Dia (Rough)</td>
<td>B107</td>
<td></td>
</tr>
<tr>
<td>Thro Bore 80 Dia (Finish)</td>
<td>B109</td>
<td></td>
</tr>
<tr>
<td>(2) Counter Bore (Rough)</td>
<td>B112</td>
<td></td>
</tr>
<tr>
<td>Contour Mill (Finish)</td>
<td>M402</td>
<td></td>
</tr>
<tr>
<td>End Mill Pocket (Rough)</td>
<td>M501, M508</td>
<td></td>
</tr>
<tr>
<td>End Mill Pocket (Finish)</td>
<td>M502</td>
<td></td>
</tr>
<tr>
<td>Side Mill Pocket Edges</td>
<td>M301</td>
<td></td>
</tr>
<tr>
<td>Shape Milling (Rough)</td>
<td>M603</td>
<td></td>
</tr>
<tr>
<td>Shape Milling (Finish)</td>
<td>M604</td>
<td></td>
</tr>
<tr>
<td>(3) Mill Pocket (Small)</td>
<td>M404, M402</td>
<td></td>
</tr>
<tr>
<td>Finish Mill Pocket (Small)</td>
<td>M402</td>
<td></td>
</tr>
<tr>
<td>Finish Mill All Sides</td>
<td>M302</td>
<td></td>
</tr>
</tbody>
</table>
### PROCESS DETAILS

**PART NO.:** 10  
**PART NAME:** BOX

<table>
<thead>
<tr>
<th>OPERATION NO.</th>
<th>DESCRIPTION OF THE OPERATION</th>
<th>TOOL(S) REQUIRED</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>Rough Mill Bottom Face</td>
<td>M101</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Bottom Face</td>
<td>M103</td>
</tr>
<tr>
<td></td>
<td>Face Mill Side B (Rough)</td>
<td>M104, M105</td>
</tr>
<tr>
<td></td>
<td>Face Mill Side B (Finish)</td>
<td>M105</td>
</tr>
<tr>
<td></td>
<td>Face Mill Side C (Rough)</td>
<td>M104, M105</td>
</tr>
<tr>
<td></td>
<td>Face Mill Side C (Finish)</td>
<td>M105</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Top Face (D)</td>
<td>M408, M409</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Top Face (D)</td>
<td>M403, M409</td>
</tr>
<tr>
<td></td>
<td>Mill Grove (End Mill)</td>
<td>M415, M407</td>
</tr>
<tr>
<td></td>
<td>Center Holes (4)</td>
<td>D201</td>
</tr>
<tr>
<td></td>
<td>Drill 15 Dia Holes (4)</td>
<td>D115</td>
</tr>
<tr>
<td></td>
<td>Face Mill E (Rough)</td>
<td>M501, M509</td>
</tr>
<tr>
<td></td>
<td>Face Mill E (Finish)</td>
<td>M502</td>
</tr>
<tr>
<td></td>
<td>End Mill Edges (Rough)</td>
<td>M401</td>
</tr>
<tr>
<td></td>
<td>End Mill Edges (Finish)</td>
<td>M402</td>
</tr>
<tr>
<td></td>
<td>Contour Mill (Rough)</td>
<td>M501</td>
</tr>
<tr>
<td></td>
<td>Contour Mill (Finish)</td>
<td>M902</td>
</tr>
<tr>
<td></td>
<td>Face Mill Boss</td>
<td>M701</td>
</tr>
<tr>
<td></td>
<td>Center Hole</td>
<td>D202</td>
</tr>
<tr>
<td></td>
<td>Drill 15 Dia Hole 24 Deep</td>
<td>D115</td>
</tr>
<tr>
<td></td>
<td>Deburr</td>
<td>S101</td>
</tr>
<tr>
<td></td>
<td>Tap 15 Dia Hole 24 Deep</td>
<td>T115</td>
</tr>
<tr>
<td>OPERATION NO.</td>
<td>DESCRIPTION OF THE OPERATION</td>
<td>TOOL(S) REQUIRED</td>
</tr>
<tr>
<td>--------------</td>
<td>------------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>(1)</td>
<td>Rough Mill Face B</td>
<td>M112, M401</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Face B</td>
<td>M104, M412</td>
</tr>
<tr>
<td></td>
<td>Center Hole</td>
<td>D201</td>
</tr>
<tr>
<td></td>
<td>Drill 30 dia Hole</td>
<td>D130</td>
</tr>
<tr>
<td>(2)</td>
<td>Rough Mill Face A</td>
<td>M405</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Face A</td>
<td>M402, M410</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Face C</td>
<td>M405</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Face C</td>
<td>M402, M410</td>
</tr>
<tr>
<td></td>
<td>Bore Dia Hole Thru</td>
<td>B101</td>
</tr>
<tr>
<td></td>
<td>Finish Bore Dia Hole Thru</td>
<td>B109</td>
</tr>
<tr>
<td></td>
<td>Enlarge Dia Bore Deep</td>
<td>B105</td>
</tr>
<tr>
<td></td>
<td>Center Holes (4)</td>
<td>D202</td>
</tr>
<tr>
<td></td>
<td>Drill 16 Dia Holes 40 Deep</td>
<td>D116</td>
</tr>
<tr>
<td></td>
<td>Tap 16 Dia Holes 40 Deep</td>
<td>T116</td>
</tr>
<tr>
<td>(3)</td>
<td>Enlarge 80 Dia Bore 100 Deep</td>
<td>B109</td>
</tr>
<tr>
<td></td>
<td>Center Hole (4) (Face C)</td>
<td>D202</td>
</tr>
<tr>
<td></td>
<td>Drill 16 Dia Holes 40 Deep</td>
<td>D116</td>
</tr>
<tr>
<td></td>
<td>Tap 16 Dia Holes 40 Deep</td>
<td>T116</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Face D</td>
<td>M112, M401</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Face D</td>
<td>M104, M412</td>
</tr>
<tr>
<td></td>
<td>Enlarge 80 Dia Bore Thru to Centre</td>
<td>B109</td>
</tr>
</tbody>
</table>
PART #12  BRACKET-A

Scale 1:10
### PROCESS DETAILS

**PART NO.: 12**  
**PART NAME:** BRACKET-A

<table>
<thead>
<tr>
<th>OPERATION NO.</th>
<th>DESCRIPTION OF THE OPERATION</th>
<th>TOOL(S) REQUIRED</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>Rough Mill Bottom Edges A &amp; B</td>
<td>M112, M401</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Bottom Edges A &amp; B</td>
<td>M104, M105</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Cavity D</td>
<td>M101, M106</td>
</tr>
<tr>
<td></td>
<td>Semi-finish Cavity D</td>
<td>M104</td>
</tr>
<tr>
<td></td>
<td>Center Hole</td>
<td></td>
</tr>
<tr>
<td>(2)</td>
<td>Drill 30 Dia Hole Thru*</td>
<td>D130</td>
</tr>
<tr>
<td></td>
<td>Ream 30 Dia Hole Thru*</td>
<td>R130</td>
</tr>
<tr>
<td></td>
<td>Mill faces 'E &amp; F (Rough)</td>
<td>M401</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Feces E &amp; F</td>
<td>M402</td>
</tr>
<tr>
<td></td>
<td>Center Holes (2)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Drill 25 Dia Hole Thru* (2)</td>
<td>D125</td>
</tr>
<tr>
<td>(3)</td>
<td>Deburr</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Bore 30 Dia Holes Thru* (2)</td>
<td>S102</td>
</tr>
<tr>
<td></td>
<td>Finish Bore 30 Dia Holes (2)</td>
<td>B108</td>
</tr>
<tr>
<td></td>
<td>Face Mill Surface H</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Rough Mill Surface (H)</td>
<td>M505, M710</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Surface (G)</td>
<td>M107, M401</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Surface (I)</td>
<td>M702</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Surface (II)</td>
<td>M702, M704</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Surface (III)</td>
<td>M502</td>
</tr>
</tbody>
</table>
### PROCESS DETAILS

**PART NO.: 13**  
**PART NAME: JUNCTION COVER**

<table>
<thead>
<tr>
<th>OPERATION NO.</th>
<th>DESCRIPTION OF THE OPERATION</th>
<th>TOOL(S) REQUIRED</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>Rough Mill Face A</td>
<td>M701, M101</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Face A</td>
<td>M702, M102</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Around Periphery B</td>
<td>M401, M413</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Around Periphery B</td>
<td>M402, M406</td>
</tr>
<tr>
<td></td>
<td>Step Mill Around Periphery B</td>
<td>M301</td>
</tr>
<tr>
<td></td>
<td>Finish Step</td>
<td>M302</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Surface C</td>
<td>M301, M306</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Surface C</td>
<td>M302, M315</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Round Edges</td>
<td>M801</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Round Edges</td>
<td>M802</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Bosses</td>
<td>M701</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Bosses</td>
<td>M702</td>
</tr>
<tr>
<td></td>
<td>Center Holes (2)</td>
<td>D202</td>
</tr>
<tr>
<td></td>
<td>Drill 40 Dia Holes (2)</td>
<td>D140</td>
</tr>
<tr>
<td></td>
<td>Ream 40 Dia Holes (2)</td>
<td>R140</td>
</tr>
<tr>
<td></td>
<td>Tap 40 Dia Hole</td>
<td>T140</td>
</tr>
<tr>
<td></td>
<td>Center Holes (2)</td>
<td>D201</td>
</tr>
<tr>
<td></td>
<td>Drill 20 Dia Hole</td>
<td>D120</td>
</tr>
<tr>
<td></td>
<td>Ream 20 Dia Hole</td>
<td>R120</td>
</tr>
<tr>
<td></td>
<td>Tap 20 Dia Hole</td>
<td>T120</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Face E</td>
<td>M401</td>
</tr>
<tr>
<td></td>
<td>Face Mill Face E</td>
<td>M402</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Inside Bottom Surface</td>
<td>M405</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Inside Edges</td>
<td>M301</td>
</tr>
<tr>
<td></td>
<td>Face Mill F</td>
<td>M501</td>
</tr>
<tr>
<td></td>
<td>Finish Mill F</td>
<td>M502</td>
</tr>
<tr>
<td></td>
<td>Center Hole (1)</td>
<td>D201</td>
</tr>
<tr>
<td></td>
<td>Drill 30 Dia Hole Thro^ Wall</td>
<td>D130</td>
</tr>
<tr>
<td></td>
<td>Deburr</td>
<td>S102</td>
</tr>
<tr>
<td></td>
<td>Bore 56 Dia Hole Thro^ Wall</td>
<td>B108</td>
</tr>
<tr>
<td></td>
<td>Face Mill G</td>
<td>M501</td>
</tr>
<tr>
<td></td>
<td>Finish Mill G</td>
<td>M502</td>
</tr>
<tr>
<td></td>
<td>Center Hole (1)</td>
<td>D201</td>
</tr>
<tr>
<td></td>
<td>Drill 30 Dia Hole Thro^ Wall</td>
<td>D130</td>
</tr>
<tr>
<td></td>
<td>Deburr</td>
<td>S102</td>
</tr>
<tr>
<td></td>
<td>Bore 56 Dia Hole Thro^ Wall</td>
<td>D130</td>
</tr>
<tr>
<td></td>
<td>Deburr</td>
<td>S102</td>
</tr>
<tr>
<td></td>
<td>Bore 56 Dia Hole Thro^ Wall</td>
<td>B108</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Face H</td>
<td>M501, M506</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Face H</td>
<td>M502, M508</td>
</tr>
<tr>
<td>OPERATION NO.</td>
<td>DESCRIPTION OF THE OPERATION</td>
<td>TOOL(S) REQUIRED</td>
</tr>
<tr>
<td>--------------</td>
<td>------------------------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>(1)</td>
<td>Rough Mill Face A</td>
<td>M102, M401</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Face A</td>
<td>M103, M105</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Face B</td>
<td>M401, M508</td>
</tr>
<tr>
<td></td>
<td>Semi Finish Face B</td>
<td>M402, M509</td>
</tr>
<tr>
<td></td>
<td>Finish Face B</td>
<td>M402, M415</td>
</tr>
<tr>
<td></td>
<td>Center Holes (4)</td>
<td>D202</td>
</tr>
<tr>
<td></td>
<td>Drill 25 Dia Holes Thro' (4)</td>
<td>D125</td>
</tr>
<tr>
<td></td>
<td>Ream 25 Dia Holes Thro' (4)</td>
<td>R125</td>
</tr>
<tr>
<td></td>
<td>Bore 35 Dia Holes Thro' (4)</td>
<td>B102</td>
</tr>
<tr>
<td></td>
<td>Center Holes (2)</td>
<td>D201</td>
</tr>
<tr>
<td></td>
<td>Drill 25 Dia Holes Thro' (2)</td>
<td>D125</td>
</tr>
<tr>
<td></td>
<td>Deburr</td>
<td>S102</td>
</tr>
<tr>
<td></td>
<td>Ream 25 Dia Holes Thro' (2)</td>
<td>R125</td>
</tr>
<tr>
<td></td>
<td>Tap 25 Dia Hole</td>
<td>T125</td>
</tr>
<tr>
<td></td>
<td>Face Mill Surface (C)</td>
<td>M401</td>
</tr>
<tr>
<td>(2)</td>
<td>Finish Mill Surface (C)</td>
<td>M412</td>
</tr>
<tr>
<td></td>
<td>Face Mill Surface (D)</td>
<td>M101</td>
</tr>
<tr>
<td></td>
<td>Face Mill Surface (D)</td>
<td>M102</td>
</tr>
<tr>
<td></td>
<td>Rough Bore Dia Holes (2)</td>
<td>B102</td>
</tr>
<tr>
<td></td>
<td>Finish Bore Dia Holes (2)</td>
<td>B103</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Face B1</td>
<td>M102, M401</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Face B1</td>
<td>M103, M105</td>
</tr>
<tr>
<td>(3)</td>
<td>Face Mill C</td>
<td>M701</td>
</tr>
<tr>
<td></td>
<td>Finish Mill C</td>
<td>M702</td>
</tr>
<tr>
<td></td>
<td>Face Mill D</td>
<td>M701</td>
</tr>
<tr>
<td></td>
<td>Finish Mill D</td>
<td>M702</td>
</tr>
<tr>
<td></td>
<td>Mill Bottom Face of Recess E</td>
<td>M401, M415</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Bottom of Recess E</td>
<td>M402, M705</td>
</tr>
<tr>
<td>(4)</td>
<td>Rough Mill Walls of Recess E</td>
<td>M103</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Walls Of Recess E</td>
<td>M102</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Recess Projection H</td>
<td>M405</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Recess Projection H</td>
<td>M406</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Contour F</td>
<td>M601</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Contour F</td>
<td>M602</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Contour G</td>
<td>M603</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Contour G</td>
<td>M604</td>
</tr>
</tbody>
</table>
PART #15

CASING

Scale 1:10
# PROCESS DETAILS

**PART NO.**: 15  
**PART NAME**: CASING

<table>
<thead>
<tr>
<th>OPERATION NO.</th>
<th>DESCRIPTION OF THE OPERATION</th>
<th>TOOL(S) REQUIRED</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>Rough Mill Surface A</td>
<td>M105, M708</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Surface A</td>
<td>M102, M702</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Sides of Box</td>
<td>M108, M302</td>
</tr>
<tr>
<td></td>
<td>End Mill Edges B</td>
<td>M401, M412</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Edges B</td>
<td>M402</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Boss C</td>
<td>M401</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Boss C</td>
<td>M408</td>
</tr>
<tr>
<td></td>
<td>Center Hole</td>
<td>D201</td>
</tr>
<tr>
<td></td>
<td>Drill 50 Dia Hole Thro&quot;</td>
<td>D150</td>
</tr>
<tr>
<td></td>
<td>Bore 60 Dia Hole Thro&quot;</td>
<td>B108</td>
</tr>
<tr>
<td></td>
<td>Finish Bore 60 Dia Hole Thro&quot;</td>
<td>B109</td>
</tr>
<tr>
<td></td>
<td>Rough Mill Bottom D</td>
<td>M412</td>
</tr>
<tr>
<td></td>
<td>End Mill Oblong Hole (Rough)</td>
<td>M416, M508</td>
</tr>
<tr>
<td></td>
<td>Finish Mill Oblong Hole</td>
<td>M417</td>
</tr>
<tr>
<td></td>
<td>Finish Face (E)</td>
<td>M402</td>
</tr>
<tr>
<td></td>
<td>Center Hole (1)</td>
<td>D202</td>
</tr>
<tr>
<td>(3)</td>
<td>Drill 45 Dia Hole Thro&quot; to Centre</td>
<td>D145</td>
</tr>
<tr>
<td></td>
<td>Bore 50 Dia Hole Thro&quot; to Centre</td>
<td>B112</td>
</tr>
<tr>
<td></td>
<td>Finish Bore Hole Thro&quot; to Centre</td>
<td>B118</td>
</tr>
</tbody>
</table>
APPENDIX C

COMPUTER PROGRAM LISTINGS
CELL FORMATION IN FMS - PART FAMILY FORMATION PROBLEM

AUTHOR - GAJANANA NADOLI
GRADUATE STUDENT, DEPT. OF INDUSTRIAL ENGINEERING
UNIVERSITY OF WINDSOR, WINDSOR, ONTARIO N9B 2Z2

THIS PROGRAM GENERATES THE INPUT FILE OF THE PROBLEM P(R)
FOR THE SAS/OR (VERSION 5) INTEGER PROGRAMMING ROUTINE

THE VARIABLE DECLARATION SECTION *********************
INTEGER NUMOP(40), SIM(40,40), DISIN(40,40)
INTEGER ONP, OPJ, A
INTEGER XM(2500), XIND, XVars, MVars, XIND, XINDJ
INTEGER RH, XX(2500), KH(2500)
INTEGER NK(40,3), NEH(40,3), NNN(3)
REAL LMDA, M(2500), COEFF(40,40)

THE DATA INPUT AND CALCULATION OF SIMILAR AND DISSIMILAR PROCESSES
BETWEEN THE PARTS ********************
READ(5, 10) N, K
READ, LAMDA

FORMAT(I2, 1X, I2)
DO 100 I = 1, N
   READ(5, 20) NUM, (OP(I, J), J = 1, NUM)
   NUMOP(I) = NUM

20 FORMAT(I2, 10(1X, A4), /(3X, 10(A4, 1X)))
100 CONTINUE

NM1 = N - 1
DO 200 I = 1, NM1
   IT = I + 1
   DO 300 J = IT, N
      SIM(I, J) = 0
      NTERM = NUMOP(I)
      NTERMJ = NUMOP(J)
      DO 400 OPJ = 1, NTERM
         DO 500 OPJ = 1, NTERMJ
            IF (OP(I, OPI) .EQ. OP(J, OPJ)) SIM(I, J) = SIM(I, J) + 1

500 CONTINUE
400 CONTINUE

CONTINUE

DISIM(I, J) = NUMOP(I) + NUMOP(J) - 2 * SIM(I, J)
COEFF(I, J) = DISIM(I, J) - LMDA * SIM(I, J)

CONTINUE 200 CONTINUE

CREATING THE SAS FORMAT INPUT DATA FOR PART FAMILY FORMATION PROBLEM

METH = 3
CALL TINIT
NBLCK = 0
EXECUTE ALLCLR
EXECUTE TIMEK
EXECUTE OBJROW
EXECUTE TIMEL
EXECUTE ALLOC
EXECUTE TIMEL
IF (METH.EQ.1) THEN
  EXECUTE CNSTR
  EXECUTE MLLM
  EXECUTE INTGER
  EXECUTE UPPER
ENDIF
IF (METH.EQ.2) THEN
  EXECUTE CNSTR2
  EXECUTE MLIN2
  EXECUTE INTGER
  EXECUTE UPPER
ENDIF
IF (METH.EQ.3) THEN
  EXECUTE CNSTR2
  EXECUTE TIMEL
  EXECUTE MLIN
  EXECUTE TIMEL
  EXECUTE INTGER
  EXECUTE TIMEL
  EXECUTE UPPER
  EXECUTE TIMEL
ENDIF
IF (METH.EQ.4) THEN
  EXECUTE CNSTR
  EXECUTE MLIN2
  EXECUTE INTGER
  EXECUTE UPPER
ENDIF
IF (INSOL.EQ.1) EXECUTE BASICS
GO TO 9999
REMOTE BLOCK TIMEL
CALL TUSED(MSEC).
NBLOCK=NBLOCK+1
PRINT "TIME USED FOR BLOCK",NBLOCK," IS ",MSEC
CALL TINIT
ENDBLOCK
C REMOTE BLOCK ALLCLR
REMOTE BLOCK ALLCLR
DU 3030 II=1,N
DO 3030 IIFAM=1,K
  XIND= (II-1)*K + IIFAM
  X(XIND)=0
3030 CONTINUE
NMI=N-1
DO 3040 II=1,NM1
  III=II+1
  DO 3050 JJ=III,N
    DU 3060 IIFAM=1,K
    LI=II
    LJ=JJ
    LIFAM=IIFAM
    EXECUTE MIJKA
    M(MINDEX)=0
3060 CONTINUE
3050 CONTINUE
3040 CONTINUE
C REMOTE BLOCK MIJKA
REMOTE BLOCK MIJKA
MINDA=0
LI1=LI1+1
ISIGN =LI1-1
IF (ISIGN.EQ.0) GO TO 3064
DO 3065 IX = 1,ISIGN
  MINDA=MINDA + (N-IX) * K
3065 CONTINUE
3064 MINDEX=MINDA+ (LJ-LI1)*K +LIFAM
ENDBLOCK

C REMOTE BLOCK OBJROW
REMOTE BLOCK OBJROW
NMI= N-1
IL=IL+1
DO 3110 J=IL,N
  DO 3120 IFAM=1,K
    IL=I
    LJ=J
    LIFAM=IFAM
    EXECUTE MIJKA
    M(MINDEX) = COEFF(I,J)
3120 CONTINUE
3110 CONTINUE
3100 CONTINUE
CTYPE="MIN"
FLAG="OBJROW"
EXECUTE PUTROW
ENDBLOCK
C REMOTE BLOCK PUTROW
REMOTE BLOCK PUTROW
XVARS=N*K
NVARS=N*(N-1)/2*K
IF (FLAG.EQ."OBJROW") THEN
  WRITE(8,4000) (X(I),I=1,XVARS)
4003 FORMAT(15("",I4))
  WRITE(8,4002) (M(J),J=1,NVARS)
4002 FORMAT(8("",F9.2))
ELSE
  DO 7856 II=1,XVARS
    KK(II) = X(II)
  7856 CONTINUE
  DO 7857 JJ=1,NVARS
    KM(JJ)=M(JJ)
  7857 CONTINUE
  WRITE(8,4112) (KK(II),II=1,XVARS)
  WRITE(8,4113) (KM(JJ),JJ=1,NVARS)
4112 FORMAT(15("",I4))
4113 FORMAT(15("",I4))
ENDIF
IF (FLAG.EQ."OBJROW") THEN
  WRITE(8,4010) CTYPE
4010 FORMAT("",A8,"",")
ELSEIF (FLAG.EQ."ALOC") THEN
  WRITE(8,4020) CTYPE,RHS
4020 FORMAT("",A8,IX,15)
ELSEIF (FLAG.EQ."CNSTR") THEN
  WRITE(8,4020) CTYPE,RHS
ELSEIF (FLAG.EQ."MLIM") THEN
  WRITE(8,4020) CTYPE,RHS
ELSEIF (FLAG.EQ."INTGER") THEN
  WRITE(8,4010) CTYPE
ELSEIF (FLAG.EQ."UPPER") THEN
  WRITE(8,4010) CTYPE
ENDIF
EXECUTE ALLCLR
ENDBLOCK
C REMOTE BLOCK ALLOC
REMOTE BLOCK ALLOC
FLAG="ALLOC"
CTYPE="EQ"
RHS=1
DO 3200 I=1,N
  DO 3210 IFAM=1,K
    XIND= (I-1)*K+ IFAM
    X(XIND)=1
  3210 CONTINUE
EXECUTE PUTROW
3200 CONTINUE
ENDBLOCK
C REMOTE BLOCK CNSTR
REMOTE BLOCK CNSTR
FLAG="CNSTR"
CTYPE="LE"
RHS=1
NH1=N-1
DO 3230 I=1,NH1
  IL=I+1
  DO 3240 J=I1,N
    DO 3250 IFAM=1,K
      LJ=J
      LIFAM=IFAM
      EXECUTE HJICA
      N(HINDEX) = -1
      XIND= (I-1)*K+ IFAM
      XINDJ= (J-1)*K+ IFAM
      X(XINDI)=1
      X(XINDJ)=1
      EXECUTE PUTROW
    3250 CONTINUE
  3240 CONTINUE
3230 CONTINUE
ENDBLOCK
C REMOTE BLOCK CNSTR2
REMOTE BLOCK CNSTR2
FLAG="CNSTR"
CTYPE="LE"
NH1=N-1
DO 3400 I=1,NH1
  DO 3410 IFAM=1,K
    XIND=(I-1)*K+ IFAM
    RHS=N-1
    X(XIND)=N-1
    IL=I+1
  3410 CONTINUE
3420 CONTINUE
LJ=I
LJ=J
LIFAM=IFAM
EXECUTE MIJKA
H(MINDEX)=1
XINDJ=(J-1)*K+IFAM
X(XINDJ)=1
3420   CONTINUE
EXECUTE PUTROW
3410   CONTINUE
3400   CONTINUE
ENDBLOCK
C REMOTE BLOCK MLIM
REMOTE BLOCK MLIM
FLAG="MLIM"
CTYPE="LE"
RHS=0
NL=M-1
DO 3260 I=1,N
   IL=I+1
   DO 3270 J=IL,N
      DO 3280 IFAM=1,K
         LI=I
         LJ=J
         LIFAM=IFAM
         EXECUTE MIJKA
         H(MINDEX)=1
         XINDI=(I-1)*K+IFAM
         X(XINDI)=1
         EXECUTE PUTROW
         LI=I
         LJ=J
         LIFAM=IFAM
         EXECUTE MIJKA
         XINDJ=(J-1)*K+IFAM
         H(MINDEX)=1
         X(XINDJ)=1
         EXECUTE PUTROW
3280   CONTINUE
3270   CONTINUE
3260   CONTINUE
ENDBLOCK
C REMOTE BLOCK MLIM2
REMOTE BLOCK MLIM2
FLAG="MLIM2"
CTYPE="LE"
RHS=0
DO 3500 I=1,N
   DO 3510 IFAM=1,K
      XIND=(I-1)*K+IFAM
      X(XIND)=1-N
   DO 3520 J=1,N
      IF (J.EQ.1) GO TO 3520
      LI=I
      LJ=J
      LIFAM=IFAM
      EXECUTE MIJKA
      H(MINDEX)=1
3520   CONTINUE
EXECUTE PUTROW
3510   CONTINUE
3500   CONTINUE
END BLOCK
C REMOTE BLOCK INTEGER
REMOVED BLOCK INTEGER
FLAG="INTEGER"
CTYPE="INTEGER"
DO 3300 I=1,N
  K11=K-1
  DO 3310 IFAM=1,K11
    XIND= (I-1)*K+IFAM
    X(XIND)=1
  3310 CONTINUE
  3300 CONTINUE
EXECUTE PUTROW
END BLOCK
C REMOTE BLOCK UPPER
REMOVED BLOCK UPPER
FLAG="UPPER"
CTYPE="UPPERBD"
DO 3700 I=1,N
  DO 3710 IFAM=1,K
    XIND=(I-1)*K+IFAM
    X(XIND)=1
  3710 CONTINUE
  3700 CONTINUE
NM1=N-1
DO 3730 I=1,NM1
  J=I+1
  DO 3740 J=1,J,N
    DO 3750 IFAM=1,K
      LI=I
      LJ=J
      LIFAM=IFAM
      EXECUTE MIJKA
      H(HINDEX)=1
  3750 CONTINUE
  3740 CONTINUE
  3730 CONTINUE
EXECUTE PUTROW
END BLOCK
9999 STOP
END
ENTRY
03
1.4502
26 N501 M502 M701 M601 M101 M401 M102 N103 M301 M702
M603 M108 M503 M504 D201 D142 D109 D202 D130 R142
R130 R148 B108 B109 B101 B115
1B N101 M103 M104 M105 M901 M602 M501 M506 M508 M402
D201 D125 R128 S104 B101 B109 B105 B104
2C M403 M404 M701 M401 M412 M405 M406 M702 M712 N101
M102 D202 D129 D203 T130 B108 B109 B101 B102
B115 B106 B112
23 N501 M514 M502 M710 N712 N211 M101 N102 M212 M503
N101 M302 M401 M405 D202 D130 D201 D120 D202 B108
B112 S109 T120
29 N102 M401 M103 M105 M602 M603 M501 M513 M502 M503
M101 M701 M702 M402 M508 M406 M610 M301 M305 D201
18 M501 M504 M502 M415 M412 M418 M901 M902 M401 M507
M509 D201 D118 D130 B108 B112 B109 S102
<table>
<thead>
<tr>
<th>28</th>
<th>M102</th>
<th>M401</th>
<th>M103</th>
<th>M105</th>
<th>M404</th>
<th>M502</th>
<th>M702</th>
<th>M402</th>
<th>M512</th>
<th>M407</th>
<th>M408</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>M502</td>
<td>H701</td>
<td>M704</td>
<td>M702</td>
<td>M413</td>
<td>M506</td>
<td>M508</td>
<td>M601</td>
<td>D201</td>
<td>D108</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D120</td>
<td>D115</td>
<td>D202</td>
<td>D105</td>
<td>S102</td>
<td>S101</td>
<td>T108</td>
<td>R120</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>M101</td>
<td>H108</td>
<td>M501</td>
<td>M401</td>
<td>M102</td>
<td>M502</td>
<td>M702</td>
<td>M301</td>
<td>M703</td>
<td>D201</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D125</td>
<td>D140</td>
<td>D203</td>
<td>B105</td>
<td>B108</td>
<td>B109</td>
<td>B112</td>
<td>B110</td>
<td>S102</td>
<td>T125</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>M102</td>
<td>H401</td>
<td>M103</td>
<td>M105</td>
<td>M402</td>
<td>M101</td>
<td>M106</td>
<td>M501</td>
<td>M506</td>
<td>M502</td>
<td></td>
</tr>
<tr>
<td></td>
<td>M508</td>
<td>M301</td>
<td>M603</td>
<td>M604</td>
<td>M404</td>
<td>M302</td>
<td>D203</td>
<td>D115</td>
<td>D130</td>
<td>S101</td>
<td></td>
</tr>
<tr>
<td></td>
<td>T115</td>
<td>B107</td>
<td>B109</td>
<td>B112</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>M101</td>
<td>M103</td>
<td>M104</td>
<td>M503</td>
<td>M105</td>
<td>M408</td>
<td>M106</td>
<td>M403</td>
<td>M415</td>
<td>M407</td>
<td></td>
</tr>
<tr>
<td></td>
<td>M501</td>
<td>M509</td>
<td>M703</td>
<td>M502</td>
<td>M401</td>
<td>M402</td>
<td>M901</td>
<td>M902</td>
<td>M701</td>
<td>D201</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>M112</td>
<td>M401</td>
<td>M104</td>
<td>M412</td>
<td>M405</td>
<td>M402</td>
<td>M410</td>
<td>D201</td>
<td>D130</td>
<td>D202</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D116</td>
<td>B101</td>
<td>B109</td>
<td>B105</td>
<td>T116</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>M112</td>
<td>M401</td>
<td>M104</td>
<td>M105</td>
<td>M101</td>
<td>M106</td>
<td>M402</td>
<td>M505</td>
<td>M710</td>
<td>M107</td>
<td></td>
</tr>
<tr>
<td></td>
<td>M702</td>
<td>M704</td>
<td>M502</td>
<td>D201</td>
<td>D130</td>
<td>D202</td>
<td>D125</td>
<td>B108</td>
<td>B109</td>
<td>R130</td>
<td>S102</td>
</tr>
<tr>
<td></td>
<td>S102</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

SIBSYS
$STOP

//
// PB JOB (R240, Nu7, 10, 5), "GAJ", CLASS=A, REGION=2048K
// EXEC WATFIV,
// FTO8FO01 DD DSM=YL.R240NU7.FAMOUT,UNIT=DASD, VOL=SER=WORKPK,
// DISP=(NEW,KEEP), SPACE=(TRK,(40,10)),
// DCB=(RECNO=80, BLKSIZE=15440, RECFM=FB)
// GO.SYSIN DD *

$JOB WATFIV REF. SECTION 4.2.2
C ---------------------------------------------------------
C CELL FORMATION IN FMS - PART FAMILY FORMATION PROBLEM
C ---------------------------------------------------------
C AUTHOR - GAJANANA NADOLI
C GRADUATE STUDENT, DEPT. OF INDUSTRIAL ENGINEERING
UNIVERSITY OF WINDSOR, WINDSOR, ONTARIO N9B 2ZZ
C ---------------------------------------------------------
C THIS PROGRAM IMPLEMENTS THE ALGORITHM FOR FINDING THE
C LOWER BOUND AND UPPERBOUND FOR FUNCTION P(R) WITH
C DIFFERENT VALUES OF R TO ESTABLISH THE REGION (L, U).
C ---------------------------------------------------------
C THE VARIABLE DECLARATION SECTION ***********************
INTEGER NUMOP(40), SIM(40, 40), DISIM(40, 40)
INTEGER OPI, OPJ, A
INTEGER X(500), XIND, XVAR, MVAR, XINDI, XINDJ
INTEGER RHS
REAL LANDA, M(2500), COEFF(40, 40), S(435), LBOUND
CHARACTER CTYPE*, FLG*8, OP*4(40, 80)
COMMON OP, NUMOP, SIM, DISIM, OPI, OPJ, COEFF, I, J, IFAM, K,
L, NN, NP, MN, N, K1
C THE DATA INPUT AND CALCULATION OF SIMILAR AND DISSIMILAR PROCESSES
C BETWEEN THE PARTS ***********************
READ(5, 10) N, K
10 FORMAT(I2, 1X, I2)
DO 100 I = 1, N
PRINT, "PART NUMBER = ", I
READ(5, 20) NUM, (OP(I, J), J = 1, NUM).
NUMOP(I) = NUM
20 FORMAT(I2, 10(I4, A4), /, (3X, 10(A4, 1X)))
WRITE(8, 20) NUMOP(I), (OP(I, J), J = 1, NUM)
100 CONTINUE
NM1 = N - 1
DO 200 I = 1, NM1
Ii = I + 1
DO 300 J = I, N
SIM(I, J) = 0
NTERM1 = NUMOP(I)
NTERMJ = NUMOP(J)
DO 400 OPJ = 1, NTERMJ
DO 500 OP1 = 1, NTERM1
IF (OP(I, OP1) .EQ. OP(J, OPJ)) SIM(I, J) = SIM(I, J) + 1
500 CONTINUE
400 CONTINUE
DISIM(I, J) = NUMOP(I) + NUMOP(J) - 2 * SIM(I, J)
300 CONTINUE
200 CONTINUE
WRITE(8, 4018)
4018 FORMAT(" BOUNDS ON THE OBJECTIVE FUNCTION FOR DIFFERENT VALUES
OF R ")
WRITE(4, 4053)
WRITE(4, 4024)
4024 FORMAT(" ")
WRITE(8, 4024)
WRITE(8,4026) N
WRITE(8,4027) K
WRITE(8,4024) N*(N-1)/2*K
WRITE(8,4053) K
WRITE(8,4024) N
WRITE(8,4024) N
WRITE(8,5000)
WRITE(8,9001)
WRITE(8,9002)
FORMAT("" # OF NEGATIVE OBJECTIVE FUNCTION "")
FORMAT("" COEFFICIENTS LOWERBOUND UPPERBOUND"" )
EXECUTE ALLCLR
DO 18 LLANDA=5.1000.5
   LANDA=LLANDA/100.00
   NH1=N-1
   DO 21 I=1,NH1
      I1=I+1
   21 CONTINUE
   DO 22 J=I1,N
      COEFF(I,J)=DISIM(I,J)-LANDA*SI(M(I,J))
22 CONTINUE
DO 89 JN=1,435
   S(JN)=0.0
89 CONTINUE
EXECUTE 08JROW
GO TO 9999
C REMOTE BLOCK ALLCLR
REMOTE BLOCK ALLCLR
DO 3030 I1=1,N
   DO 3030 I1FAM=1,K
      XIND=(I1-1)*K+I1FAM
      X(XIND)=0
   3030 CONTINUE
   NH1=N-1
   DO 3040 I1=1,NH1
      I1I=I1+1
   3040 CONTINUE
   DO 3050 JJ=I1I,N
      DO 3050 JJFAM=1,K
         L1=I1I
         L1J=JJ
         LIFAM=I1FAM
         EXECUTE MIJKA
         N(NINDEX)=0
      3050 CONTINUE
   3050 CONTINUE
   CONTINUE
END BLOCK
C REMOTE BLOCK MIJKA
REMOTE BLOCK MIJKA
MIIND=0
L1L=L1+1
ISIGN =L1-1
IF (ISIGN.EQ.0) GO TO 3064
DO 3063 IX = 1,ISIGN
MINDA = MINDA + (N-IX) * K

CONTINUE

MINDEX = MINDA + (LJ-LII)*K + LIFAM

ENDBLOCK.

C REMOTE BLOCK OBJROW

REMOTE BLOCK OBJROW

NH1 = N-1
NEG = 0
MS = 1

DO 3100 I=1,NH1
     II = I+1
     DO 3110 J=II,N
          DO 3120 IFAM = 1,K
               LI = I
               LJ = J
               LIFAM = IIFAM
               EXECUTE MIJKA
               M(MINDEX) = COEFF(I,J)
               IF(COEFF(I,J).LT.0) NEG = NEG + 1
               IF(IFAM.EQ.1) THEN
                   S(MS) = COEFF(I,J)
                   MS = MS + 1
               ENDIF
          3120    CONTINUE
     3110    CONTINUE
   3100    CONTINUE

CTYPE = "MIN"
FLAG = "OBJROW"
IES = N/K
IPAIRS = IES*(IES-1)/2*K + (N-IES*K)*IES
MPAIRS = N*(N-1)/2
LBOUND = 0.000
UBOUND = 0.000
ICOUNT = 0

DO 889 KI = 1,435
     INTS = S(KI)*10000.000
     IF (INTS.NE.0) THEN
         ICOUNT = ICOUNT + 1
     ELSE
         GO TO 889
     ENDIF
     IF (S(KI).LT.0.00) THEN
         LBOUND = LBOUND + S(KI)
     ENDIF
     IF (S(KI).GT.0.00) THEN
         IF (ICOUNT.LE.IPAIRS) LBOUND = LBOUND + S(KI)
     ENDIF
  889    CONTINUE

ICOUNT = 0

DO 899 KI = 1,435
     INTS = S(435-KI+1)*10000.000
     IF (INTS.NE.0) THEN
         ICOUNT = ICOUNT + 1
     ELSE
         GO TO 899
     ENDIF
     IF (S(435-KI+1).GT.0.00) THEN
         UBOUND = UBOUND + S(435-KI+1)
     ENDIF
     IF (S(-KJ-KI+1).LT.0.00) THEN


IF(ICOUNT.LE.IPAIRS) UBOUND=UBOUND+S(435-KI+1)
ENDIF
899 CONTINUE
EXECUTE PUTROW
ENDBLOCK
C REMOTE BLOCK PUTROW
REMOTE BLOCK PUTROW
XVARS=N*K
MVARS=N*(N-1)/2*K
EXECUTE ALLCLR
ENDBLOCK
9999 STOP
END
ENTRY
THE INPUT DATA IS THE SAME AS THE ONE LISTED FOR
THE PROGRAM 1.

$IBSYS
$STOP
//
C CELL FORMATION IN FMS — PART FAMILY FORMATION PROBLEM

C A U T H O R — G A J A N A N A N A D O L I


C T H E V A R I A B L E D E C L A R A T I O N S E C T I O N

***

INTEGER NUMOP(30), SIM(30,30), DISIM(30,30)
INTEGER OP1, OPJ, A
INTEGER XIND, XVALS, XINDI, XINDJ
INTEGER RHS
INTEGER MK(30,3), MEH(30,3), NNN(3), FAMN(3), FAMD(3)
INTEGER FSIM(30,3), FDISIM(30,3), NSIM(30,30)
INTEGER NDISIM(30,30)
REAL LAMDA, OP*4(30,50), COEFF(30,30), PFCOEFF(30,3), M(500), X(500)
REAL FAMC(3), HINC
CHARACTER CTYP#8, FLAG#8
COMMON OP, NUMOP, SIM, DISIM, OPI, OPJ, COEFF, I, J, IFAN, K,
L, NM, NP, MN, N, NL


C B E T W E E N T H E P A R T S

***

READ(5,10) N, K
10 FORMAT(I2,1X,I2)
DO 100 I=1,N
READ(5,20) NUM,(OP(I,J),J=1,NUM)
NUMOP(I)=NUM
20 FORMAT(I2,10(I1X,A4),/,(3X,10(A4,1X)))
WRITE(6,20) NUMOP(I),(OP(I,J),J=1,NUM)
100 CONTINUE

NML=N-1
DO 200 I=1,NML
I=I+1
DO 300 J=I,N
SIN(I,J)=0
NTERMI=NUMOP(I)
NTERMJ=NUMOP(J)
DO 400 OPJ=1,NTERMJ
DO 500 OPJ=1,NTERMI
IF (OP(I,OPJ).EQ.OP(J,OPJ)) SIM(I,J)=SIM(I,J)+1
500 CONTINUE
400 CONTINUE
DISIM(I,J)=NUMOP(I)+NUMOP(J)-2*SIM(I,J)
300 CONTINUE
200 CONTINUE

C R E A D I N G T H E I N I T I A L C O N F I G U R A T I O N
LAMDA = 2.4562420
DO 2010 IFAM = 1, K
   DO 2010 I = 1, N
      MK(I, IFAM) = 0
      MEM(I, IFAM) = 0
   CONTINUE
   DO 2020 IFAM = 1, K
      READ(5, 2030) NUMF, (MK(I, IFAM), I = 1, NUMF)
      WRITE(6, 2036) NUMF, (MK(I, IFAM), I = 1, NUMF)
   CONTINUE
2036 FORMAT("INITIAL CONFIGURATION", I2, 10(1X, I2))
   NNN(IFAM) = NUMF
2030 FORMAT(I2, 20(1X, I2))
   CONTINUE
   DO 2050 IFAM = 1, K
      DO 2060 I = 1, N
         LU = NNN(IFAM)
         DO 2070 LL = 1, LU
            IF (MK(LL, IFAM) .EQ. I) MEM(I, IFAM) = 1
         CONTINUE
      CONTINUE
2070 CONTINUE
2060 CONTINUE
2050 CONTINUE
TFAMD = 0.0
   DO 2200 IFAM = 1, K
      N1 = N - 1
      FANN(IFAM) = 0
      FAMD(IFAM) = 0
      DO 2240 I = 1, N1
         II = I + 1
         DO 2240 J = 1, N
            FANN(IFAM) = FANN(IFAM) + MEM(I, IFAM) * MEM(J, IFAM) * DISIM(I, J)
            FAMD(IFAM) = FAMD(IFAM) + MEM(I, IFAM) * MEM(J, IFAM) * SIM(I, J)
         CONTINUE
      CONTINUE
2240 TFAMD = TFAMD + FAMD(IFAM)
2200 CONTINUE
   DO 188 IFAM = 1, K
      FANC(IFAM) = FANN(IFAM) / TFAMD
   CONTINUE
   MINC = 0
   DO 2260 IFAM = 1, K
      IF (FANC(IFAM) .GT. MINC) THEN
         NF = IFAM
         MINC = FANC(IFAM)
      ENDIF
2260 CONTINUE
C IMPOSING THE NF ************
   MF = 2
   NEWM = MNN(MF)
   NEWM1 = NEWM - 1
   DO 2300 NI = 1, NEWM1
      NEWN = NI + 1
      DO 2400 NJ = NI, NEWN
         I = MK(NI, MF)
         J = MK(NJ, MF)
         IF (I .LT. J) THEN
            NSIM(NI, NJ) = SIM(I, J)
            NDISIM(NI, NJ) = DISIM(I, J)
         ENDIF
         IF (I .GT. J) THEN
            NSIM(NI, NJ) = SIM(J, I)
            NDISIM(NI, NJ) = DISIM(J, I)
   ENDIF
2300 CONTINUE
2400 CONTINUE
}
ENDIF
CONTINUE
DO 2401 I=1,N
  DO 2410 IFAM=1,K
    FSIM(I,IFAM)=0
    FDISIM(I,IFAM)=0
  CONTINUE
2401
CONTINUE
DO 2430 NI=1,NEWN
  NPART=MK(NI,MF)
  DO 2440 IFAM=1,K
    IF (IFAM.EQ.MF) GO TO 2440
    DO 2460 J=1,N
      IF (J.EQ.NPART) GO TO 2460
      IF (J.GT.NPART) THEN
      FSIM(NI,IFAM)=FSIM(NI,IFAM)+SIM(NPART,J)*MEM(J,IFAM)
      FDISIM(NI,IFAM)=FDISIM(NI,IFAM)+DISIM(NPART,J)*MEM(J,IFAM)
      ELSE
      FSIM(NI,IFAM)=FSIM(NI,IFAM)+SIM(J,NPART)*MEM(J,IFAM)
      FDISIM(NI,IFAM)=FDISIM(NI,IFAM)+DISIM(J,NPART)*MEM(J,IFAM)
      ENDIF
2460 
  CONTINUE
2430
CONTINUE
NI=NI-1
DO 2462 I=1,N
  II=I+1
  DO 2470 J=1,NI
    SIM(I,J)=0
    DISIM(I,J)=0
2470
CONTINUE
2462
CONTINUE
NEWM=NEWN-1
DO 2480 NI=1,NEWM
  NI=NI+1
  DO 2490 NJ=NI,NEWN
    SIM(NI,NJ)=NSIM(NI,NJ)
    DISIM(NI,NJ)=NDISIM(NI,NJ)
    COEFF(NI,NJ)=DISIM(NI,NJ)-LANDA*SIM(NI,NJ)
2490
CONTINUE
2480
CONTINUE
DO 2550 I=1,NEWN
  IFAM=1,K
  FCoeff(I,IFAM)=FDISIM(I,IFAM)-LANDA*FSIM(I,IFAM)
2550 
CONTINUE
N=NEWN
C=0.0
DO 58 IFAM=1,K
  IF (IFAM.EQ.NF) GO TO 58
  C=C+FAH(N,IFAM)-LANDA*FAMD(IFAM)
PRINT "THE CONSTANT FACTOR =", C
58
CONTINUE
PRINT "THE CONSTANT FACTOR =", C
DO 45 I=1,N
  IFAM=1,K
  IF (IFAM.EQ.NF) GO TO 46
  PRINT "FSIM(" , I, IFAM, ") =" , FSIM(I,IFAM)
  PRINT "FDISIM(" , I, IFAM, ") =" , FDISIM(I,IFAM)
CONTINUE

EXECUTE ALLCLR
EXECUTE OBJROW
EXECUTE ALLOC
IF (METH.EQ.1) THEN
  EXECUTE CNSTR
  EXECUTE MLIM
  EXECUTE INTEGER
  EXECUTE UPPER
ENDIF
IF (METH.EQ.2) THEN
  EXECUTE CNSTR2
  EXECUTE MLIM2
  EXECUTE INTEGER
  EXECUTE UPPER
ENDIF
IF (METH.EQ.3) THEN
  EXECUTE CNSTR2
  EXECUTE MLIM
  EXECUTE INTEGER
  EXECUTE UPPER
ENDIF
IF (METH.EQ.4) THEN
  EXECUTE CNSTR
  EXECUTE MLIM2
  EXECUTE INTEGER
  EXECUTE UPPER
ENDIF
GO TO 9999

REMOTE BLOCK ALLCLR
REMOTE BLOCK ALLCLR
DO 3030 II=1,N
DO 3030 IIFAM=1,K
  XIND = (II-1)*K + IIFAM
  X(XIND) = 0
3030 CONTINUE

NM1 = N-1
DO 3040 II=1,NM1
  JJ = II+1
  DO 3060 IIFAM=1,K
    LI = II
    LJ = JJ
    LIFAM = IIFAM
    EXECUTE NIJKAK
    M(MINDEX) = 0
  3060 CONTINUE
3050 CONTINUE
3040 CONTINUE
END BLOCK

REMOTE BLOCK NIJKAK
REMOTE BLOCK NIJKAK
MINDA = 0
LI1 = LI+1
ISIGN = LI-1
IF (ISIGN.EQ.0) GO TO 3064
DO 3065 IX = 1, ISIGM
   MINDA = MINDA + (N - IX) * K
3065 CONTINUE
3064 MINDEX = MINDA + (LJ - LI1) * K + LIFAM
ENDBLOCK
C REMOTE BLOCK OBJROW
REMOTE BLOCK OBJROW
NML = N - 1
DO 3100 I = 1, NML
   II = I + 1
   DO 3110 J = II, N
      DO 3120 IFAM = 1, K
         LI = I
         LJ = J
         LIFAM = IFAM
         EXECUTE MIJKA
         N(MINDEX) = COEFF(I, J)
      3120 CONTINUE
3110 CONTINUE
ENDBLOCK
C REMOTE BLOCK PUTROW
REMOTE BLOCK PUTROW
XVARS = N * K
NVAR = N * (N - 1) / 2 * K
WRITE(8, 4000) (X(II), II = 1, XVARS)
WRITE(8, 4002) (M(JJ), JJ = 1, NVARS)
4002 FORMAT(9(‘,’ ,F7.2))
4000 FORMAT(9(‘,’ ,F7.2))
IF (FLAG.EQ.’OBJROW’) THEN
   WRITE(8, 4010) CTYPE
   4010 FORMAT(‘,’ ,A8, ‘.’)
ELSEIF (FLAG.EQ.’ALLOC’) THEN
   WRITE(8, 4020) CTYPE, RHS
   4020 FORMAT(‘,’ ,A8 ,I2)
ELSEIF (FLAG.EQ.’CNSTR’) THEN
   WRITE(8, 4020) CTYPE, RHS
ELSEIF (FLAG.EQ.’NLIM’) THEN
   WRITE(8, 4020) CTYPE, RHS
ELSEIF (FLAG.EQ.’INTER’) THEN
   WRITE(8, 4020) CTYPE
ELSEIF (FLAG.EQ.’UPPER’) THEN
   WRITE(8, 4020) CTYPE
ENDIF
EXECUTE ALLCLR
ENDBLOCK
C REMOTE BLOCK ALLOC
REMOTE BLOCK ALLOC
FLAG = ’ALLOC’
CTYPE = ’EQ’
RHS = 1
DO 3200 I=1,N
   DO 3210 IFAM=1,K
      XIND = (I-1)*K + IFAM
      X(XIND) = 1
   3210   CONTINUE
   EXECUTE PUTROW
   CONTINUE
3200   CONTINUE
       ENDBLOCK
C REMOTE BLOCK CNSTR
   REMOTE BLOCK CNSTR
   FLAG="CNSTR```
   CTYPE="LE```
   RHS=1
   NM1=N-1
   DO 3230 I=1,NM1
      IL=I+1
   DO 3240 J=IL,N
   DO 3250 IFAM=1,K
      LI=I
      LJP=J
      LIFAM=IFAM
      EXECUTE NIJKAK
      M(MINDEX) = -1
      XINDI = (I-1)*K + IFAM
      XINDJ = (J-1)*K + IFAM
      X(XINDI) = 1
      X(XINDJ) = 1
   EXECUTE PUTROW
   CONTINUE
3250   CONTINUE
3240   CONTINUE
3230   CONTINUE
       ENDBLOCK
C REMOTE BLOCK CNSTR2
   REMOTE BLOCK CNSTR2
   FLAG="CNSTR```
   CTYPE="LE```
   NM1=N-1
   DO 3400 I=1,NM1
   DO 3410 IFAM=1,K
      XIND = (I-1)*K + IFAM
      RHS=N-I
      X(XIND) = N-1
      IL=I+1
   DO 3420 J=IL,N
      LI=I
      LJP=J
      LIFAM=IFAM
      EXECUTE NIJKAK
      M(MINDEX) = -1
      XINDJ = (J-1)*K + IFAM
      X(XINDJ) = 1
   CONTINUE
   EXECUTE PUTROW
3420   CONTINUE
3410   CONTINUE
3400   CONTINUE
       ENDBLOCK
C REMOTE BLOCK MLIM
   REMOTE BLOCK MLIM
   FLAG="MLIM```
   CTYPE="LE```

RHS = 0
NM1 = N-1
DO 3260 I = 1, NM1
   IL = I + 1
   DO 3270 J = 1, N
      DO 3260 IFAM = 1, K
         LI = I
         LJ = J
         LIFAM = IFAM
         EXECUTE MIJKA
         M(MINDEX) = 1
         XINDI = (I-1)*K + IFAM
         X(XINDI) = 1
         EXECUTE PUTROW
         LI = I
         LJ = J
         LIFAM = IFAM
         EXECUTE MIJKA
         XINDJ = (J-1)*K + IFAM
         M(MINDEX) = 1
         X(XINDJ) = 1
         EXECUTE PUTROW
 3280 CONTINUE
 3270 CONTINUE
 3260 CONTINUE
END BLOCK
C REMOTE BLOCK MLIM2
REMOTE BLOCK MLIM2
FLAG = "MLIM"
CTYPE = "LE"
RHS = 0
DO 3500 I = 1, N
   DO 3510 IFAM = 1, K
      XIND = (I-1)*K + IFAM
      X(XIND) = I - N
   DO 3520 J = 1, N
      IF (J .EQ. I) GO TO 3520
      LI = I
      LJ = J
      LIFAM = IFAM
      EXECUTE MIJKA
      M(MINDEX) = 1
 3520 CONTINUE
EXECUTE PUTROW
3510 CONTINUE
3500 CONTINUE
END BLOCK
C REMOTE BLOCK INTEGER
REMOTE BLOCK INTEGER
FLAG = "INTEGER"
CTYPE = "INTEGER"
DO 3300 I = 1, N
   K11 = K - 1
   DO 3310 IFAM = 1, K11
      XIND = (I-1) * K + IFAM
      X(XIND) = 1
   3310 CONTINUE
3300 CONTINUE
EXECUTE PUTROW
END BLOCK
C REMOTE BLOCK UPPER
REMOTE BLOCK UPPER
FLAG="UPPER"
CTYPE="UPPERBD"
DO 3700 I=1,N
   DO 3710 IFAM=1,K
      XIND=(I-1)*K +IFAM
      X(XIND)=1
   3710   CONTINUE
3700   CONTINUE
NM1=N-1
DO 3730 I=1,NM1
   IL=I+1
   DO 3740 J=IL,N
      DO 3750 IFAM=1,K
         LI=I
         LJ=J
         LIJAM=IFAM
         EXECUTE MIJKA
         M(NINDEX)=1
3750—   CONTINUE
3740   CONTINUE
3730   CONTINUE
EXECUTE PUTROW
ENDBLOCK
9999 STOP
END
ENTRY $IBSYS
$STOP
//


//PD  JOB  (R240,NU7,3,5),"GAJ",CLASS=A,REGION=2048K
//  EXEC   WATFIV
//FO88FOO1 DD DSN=WYL.R240NUT.PAMOUT,UNIT=DSAD,Vol=SER-WORKPK,
//  DISP=(NEW,KEEP),SPACE=(TRK,(40,10)),
//  DCB=(LBECL=80,BLKSIZ=15440,RECFM=FB)
//GO.SYSIN DD *
$JOB  WATFIV
----------
C  REF. SECTION 4.3.2
C
C  CELL FORMATION IN FMS - PART FAMILY FORMATION PROBLEM
C
C  AUTHOR - GAJANANA NADOLI
C  GRADUATE STUDENT, DEPT. OF INDUSTRIAL ENGINEERING
C  UNIVERSITY OF WINDSOR, WINDSOR, ONTARIO N9B 222
C
C  THIS PROGRAM IMPLEMENTS THE ALGORITHM FOR FINDING THE
C  UPPER BOUND AND LOWER BOUND FOR SUBPROBLEMS OF THE TYPE
C  SOLVED IN THE ITERATIONS OF THE APPROXIMATION PROCEDURE
C  CALCULATING THESE BOUNDS WITH DIFFERENT VALUES OF R
C  ESTABLISHES THE REGION (L,U) FOR THESE SUBPROBLEMS.
C
C  THE VARIABLE DECLARATION SECTION ***********************
INTEGER NUMOP(30),SIM(30,30),DISIM(30,30)
INTEGER OPJ,A
INTEGER XND,XVAR,VAR,XNDI,XNDJ
INTEGER RHS
INTEGER HK(30,3),MEM(30,3),NNM(3),FAHN(3),FAMD(3)
INTEGER FSIM(30,3),FDISIM(30,3),NSIM(30,30)
INTEGER NDISIM(30,30)
REAL LAMDA,OP*4(30,50),COEFF(30,30),FCOEFF(30,3),M(500),K(500)
REAL S(315),LBOWN,FBM(3),WNC
CHARACTER CTYPE*8,FLAG*8
COMMON OP,NUMOP,NUM,DISIM,OPI,OPJ,COEFF,I,J,IFAM,K,
  L,NN,MP,HN,N,K1
C
C  THE DATA INPUT AND CALCULATION OF SIMILAR AND DISSIMILAR PROCESSES
C  BETWEEN THE PARTS **********************
READ(5,10) N,K
10 FORMAT(I2,1X,I2)
DO 100 I=1,N
   READ(5,20) NUM,OP(I,J),J=1,NUM
   NUMOP(I)=NUM
20 FORMAT(I2,10(I2,1X,1A4),/,(1X,10(A4,1X)))
WRITE(6,20) NUMOP(I),OP(I,J),J=1,NUM
100 CONTINUE
N1=N-1
DO 200 I=1,N1
   I=I+1
   DO 300 J=1,N
      SIM(I,J)=0
      NTERM=NUMOP(I)
      NTERMJ=NUMOP(J)
      DO 400 OPJ=1,NTERM
         IF (OP(I,OPJ).EQ.OP(J,OPJ)) SIM(I,J)=SIM(I,J)+1
300 CONTINUE
400 CONTINUE
C  DISIM(I,J)=NUMOP(I)+NUMOP(J)-2*SIM(I,J)
C  CONTINUE
C  CONTINUE
C  CONTINUE
C READING THE INITIAL CONFIGURATION
LAMDA=1.00
DO 2010 IFAM =1,K
   DO 2010 I=1,N
      MK(I,IFAM)=0
      MEM(I,IFAM)=0
   2010 CONTINUE
   DO 2020 IFAM=1,K
      READ(5,2030) NUNF,(MK(I,IFAM),I=1,NUNF)
      WRITE(6,2036) NUNF,(MK(I,IFAM),I=1,NUNF)
   2036 FORMAT('INITIAL CONFIGURATION',12,10(1X,I2))
      NNN(IFAM)=NUNF
   2030 FORMAT(12,20(1X,I2))
   2020 CONTINUE
   DO 2050 IFAM=1,K
      DO 2060 I=1,N
         LU=NNN(IFAM)
         DO 2070 LL=1,LU
            IF (MK(LL,IFAM).EQ.1) MEM(I,IFAM) = 1
         2070 CONTINUE
      2060 CONTINUE
   2050 CONTINUE
   TFAND=0.0
   DO 2200 IFAM=1,K
      N1=N-1
      FAMH(IFAM)=0
      FAMD(IFAM)=0
      DO 2240 I=1,N1
         I=I+1
         DO 2240 J=I,N
            FAMN(IFAM)=FAMN(IFAM)+MEM(I,IFAM)*MEM(J,IFAM)*DISIM(I,J)
            FAMD(IFAM)=FAMD(IFAM)+MEM(I,IFAM)*MEM(J,IFAM)*SIN(I,J)
         2240 CONTINUE
      TFAND=TFAND+FAMD(IFAM)
   2200 CONTINUE
   DO 188 IFAM=1,K
      FAMC(IFAM)=FAMN(IFAM)/TFAND
   188 CONTINUE
   MINDC=0
   MF=1
   DO 2260 IFAM=1,K
      IF (FAMC(IFAM).GT.MINDC) THEN
         MF=IFAM
         MINDC=FAMC(IFAM)
      ENDIF
   2260 CONTINUE
   C
   NEWN=NNN(MF)
   NEWN1=NEWN-1
   DO 2300 NI=1,NEWN1
      NI=NI+1
      DO 2400 NJ=NI,NEWN
         I=MK(NI,MF)
         J=MK(NJ,MF)
         IF (I.LT.J) THEN
            NSIM(I,NJ)=SIM(I,J)
            NSDIS(I,NJ)=DISIM(I,J)
         ENDIF
         IF (I.GT.J) THEN
            NSIM(NI,NJ)=SIM(J,I)
         ENDIF
   2400 CONTINUE
   2300 CONTINUE
NDISIM(NI,NJ)=DISIM(J,I)

ENDIF

2400  CONTINUE
2300  CONTINUE

DO 2401 I=1,N
    DO 2410 IFAM=1,K
        FSIM(I,IFAM)=0
        FDISIM(I,IFAM)=0
    CONTINUE
2410  CONTINUE

DO 2430 NI=1,NEWN
    NPART=MK(NI,MF)
    DO 2440 IFAM=1,K
        IF (IFAM.EQ.MF) GO TO 2440
        DO 2460 J=1,N
            IF (J.GT.NPART) THEN
                FSIM(NI,IFAM)=FSIM(NI,IFAM)+SIM(NPART,J)*MEM(J,IFAM)
                FDISIM(NI,IFAM)=FDISIM(NI,IFAM)+DISIM(NPART,J)*MEM(J,IFAM)
            ELSE
                FSIM(NI,IFAM)=FSIM(NI,IFAM)+SIM(J,NPART)*MEM(J,IFAM)
                FDISIM(NI,IFAM)=FDISIM(NI,IFAM)+DISIM(J,NPART)*MEM(J,IFAM)
            ENDIF
        CONTINUE
2460  CONTINUE
2440  CONTINUE
2430  CONTINUE

NI=NI-1
    DO 2462 I=1,N
        NI=NI+1
    CONTINUE

DO 2470 J=1,N
    SIM(I,J)=0
    DISIM(I,J)=0
2470  CONTINUE

WRITE(8,4018)
4018 FORMAT(" BOUNDS ON THE OBJECTIVE FUNCTION FOR DIFFERENT VALUES OF \alpha\"")
WRITE(8,4053)
WRITE(8,4024)
4024 FORMAT(" \alpha")
WRITE(8,4024)
WRITE(8,4026)
4026 FORMAT(" \# OF PARTS=",I2)
WRITE(8,4027)
4027 FORMAT(" \# OF FAMILIES=",I2)
WRITE(8,4024)
WRITE(8,4029)
4029 FORMAT(" \# OF NON-ZERO COEFFICIENTS=",I4)
WRITE(8,4053)
WRITE(8,4024)
WRITE(8,4024)
WRITE(8,9000)
9000 FORMAT(" R \# OF NEGATIVE OBJECTIVE FUNCTION BOUNDS")
9001 FORMAT(" COEFFICIENTS LOWERBOUND UPPERBOUND")
9002 FORMAT(" ------------- ---------------")
EXECUTE ALLCLR
DO 13 LLAMDA=200,500,10
LAMDA=LLAMDA/100.00
EXECUTE FAMIL
NM1=NM-1
DO 21 I=1,NM1
   IL=I+1
   DO 22 J=IL,N
      COEFF(I,J)=DISIM(I,J)-LAMDA*SIM(I,J)
      22 CONTINUE
   CONTINUE
21 CONTINUE
DO 89 JM=1,315
   S(JM)=0.0
89 CONTINUE
EXECUTE OBJROW
CONTINUE
GO TO 9999
C REMOTE BLOCK FAMIL
REMOTE BLOCK FAMIL
NEWN1=NEWN-1
   DO 2480 NI=1,NEWN1
      NI=NI+1
      DO 2490 NJ=NI1,NEWN
         SIM(NI,NJ)=NSIM(NI,NJ)
         DISIM(NI,NJ)=NDISIM(NI,NJ)
         COEFF(NI,NJ)=DISIM(NI,NJ)-LAMDA*SIM(NI,NJ)
      2490 CONTINUE
   CONTINUE
2480 CONTINUE
DO 2550 I=1,NEWN
   DO 2560 IFAM=1,K
      FCOEFF(I,IFAM)=FDISIM(I,IFAM)-LAMDA*FSIM(I,IFAM)
   2560 CONTINUE
2550 CONTINUE
C=0.0
DO 58 IFAM=1,K
   IF (IFAM.EQ.MF) GO TO 58
   C=C+FAMN(IFAM)-LAMDA*FAHNF(IFAM)
   IF (LAMDA.GT.0.15) GO TO 58
58 PRINT, 'CONSTANT','IFAM',' IS ','FAMN(IFAM)-LAMDA*FAHNF(IFAM)
   CONTINUE
   IF (LAMDA.GT.0.15) GO TO 76
   PRINT, 'THE CONSTANT FACTOR = ',C
   DO 45 I=1,N
      DO 46 IFAM=1,K
         IF (IFAM.EQ.MF) GO TO 46
         PRINT, 'FSIM(',I,IFAM,') = ',FSIM(I,IFAM)
         PRINT, 'FDISIM(',I,IFAM,') = ',FDISIM(I,IFAM)
      46 CONTINUE
   45 CONTINUE
76 DUNH=0.00
ENDBLOCK
C REMOTE BLOCK ALLCLR
REMOTE BLOCK ALLCLR
DO 3030 II=1,N
   DO 3030 IIFAM=1,K
      XIND=(II-1)*K+IIFAM
      XIND=0
   3030 CONTINUE
NM1=N-1
DO 3040 II=1,NM1
   II=II+1
   DO 3050 JJ=II1,N
3040 CONTINUE
3050 CONTINUE

DO 3060 IIFAM=1,K
   LI=II
   LJ=JJ
   LIFAM=IIFAM
   EXECUTE MIJKA
   M(MINDEX)=O
3060   CONTINUE
3050   CONTINUE
3040   END BLOCK
C REMOTE BLOCK MIJKA
   ***************
   REMOTE BLOCK MIJKA
   MINDA=0
   LI=LI+1
   ISIGH =LI-1
   IF (ISIGH.EQ.0) GO TO 3064
   DO 3065 IX = 1,ISIGH
       MINDA=MINDA + (N-IX) * K
3065   CONTINUE
3064   MINDEX=MINDA+ (LJ-LII)*K +LIFAM
   END BLOCK
C REMOTE BLOCK OBJROW
   ***************
   REMOTE BLOCK OBJROW
   NML= N-1
   NEG=0
   MS=1
   DO 3100 I=1,NML
       LI=I+1
       DO 3110 J=I,N,
       DO 3120 IFAH=1,K
       LI=I
       LJ=J
       LIFAH=IFAM
       EXECUTE MIJKA
       M(MINDEX) = COEFF(I,J)
       IF(COEFF(I,J).LT.0) NEG=NEG+1
       IF(IFAM.EQ.1) THEN
           S(MS)=COEFF(I,J)
           MS=MS+1
       ENDIF
3120   CONTINUE
3110   CONTINUE
3100   CONTINUE
   DO 312 I=1,N
   DO 312 IFAM=1,K
       XIND=(I-1)*K + IFAM
       X(XIND)=FCOEFF(1,IFAM)
       IF (I.EQ.1) X(XIND)=X(XIND)+C
312   CONTINUE
CTYPE="MIN"
FLAG="OBJROW"
LPOS=N*(N-1)/2*K-NEG
CALL SORT(315,S)
IES=N/K
IPAIRS=IES*(IES-1)/2*K+(N-IES*K)*IES
MPAIRS=N*(N-1)/2
LBOUND=0.000
UBOUND=0.000
ICOUNT=0
DO 3299 KI=1,315
INTS = S(KI)*10000.000
IF (INTS.NE.0) THEN
  ICOUNT = ICOUNT + 1
ELSE
  GO TO 889
ENDIF
IF (S(KI).LT.0.00) THEN
  LBOND = LBOND + S(KI)
ENDIF
IF (S(KI).GT.0.00) THEN
  IF (ICOUNT.LE.IPAIRS) LBOND = LBOND + S(KI)
ENDIF
889 CONTINUE
ICOUNT = 0
DO 899 KI = 1, 315
   INTS = S(315 - KI + 1)*10000.000
   IF (INTS.NE.0) THEN
      ICOUNT = ICOUNT + 1
   ELSE
      GO TO 899
   ENDIF
   IF (S(315 - KI + 1).GT.0.00) THEN
      UBOUND = UBOUND + S(315 - KI + 1)
   ENDIF
   IF (S(315 - KI + 1).LT.0.00) THEN
      IF (ICOUNT.LE.IPAIRS) UBOUND = UBOUND + S(315 - KI + 1)
   ENDIF
899 CONTINUE
DO 891 I = 1, N
   TM = TM - 1000000000.00
   TL = TL - 1000000000.00
   DO 892 IFAM = 1, K
      XIND = (I - 1)*K + IFAM
      IF (X(XIND).GT.TM) TM = X(XIND)
      IF (X(XIND).LT.TL) TL = X(XIND)
892 CONTINUE
ENDBLOCK
C REMOTE BLOCK PUTROW
REMOTE BLOCK PUTROW
EXECUTE ALLCLR
ENDBLOCK
9999 STOP
END
$ENTRY
$SIBSYS
$STOP
C CELL FORMATION IN FMS - MACHINE GROUP ALLOCATION
C
C AUTHOR - GAJANANA NADOLI
C GRADUATE STUDENT, DEPT. OF INDUSTRIAL ENGINEERING
C UNIVERSITY OF WINDSOR, WINDSOR, ONTARIO N9B 222
C
C THIS PROGRAM Generates the INPUT FILE OF THE PROBLEM
C "P1" FOR THE SAS/OR INTEGER PROGRAMMING ROUTINE.
C ALLOCATION OF MACHINE GROUPS TO THE PART FAMILIES FORMED.
C
C VARIABLE DECLARATION SECTION
INTEGER N(30),P,OPN,A(30,3,30,30),NUMOP(30,3),ROUTES(30,3)
INTEGER X(500),XIND,N(5000),IRR(30)
INTEGER TABL(3,1300,50),ENTRY(3),JCOUNT(30),KCOUNT(30)
INTEGER RPTF(3,1300),LID(99),RHS,NLID(15),CARQ,OPN1
INTEGER ONEOBJ(15,3),ONEAVL(15,3,30)
CHARACTER CTYPE*8,FLAG*8
C READING THE OPERATION DATA FOR THE PARTS
MAXOP=15
MAXMLT=1300
MAXWID=MAXOP+2+20
MQ=MAXOP+1
MSLN=MAXOP+2
READ(5,10) NMACH,K,(N(IFAM),IFAM=1,K)
10 FORMAT(I2,2X,I2,10(2X,12))
WRITE(6,10) NMACH,K,(N(IFAM),IFAM=1,K)
DO 100 IFAM=1,K
NUM=N(IFAM)
DO 110 J=1,NUM
READ(5,20) OPN
WRITE(6,20) OPN
20 FORMAT(I2)
NUMOP(IFAM,J)=OPN
DO 120 P=1,OPN
READ(5,30) (A(J,IFAM,P,IM),IM=1,NMACH)
WRITE(6,30) (A(J,IFAM,P,IM),IM=1,NMACH)
30 FORMAT(I15(I1,1X))
120 CONTINUE
110 CONTINUE
100 CONTINUE
DO 200 IFAM=1,K
DO 210 ISL=1,MAXMLT
RPTF(IFAM,ISL)=0
DO 220 IWID=1,MAXWID
TABL(IFAM,ISL,IWID)=0
220 CONTINUE
210 CONTINUE
ENTRY(IFAM)=0
200 CONTINUE
DO 4100 IFAM=1,K
DO 4110 MNM=1,NMACH
ONEOBJ(MNM,IFAM)=0
4100 CONTINUE
NPARTS=N(IFAM)
DO 4120 JJJ=1,NPARTS
ONEAVL(MIH,IFAM,JJJ)=0
4120 CONTINUE
4110 CONTINUE
4100 CONTINUE
C MAIN SEGMENT OF THE PROGRAM
EXECUTE MTERMS
EXECUTE CLRALL
EXECUTE OBJROW
EXECUTE AVAIL
EXECUTE ALLOC
EXECUTE TYP101
EXECUTE TYP201
EXECUTE INTGER
EXECUTE UPPER
EXECUTE OUT
GO TO 9999
C REMOTE BLOCK OBJROW
REMOTE BLOCK OBJROW
EXECUTE MCOBJ
EXECUTE MLTOBJ
FLAG="SPL"
CTYPE="MAX"
EXECUTE PUTROW
ENDBLOCK
C REMOTE BLOCK MCOBJ
REMOTE BLOCK MCOBJ
DO 1000 MH=1,NMACH
DO 1010 IIFAM=1,K
XIND=(MH-1)*K+IIFAM
X(XIND)=ONEOBJ(MH,IIFAM)
1010 CONTINUE
1000 CONTINUE
ENDBLOCK
C REMOTE BLOCK UPER
REMOTE BLOCK UPER
DO 700 MM=1,NMACH
DO 710 IIFAM=1,K
XIND=(MM-1)*K+IIFAM
X(XIND)=1
710 CONTINUE
700 CONTINUE
MIND=0
DO 711 IIFAM=1,K
NENTRY=ENTRY(IFAM)
DO 712 IE=1,NENTRY
MIND=MIND+1
M(MIND)=1
712 CONTINUE
711 CONTINUE
FLAG="SPL"
CTYPE="UPPERBD"
EXECUTE PUTROW
ENDBLOCK
C REMOTE BLOCK MTERMS
REMOTE BLOCK MTERMS
DO 2000 IFAM=1,K
NUH=N(IFAM)
DO 2010 J=1,NUH
**OPN=NUMOP(IFAM,J)**

### EXECUTE GEN

2010  CONTINUE
2000  CONTINUE
C REMOTE BLOCK GEN
REMOTE BLOCK GEN
ROUTES(J,IFAM)=0
IA=1
P=1
ID1=0
ID2=0
ID3=0
ID4=0
ID5=0
ID6=0
ID7=0
ID8=0
ID9=0
ID10=0
ID11=0
ID12=0
ID13=0
ID14=0
ID15=0
DO 3010 ID1=1,NMACH
  IA=1
  IA=IA*(A(J,IFAM,1,ID1))
  IF (IA.EQ.0) THEN
    GO TO 3010
  ENDF
  IF (OPN.EQ.1) THEN
    ROUTES(J,IFAM)=ROUTES(J,IFAM)+1
    EXECUTE RTFILE
    GO TO 3010
  ENDF
DO 3020 ID2=1,NMACH
  IA=1
  IA=IA*(A(J,IFAM,2,ID2))
  IF (IA.EQ.0) THEN
    GO TO 3020
  ENDF
  IF (OPN.EQ.2) THEN
    ROUTES(J,IFAM)=ROUTES(J,IFAM)+1
    EXECUTE RTFILE
    GO TO 3020
  ENDF
DO 3030 ID3=1,NMACH
  IA=1
  IA=IA*(A(J,IFAM,3,ID3))
  IF (IA.EQ.0) THEN
    GO TO 3030
  ENDF
  IF (OPN.EQ.3) THEN
    ROUTES(J,IFAM)=ROUTES(J,IFAM)+1
    EXECUTE RTFILE
    GO TO 3030
  ENDF
DO 3040 ID4=1,NMACH
  IA=1
IA = IA*(A(J, IFAM, 4, ID4))
IF (IA.EQ.0) THEN
   GO TO 3040
ENDIF
IF (OPN.EQ.4) THEN
   ROUTES(J, IFAM) = ROUTES(J, IFAM) + 1
   EXECUTE RTFILE
   GO TO 3040
ENDIF
DO 3050 ID5 = 1, NMACH
   IA = 1
   IA = IA*(A(J, IFAM, 5, ID5))
   IF (IA.EQ.0) THEN
      GO TO 3050
   ENDIF
   IF (OPN.EQ.5) THEN
      ROUTES(J, IFAM) = ROUTES(J, IFAM) + 1
      EXECUTE RTFILE
      GO TO 3050
   ENDIF
   DO 3060 ID6 = 1, NMACH
      IA = 1
      IA = IA*(A(J, IFAM, 6, ID6))
      IF (IA.EQ.0) THEN
         GO TO 3060
      ENDIF
      IF (OPN.EQ.6) THEN
         ROUTES(J, IFAM) = ROUTES(J, IFAM) + 1
         EXECUTE RTFILE
         GO TO 3060
      ENDIF
      DO 3070 ID7 = 1, NMACH
         IA = 1
         IA = IA*(A(J, IFAM, 7, ID7))
         IF (IA.EQ.0) THEN
            GO TO 3070
         ENDIF
         IF (OPN.EQ.7) THEN
            ROUTES(J, IFAM) = ROUTES(J, IFAM) + 1
            EXECUTE RTFILE
            GO TO 3070
         ENDIF
   ENDIF
   DO 3080 ID8 = 1, NMACH
      IA = 1
      IA = IA*(A(J, IFAM, 8, ID8))
      IF (IA.EQ.0) THEN
         GO TO 3080
      ENDIF
      IF (OPN.EQ.8) THEN
         ROUTES(J, IFAM) = ROUTES(J, IFAM) + 1
         EXECUTE RTFILE
         GO TO 3080
      ENDIF
      DO 3090 ID9 = 1, NMACH
         IA = 1
         IA = IA*(A(J, IFAM, 9, ID9))
         IF (IA.EQ.0) THEN
            GO TO 3090
         ENDIF
         IF (OPN.EQ.9) THEN
            Do
ROUTES(J,IFAM)=ROUTES(J,IFAM)+1
EXECUTE RTFILE
GO TO 3090
ENDDIF
DO 3100 ID10=1,NMACH
IA=1
IA=IA*(A(J,IFAM,10,ID10))
IF (IA.EQ.0) THEN
GO TO 3100
ENDIF
IF (OPN.EQ.10) THEN
ROUTES(J,IFAM)=ROUTES(J,IFAM)+1
EXECUTE RTFILE
GO TO 3100
ENDIF
DO 3110 ID11=1,NMACH
IA=1
IA=IA*(A(J,IFAM,11,ID11))
IF (IA.EQ.0) THEN
GO TO 3110
ENDIF
IF (OPN.EQ.11) THEN
ROUTES(J,IFAM)=ROUTES(J,IFAM)+1
EXECUTE RTFILE
GO TO 3110
ENDIF
DO 3120 ID12=1,NMACH
IA=1
IA=IA*(A(J,IFAM,12,ID12))
IF (IA.EQ.0) THEN
GO TO 3120
ENDIF
IF (OPN.EQ.12) THEN
ROUTES(J,IFAM)=ROUTES(J,IFAM)+1
EXECUTE RTFILE
GO TO 3120
ENDIF
DO 3130 ID13=1,NMACH
IA=1
IA=IA*(A(J,IFAM,13,ID13))
IF (IA.EQ.0) THEN
GO TO 3130
ENDIF
IF (OPN.EQ.13) THEN
ROUTES(J,IFAM)=ROUTES(J,IFAM)+1
EXECUTE RTFILE
GO TO 3130
ENDIF
DO 3140 ID14=1,NMACH
IA=1
IA=IA*(A(J,IFAM,14,ID14))
IF (IA.EQ.0) THEN
GO TO 3140
ENDIF
IF (OPN.EQ.14) THEN
ROUTES(J,IFAM)=ROUTES(J,IFAM)+1
EXECUTE RTFILE
GO TO 3140
ENDIF
DO 3150 ID15=1,NMACH
IA=1
IA=IA*(A(J,IFAM,15,ID15))
IF (IA.EQ.0) THEN
   GO TO 3150
ENDIF
   IF (OPN.EQ.15) THEN
      ROUTES(J,IFAM)= ROUTES(J,IFAM)+1
   EXECUTE RTFILE
   GO TO 3150
ENDIF
3150 CONTINUE
3140 CONTINUE
3130 CONTINUE
3120 CONTINUE
3110 CONTINUE
3100 CONTINUE
3090 CONTINUE
3080 CONTINUE
3070 CONTINUE
3060 CONTINUE
3050 CONTINUE
3040 CONTINUE
3030 CONTINUE
3020 CONTINUE
3010 CONTINUE
ENDBLOCK
C REMOTE BLOCK RTFILE
   REMOTE BLOCK RTFILE
   IDUP=0
   LID(1)=ID1
   LID(2)=ID2
   LID(3)=ID3
   LID(4)=ID4
   LID(5)=ID5
   LID(6)=ID6
   LID(7)=ID7
   LID(8)=ID8
   LID(9)=ID9
   LID(10)=ID10
   LID(11)=ID11
   LID(12)=ID12
   LID(13)=ID13
   LID(14)=ID14
   LID(15)=ID15
   DO 1 IXX=1,15
      NLID(IXX)=0
   CONTINUE
      CARQ=OPN
      OPNL=OPN-1
   DO 2 IX=1,OPNL
      IF (LID(IX).EQ.999) GO TO 2
      IX1=IX+1
   DO 3 IY=IX1,OPN
      IF (LID(IY).EQ.LID(IX)) THEN
         LID(IY)=999
         CARQ=CARQ-1
      ENDIF
   CONTINUE
   CONTINUE
   NC=0
ENDBLOCK
C REMOTE BLOCK MLTOBJ
REMOTE BLOCK MLTOBJ
MIND=0
DO 2100 IFAM=1,K
 NENTRY=IENTRY(IFAM)
 DO 2110 IE=1,NENTRY
 MIND=MIND+1
 M(MIND)=RPTF(IFAM,IE)+1
2110 CONTINUE
2100 CONTINUE
ENDBLOCK
C REMOTE BLOCK AVAIL
REMOTE BLOCK AVAIL
MINDF=0
DO 2140 IFAM=1,K
 IF (IFAM.NE.1) MINDF=MINDF+IENTRY(IFAM-1)
 MIND=MINDF
 NENTRY=IENTRY(IFAM)
 NPARTS=N(IFAM)
 DO 2150 JJ=1, NPARTS
 DO 2160 IE=1,NENTRY
 MIND=MIND+1
 IREP=MSLN+RPTF(IFAM,IE)+1
 C IF (IFAM.EQ.1) THEN
 C IF (IE.EQ.3) PRINT,˝RPTF+1˝, RPTF(IFAM,IE)+1
 C IF
 M=N=MSLN+1
 MCOEF=0
 C DO 2170 IX=IB,IREP
 C IF (TABL(IFAM,IE,IX).EQ.JJ) MCOEF=MCOEF+1
2170 CONTINUE
 JPOS=MSLN+JJ;
 M(NPOS)=TABL(IFAM,IE,JPOS)
 M(MIND)=MCOEF
2160 CONTINUE
 DO 4600 MMH=1,NMACH
 XIND=(MMH-1)*K+IFAM
 X(XIND)=ONEAVL(MMH,IFAM,JJ)
4600 CONTINUE
CTYPE=˝CE˝
RHS=1
EXECUTE PUTROW
MIND=MINDF
2150 CONTINUE
2140 CONTINUE
ENDBLOCK
C REMOTE BLOCK TYP101
REMOTE BLOCK TYP101
MIND=0
DO 2300 IFAM=1,K
 NENTRY=IENTRY(IFAM)
 DO 2310 IE=1,NENTRY
 MIND=MIND+1
 M(MIND)+1
 OPN=TABL(IFAM,IE,MQ)
 DO 2320 JJ=1,OPN
 IX=TABL(IFAM,IE,IJ)
 XIND=(IX-1)*K+IFAM
 X(XIND)=X(XIND)+1
DO 4 IX = 1, 0PN
   IF (LID(IX) .NE. 999) THEN
      NC = NC + 1
      NLID(NC) = LID(IX)
   ENDIF
   CONTINUE
   IF (NC .NE. CARQ) PRINT, "********ERROR********" C
   PRINT, "J, IFAM, NC", J, IFAM, NC
   IF (CARQ .EQ. 1) THEN
      DO 4500 IXY = 1, 0PN
         IF (LID(IXY) .NE. 999) THEN
            MMM = LID(IXY)
            ONEOBJ(MMM, IFAM) = ONEOBJ(MMM, IFAM) + 1
            ONEAVL(MMM, IFAM, J) = ONEAVL(MMM, IFAM, J) + 1
         ENDIF
      ENDIF
      4500 CONTINUE
   GO TO 8
   ENDIF
   IF (ENTRY(IFAM) .EQ. 0) THEN
      GO TO 6500
   ELSE
      IENTRY = ENTRY(IFAM)
      DO 6000 ICHK = 1, IENTRY
         IF (IDUP .EQ. 1) GO TO 6500
         IF (TABL(IFAM, ICHK, MQ) .NE. CARQ) GO TO 6000
         DO 6010 JH = 1, NHACH
            JCOUNT(JH) = 0
            DO 6020 IX = 1, CARQ
               IF (NLID(IX) .EQ. JH) JCOUNT(JH) = JCOUNT(JH) + 1
            CONTINUE
            KCOUNT(JH) = 0
            DO 6030 IY = 1, CARQ
               IF (TABL(IFAM, ICHK, IY) .EQ. JH) KCOUNT(JH) = KCOUNT(JH) + 1
            CONTINUE
            IF (JCOUNT(JH) .NE. KCOUNT(JH)) GO TO 6000
         CONTINUE
         IDUP = 1
         IDUPSL = ICHK
      6000 CONTINUE
   ENDIF
   IF (IDUP .EQ. 1) THEN
      RPTF(IFAM, IDUPSL) = RPTF(IFAM, IDUPSL) + 1
      C
      JPOS = MSLN + 1 + RPTF(IFAM, IDUPSL)
      TABL(IFAM, IDUPSL, JPOS) = J
      JPOS = MSLN + J
      TABL(IFAM, IDUPSL, JPOS) = TABL(IFAM, IDUPSL, JPOS) + 1
   ELSE
      IENTRY(IFAM) = IENTRY(IFAM) + 1
      NENTRY = IENTRY(IFAM)
      DO 6550 IX = 1, CARQ
         TABL(IFAM, NENTRY, IX) = NLID(IX)
      CONTINUE
      TABL(IFAM, NENTRY, HQ) = CARQ
      TABL(IFAM, NENTRY, MSLN) = NENTRY
      JFIRST = MSLN + 1
      C
      TABL(IFAM, NENTRY, JFIRST) = J
      JPOS = MSLN + J
      TABL(IFAM, NENTRY, JPOS) = TABL(IFAM, NENTRY, JPOS) + 1
   ENDIF
   6 DUZN = 0.00
2320    CONTINUE
        RHS=OPN-1
        CTYPE="LE"
        EXECUTE PUTROW
2310.   CONTINUE
2300    CONTINUE
        ENDBLOCK
C REMOTE BLOCK TYP201
REMOTE BLOCK TYP201
MIND=0
DO 2400 IPAM=1,K
    NENTRY=ENTRY(IFAM)
    DO 2410 IE=1,NENTRY
        MIND=MIND+1
        OPN=TABLE(IFAM,IE,MQ)
        DO 954 MREP=1,NMACH
            IRR(MREP)=0
        954 CONTINUE
        DO 2420 IJ=1,OPN
            X(MIND)=1
            IX=TABLE(IFAM,IE,IJ)
            IF (IRR(IX).EQ.0) THEN
                XIND=(IX-1)*K +IFAM
                X(XIND)=1
                CTYPE="LE"
                RHS=0
                EXECUTE PUTROW
            ENDIF
            IRR(IX)=1
        2420 CONTINUE
2410 CONTINUE
2400 CONTINUE
        ENDBLOCK
C REMOTE BLOCK INTGER
REMOTE BLOCK INTGER
DO '2600 INTM=1,NMACH
    INTK=K-1
    DO 2610 INTFAM=1,INTK
        IXIND=(INTM-1)*K +INTFAM
        X(IXIND)=1
    2610 CONTINUE
2600 CONTINUE
        CTYPE = "INTEGER"
        FLAG="SPL"
        EXECUTE PUTROW
        ENDBLOCK
C REMOTE BLOCK PUTROW
REMOTE BLOCK PUTROW
IXTRHS = NMACH*K
INTRHS=0
DO 2500 IIIFAM=1,K
    INTRHS=INTRHS+ENTRY(IIIFAM)
2500 CONTINUE
WRITE(8,7000) (X(IPR),IPR=1,IXTRHS),(M(IPR),IPR=1,INTRHS)
7000 FORMAT(2O(IK,I3))
    IF (FLAG.EQ."SPL") THEN
        WRITE(8,7010) CTYPE
    7010 FORMAT(IK,A8,')
    ELSE
        WRITE(8,7020) CTYPE,RHS
7020   FORMAT(1X,A8,1X,I2)
ENDIF
FLAG="REG"
EXECUTE CLRALL
ENDBLOCK
C REMOTE BLOCK CLRALL
REMOTE BLOCK CLRALL
DO 2700 HCLR=1,NMACH
   DO 2710 ICLRF=1,K
      XIND=(HCLR-1)*K + ICLRF
      X(XIND)=0
   CONTINUE
2710   CONTINUE / IMTRMS=0
   DO 2730 ICLRF=1,K
      IMTRMS=IMTRMS+ENTRY(ICLRF)
2730   CONTINUE
   DO 2750 LMIND=1,IMTRMS
      X(LMIND)=0
   CONTINUE
2750   CONTINUE
   FLAG="REG"
   ENDBLOCK
REMOTE BLOCK ALLOC
DO 5000 IIM=1,NMACH
   DO 5010 IFAM=1,K
      XIND = (IIM-1)*K + IFAM
      X(XIND) = 1
5010   CONTINUE
   RHS=1
   CTYPE="EQ"
   IF (.NOT.(IIM.EQ.2)) THEN
      RHS = 2
   ENDIF
   IF (IIM.EQ.6) RHS=2
   EXECUTE PUTROW
5000   CONTINUE
END BLOCK
9999 STOP
END
ENTRY
  1203060405
03 PART 1
  0 0 0 1 0 1 1 0 0 1 0
  1 0 0 0 0 0 1 0 0 0
  0 0 1 0 0 0 0 0 0 1 0
02 PART 3
  0 0 0 1 1 0 1 0 1 0
  0 0 0 0 1 0 0 0 1 0 1
03 PART 4
  0 0 0 0 0 1 0 1 0 0 0
  0 1 0 0 1 0 0 0 0 0 1
  0 0 0 1 0 0 0 0 0 0
03 PART 5
  0 1 0 0 0 1 0 0 0 0 0
  0 0 1 0 0 0 0 0 0 0 0
  0 0 0 0 1 0 0 0 0 0
03 PART 8
  1 0 1 0 0 0 0 0 0 0 0
  1 0 0 0 0 0 1 0 0 0
  0 1 0 0 0 0 0 0 1 0 0
03 PART 13
0 1 0 0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 0 0 0 1
0 0 1 0 0 0 0 0 1 0 0 0
02 PART 6
1 0 0 0 0 0 0 0 0 0 1 0
0 0 0 1 1 1 0 0 1 0 0 0
03 PART 11
1 0 1 0 0 0 0 0 0 0 0 0
0 0 1 0 0 0 0 1 0 0 0
1 0 0 0 0 0 0 1 0 0 0
03 PART 12
1 0 0 1 0 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0 1
1 0 0 0 0 0 0 1 0 0 0
03 PART 15
1 0 1 0 0 0 0 0 0 0 0 0
0 1 0 0 1 0 0 0 0 0 0 0
0 0 1 0 0 0 0 0 0 0 1 0
02 PART 2
1 0 0 1 0 1 0 0 0 0 0 0
1 0 0 0 1 0 0 0 0 1 0 0
04 PART 7
0 1 0 0 0 0 0 0 0 1 1 0
0 0 0 0 0 0 1 0 0 0 0 1
0 0 0 1 0 0 1 0 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0 0
03 PART 9
0 1 0 0 0 1 0 0 0 1 0 0
0 1 0 0 0 1 0 0 0 0 0 0
0 0 1 0 0 0 0 0 0 1 0 0
02 PART 10
0 1 0 0 0 1 0 0 0 1 0 1
0 0 0 1 0 0 1 0 0 0 0 0
04 PART 14
0 0 0 0 0 1 1 0 0 0 0 0
0 0 0 0 0 1 0 0 0 0 0 0
0 0 1 0 0 0 1 0 0 1 0 0
0 1 0 1 0 0 0 0 0 0 0 0
SIBSYS
$STOP
//
//PF JOB (R240,NU7,5,5), "GAI", CLASS=A, REGION=2048K
// EXEC WATFIV
// FTP08F001 DD DSN=WYR.R240NU7.PAMOUT, UNIT=DSAD, VOL=SER=WORKPK.
// DISP=(NEW,KEEP), SPACE=(TRK,(40,10)),
// DGB=(LRECL=80, BLKSIZE=15440, RECPS=FB)
// G0 SYSTP DD
$JOB WATFIV REF. SECTION 5.3.2
C ============================================================
C CELL FORMATION IN FMS - MACHINE GROUP ALLOCATION
C ============================================================
C AUTHOR - GAJANANA NADOLI
C GRADUATE STUDENT, DEPT. OF INDUSTRIAL ENGINEERING
C UNIVERSITY OF WINDSOR, WINDSOR, ONTARIO N9B 222
C
C THIS PROGRAM GENERATES THE INPUT FILE OF THE PROBLEM
C "INF" FOR THE SAS/OR INTEGER PROGRAMMING ROUTINE.
C IDENTIFICATION OF THE MACHINES CAUSING INFEASIBILITY IN
C MACHINE GROUP ALLOCATION.
C
C VARIABLE DECLARATION SECTION ********************
INTEGER N(50), P, OPN, A(30, 3, 30), NUMOP(3, 30), ROUTES(30, 3)
INTEGER X(500), XIND, M(5000), IRR(30), XINDJ, XINDK
INTEGER TABL(3, 1300, 50), ENTRY(3), JCOUNT(30), KCOUNT(30)
INTEGER RPTF(3, 1300), LID(99), RHS, NLOAD(15), CARQ, OPNL
INTEGER ONEOBJ(15, 3), ONEAVL(15, 3, 30)
CHARACTER CTYYPE*8, FLAG*N

C READING THE OPERATION DATA FOR THE PARTS ********************
IMPOS*2
MAXOP*15
MAXMLT*1300
MAXWID*MAXOP+2+20
MAXW*MAXOP+1
MAXW*MAXOP+2
READ(5, 10) NMACH, K, (N(IFAM), IFAM=1, K)
FORMAT(12, 2X, 12, 10(2X, I2))
WRITE(6, 10) NMACH, K, (N(IFAM), IFAM=1, K)
DO 100 IFAM=1, K
NUM=N(IFAM)
DO 110 J=1, NUM
READ(5, 20) OPN
WRITE(6, 20) OPN
FORMAT(12)
NUMOP(IFAM, J)=OPN
DO 120 P=1, OPN
READ(5, 30) (A(J, IFAM, P, IN), IN=1, NUMACH)
WRITE(6, 30) (A(J, IFAM, P, IN), IN=1, NUMACH)
FORMAT(12)
120 CONTINUE
110 CONTINUE
100 CONTINUE
DO 200 IFAM=1, K
DO 210 I=1, MAXMLT
RPTF(IFAM, I)=0
DO 220 J=1, MAXWID
TABL(IFAM, I, J)=0
220 CONTINUE
210 CONTINUE
ENTRY(IFAM)=0
200 CONTINUE
DO 400 IFAM=1, K

220 CONTINUE
   CONTINUE
   IENTRY(IFAH)=0
200 CONTINUE
   DO 4100 IFAM=1,K
   DO 4110 MMM=1,NMACH
   ONEOBJ(MMM,IFAH)=0
   NPARTS=N(IFAH)
   DO 4120 JJJ=1,NPARTS
   ONEAVL(MMM,IFAH,JJJ)=0
4120 CONTINUE
4110 CONTINUE
4100 CONTINUE
   MTRMS=NMAC*K*(K-1)/2+NMAC
C MAIN SEGMENT OF THE PROGRAM
EXECUTE CLRALL
EXECUTE NEWOBJ
EXECUTE MINI
EXECUTE ALLOC
EXECUTE LINI
EXECUTE INTGER
EXECUTE UPPER
GO TO 9999
C REMOTE BLOCK NEWOBJ
REMOTE BLOCK NEWOBJ
DO 1001 HH=1,NMACH
   DO 1011 IIAH=1,K
   XIND=(HH-1)*K+IIAH
   X(XIND)=1
1011 CONTINUE
1001 CONTINUE
   MIND=1
   DO 8000 MMS=1,NMACH
      K1=K-1
      DO 8010 JFAM=1,K1
         K2=JFAM+1
         DO 8020 KFAM=K2,K
            M(MIND)=37
            MIND=MIND+1
8020 CONTINUE
8010 CONTINUE
8000 CONTINUE
   DO 8030 MMS=1,NMACH
      M(MIND)=1369
      MIND=MIND+1
8030 CONTINUE
8500 DUMM=0.00
   FLAG="SPL"
   CTYPE="MIN"
EXECUTE PUTROW
ENDBLOCK
C REMOTE BLOCK MINI
REMOTE BLOCK MINI
DO 9000 IFAM=1,K
   NUM=N(IFAH)
DO 9010 J=1,NUM
                   OPN=NUMOP(IFAH,J)
DO 9020 P=1,OPN
   DO 9030 RHS=1,NMACH
      XIND=(MHS-1)*K+IFAM
      X(XIND)=A(J,IFAH,P,MHS)
9030 CONTINUE
CTYPE="GE"
RHS=1
EXECUTE PUTROW
9020 CONTINUE
9010 CONTINUE
9000 CONTINUE
ENDBLOCK
C REMOTE BLOCK UPPER
REMOTE BLOCK UPPER
DO 700 MM=1,NMACH
   DO 710 IIFAM=1,K
      XIND=(MM-1)*K+IIFAM
      X(XIND)=1
710 CONTINUE
700 CONTINUE
MIND=1
DO 8005 MHS=1,NMACH
   K1=K-1
   DO 8015 JFAM=1,K1
      K2=JFAM+1
      DO 8025 KFAH=K2,K
         M(MIND)=1
         MIND=MIND+1
8025 CONTINUE
8015 CONTINUE
8005 CONTINUE
   DO 8035 MHS=1,NMACH
      M(MIND)=1
      MIND=MIND+1
8035 CONTINUE
8505 DUMMM=0.00
C MIND=0
C DO 711 IFAH=1,K
C NENTRY=IENTRY(IFAM)
C DO 712 IE=1,NENTRY
C MIND=MIND+1
C M(MIND)=1
C712 CONTINUE
C711 CONTINUE
FLAG="SPL"
CTYPE="UPPERBD"
EXECUTE PUTROW
ENDBLOCK
REMOTE BLOCK LIN1
K:MIND=1
DO 8002 LHMS=1,NMACH
   K1=K-1
   DO 8012 LJFAH=1,K1
K2 = LINFAM + 1
DO 8022 LKFAM = K2, K
   XINDJ = (LMHS - 1) * K + LINFAM
   XINDK = (LMHS - 1) * K + LKFAM
   X(XINDJ) = 1
   X(XINDK) = 1
   M(KMIND) = -1
   CTYPE = 'LE'
   RHS = 1
   EXECUTE PUTROW
   X(XINDJ) = -1
   M(KMIND) = 1
   CTYPE = 'LE'
   RHS = 0
   EXECUTE PUTROW
   KMIND = KMIND + 1
8022 CONTINUE

8012 CONTINUE
8002 CONTINUE
DO 8015 KMH = 1, NMACH
   M(KMIND) = -1
   DO 8018 IFN = 1, K
      IXIND = (KMH - 1) * K + IFN
      X(IXIND) = 1
8018 CONTINUE
   RHS = 2
   CTYPE = 'LE'
   EXECUTE PUTROW
DO 8609 IFN = 1, K
   M(KMIND) = 1
   IXIND = (KMH - 1) * K + IFN
   X(IXIND) = -1
   RHS = 0
   CTYPE = 'LE'
   EXECUTE PUTROW
8609 CONTINUE
   KMIND = KMIND + 1
8815 CONTINUE
ENDBLOCK
C REMOTE BLOCK INTEGER
REMOTE BLOCK INTEGER
DO 2600 INTM = 1, NMACH
   INTK = K - 1
   DO 2610 INTFAM = 1, INTK
      IXIND = (INTM - 1) * K + INTFAM
      X(IXIND) = 1
2610 CONTINUE
2600 CONTINUE
   CTYPE = 'INTEGER'
   FLAG = 'SPL'
   EXECUTE PUTROW
ENDBLOCK
EXECUTE PUTROW
ENDBLOCK
C REMOTE BLOCK PUTROW
REMOTE BLOCK PUTROW
IXTRMS = NMACH*K
C IMTRMS=0
C DO 2500 IIFAH=1,K
C IMTRMS=IMTRMS+ENTRY(IIFAH)
C2500 CONTINUE
    WRITE(8,7000) (X(IPR),IPR=1,IXTRMS),(M(JPR),JPR=1,MTRMS)
7000 FORMAT(15(1X,I4))
    IF (FLAG.EQ."SPL") THEN
        WRITE(8,7010) CTYP
7010 FORMAT(1X,A8," .")
    ELSE
        WRITE(8,7020) CTYP,RHS
7020 FORMAT(1X,A8,1X,I2)
ENDIF
FLAG="REG"
EXECUTE CLRALL
ENDBLOCK
C REMOTE BLOCK CLRALL
REMOTE BLOCK CLRALL
DO 2700 MCLR=1,NMACH
    DO 2710 ICLR=1,K
        XIND=(MCLR-1)*K+ICLR
        X(XIND)=0
    CONTINUE
2710 CONTINUE
2700 CONTINUE
MIND=1
    DO 8001 MHS=1,NMACH
        K1=K-1
        DO 8011 JFAM=1,K1
            K2=JFAM+1
            DO 8021 KFAM=K2,K
                M(MIND)=0
                MIND=MIND+1
3021 CONTINUE
8021 CONTINUE
8011 CONTINUE
8001 CONTINUE
    DO 8031 MHS=1,NMACH
        M(MIND)=0
        MIND=MIND+1
8031 CONTINUE
8501 DUMMY=0.00
REMOTE BLOCK ALLOC
    DO 5000 IIM=1,NMACH
        DO 5010 IFAM=1,K
            XIND=(IIM-1)*K+IFAM
            X(XIND) = 1
5010 CONTINUE
RHS=3
CTYP="LE"
EXECUTE PUTROW
5000 CONTINUE
ENDBLOCK
STOP
END
ENTRY
$1BSYS
$STOP
//
APPENDIX D

ITERATION LOGS FOR DIFFERENT TRIALS OF APPROXIMATION PROCEDURE
### Table 9b

**Iteration Log for the Approximation Procedure**

Number of Parts = 15
Random Starting Partition = 2

<table>
<thead>
<tr>
<th>ITR</th>
<th>INITIAL ALLOCATION</th>
<th>INTERMEDIATE OBJ R AND FINAL FN. NEW ALLOCATIONS Z(R,X) CDC Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>[1,2,11,12,13,14]</td>
<td>3.55 [2,11] -77.5 3.283 High, [3,4,9,10,15] Choose [5,6,7,8,1,12,13,14]</td>
</tr>
<tr>
<td></td>
<td>[3,4,9,10,15]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>[5,6,7,8]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MF=1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CDC=3.697</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3.283 Same as above 0.01 3.283 OK</td>
<td></td>
</tr>
</tbody>
</table>

| 1   | [2,11] [3,4,9,10,15] [5,6,7,8,1,12,13,14] | 3.05 [2,11] 1.8 3.053 Low, [3,4,9,10,15,1,5,7,13,14] Choose [6,8,12] |
|     |                                               |                                                               |
|     | MF=3                                           |                                                               |
|     | CDC=3.283                                       |                                                               |
|     | 3.053 Same as above 0.01 3.053 OK |

| 2   | [2,11] [3,4,9,10,15,1,5,7,13,14] [6,8,12] | 2.70 [2,11,15] 73.7 2.931 R low [5,7,9,10,13,14] Choose [6,8,12,1,3,4] |
|     |                                               |                                                               |
|     | MF=2                                           |                                                               |
|     | CDC=3.053                                       |                                                               |
|     | 2.93 [2,11] -0.10 2.93 OK |

(Contd.)
<table>
<thead>
<tr>
<th>Iter No.</th>
<th>INITIAL ALLOCATION</th>
<th>INTERMEDIATE OBJ AND FINAL FN ALLOCATIONS</th>
<th>NEW OBJ Z(R,X)</th>
<th>CDC Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>[2,11]</td>
<td>* Based on range analysis choose R=2.775</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[1,4,5,7,9,10,13,14]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[3,6,8,12,15]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MF=3</td>
<td>2.775</td>
<td>[2,11]</td>
<td>Low,</td>
</tr>
<tr>
<td></td>
<td>CDC=2.93</td>
<td></td>
<td>[5,7,9,10,13,14]</td>
<td>Choose</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>[3,6,8,12,15,14]</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>2.928</td>
<td>Same as above</td>
<td>OK</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2.928</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>[2,11]</td>
<td>* Based on range analysis choose R=2.93</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[5,7,9,10,13,14]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[3,6,8,12,15,14]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MF=3</td>
<td>2.93</td>
<td>[2,11,12]</td>
<td>Low,</td>
</tr>
<tr>
<td></td>
<td>CDC=2.93</td>
<td></td>
<td>[1,4,5,7,9,14]</td>
<td>Choose</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>[10,13,14]</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>[3,6,8,15]</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>2.895</td>
<td>Same as above</td>
<td>-0.02</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2.895</td>
<td>OK</td>
</tr>
<tr>
<td>5</td>
<td>[2,11,12]</td>
<td>* Based on range analysis choose R=2.80</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[1,4,5,7,9,10,13,14]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[3,6,8,15]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MF=2</td>
<td>2.80</td>
<td>[2,11,12]</td>
<td>Low,</td>
</tr>
<tr>
<td></td>
<td>CDC=2.895</td>
<td></td>
<td>[5,7,9,10,13,14]</td>
<td>Choose</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>[3,6,8,15,1,4]</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>2.869</td>
<td>Same as above</td>
<td>-0.01</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2.869</td>
<td>OK</td>
</tr>
</tbody>
</table>

Refer iteration 4, step 2 onwards in RSP #1
### Table 9c

**Iteration Log for the Approximation Procedure**

Number of Parts = 15  
Random Starting Partition = 3

<table>
<thead>
<tr>
<th>ITR No.</th>
<th>INITIAL ALLOCATION</th>
<th>INTERMEDIATE OBJ AND FINAL ALLOCATIONS</th>
<th>Z(R,X)</th>
<th>CDC</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>[1,5,7,9,13,14]</td>
<td>* Based on range analysis choose R=3.10</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[3,6,11,15]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[2,4,8,10,12]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MF=1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CDC=3.157</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3.10 [7,13,14]</td>
<td>7.90 3.124 Low,</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[3,6,11,15]</td>
<td>Choose 3.124</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[2,4,8,10,12]</td>
<td>1,5,9</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3.124 Same as above</td>
<td>0 3.124 OK</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>[7,13,14]</td>
<td>* Based on range analysis choose R=2.90</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[3,6,11,15]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[2,4,8,10,12,1,5,9]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MF=3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CDC=3.124</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.90 [7,13,14]</td>
<td>22.4 2.961 Low,</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[3,6,11,15]</td>
<td>Choose 2.961</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[2,4,8,10,12,1,5,9]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.961 Same as above</td>
<td>0 2.961 OK</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>[1,4,5,7,8,9,13,14]</td>
<td>* Based on the range analysis choose R = 2.825</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[3,6,11,15]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[2,10,12]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MF=1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CDC=2.961</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.825 [1,4,5,8,9,13]</td>
<td>15.56 2.876 Low,</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[3,6,11,15]</td>
<td>Choose 2.876</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[2,10,12,7,14]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.876 Same as above</td>
<td>0.01 2.876 OK</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(Contd.)
<table>
<thead>
<tr>
<th>Itr INITIAL ALLOCATION</th>
<th>INTERMEDIATE OBJ R AND FINAL FN. NEW ALLOCATIONS Z(R,X) CDC Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>[1,4,5,8,9,13] [3,6,11,15] [2,7,10,12,14]</td>
<td>* Based on range analysis choose R=2.876</td>
</tr>
<tr>
<td>3</td>
<td>2.876 Same as 0.02 2.876 OK Initial</td>
</tr>
<tr>
<td>MF=1 CDC=2.876</td>
<td>Family configuration not changed</td>
</tr>
<tr>
<td>Choose the family with next highest $D_k$ i.e., MF=3. Based on the range analysis R=2.80</td>
<td></td>
</tr>
<tr>
<td>2.80 [1,4,5,8,9,12.90 2.875 Low 13] [3,6,11,12,15] 2.875 [2,7,10,14]</td>
<td></td>
</tr>
<tr>
<td>2.875 Same as above -0.01 2.875 OK</td>
<td></td>
</tr>
<tr>
<td>[1,4,5,8,9,13] [3,6,11,12,15] [2,7,10,14]</td>
<td>* Based on range analysis choose R=2.75</td>
</tr>
<tr>
<td>4</td>
<td>2.75 [1,4,5,8,5,9,12.2 2.819 Low, 13,3] [6,11,12,15] 2.818 [2,7,10,14]</td>
</tr>
<tr>
<td>MF=2 CDC=2.875</td>
<td></td>
</tr>
<tr>
<td>2.818 [1,3,4,5,8,9,13] [6,11,12,15] [2,7,10,14]</td>
<td>(Contd.)</td>
</tr>
</tbody>
</table>
TABLE 9c (Continued)

<table>
<thead>
<tr>
<th>Itr</th>
<th>INITIAL ALLOCATION</th>
<th>INT. OBJ FN. AND FINAL ALLOCATIONS</th>
<th>Z(R,X)</th>
<th>CDC</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>[1,3,4,5,8,9,13]</td>
<td>* Based on range analysis choose</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[6,11,12,15]</td>
<td>R=2.725</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[2,7,10,14]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MF=1</td>
<td>2.725 [1,3,4,5,8,13] 23 2.799</td>
<td>Low,</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CDC=2.818</td>
<td></td>
<td>Choose</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>2.799 [6,11,12,15] 2.799</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Same as above 0 2.799 OK</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| 6   | [1,3,4,5,8,13]    | * Based on range analysis choose  |
|     | [6,11,12,15]       | R=2.799                           |        |     |          |
|     | [2,7,9,10,14]     | 2.799 Same as initial 0 2.799 OK  |        |     |          |
|     | MF=1               | Family configuration not changed. |
|     | CDC=2.799          | Choose the family with next highest $D_k$ |
|     |                     | i.e., MF=3. Based on range analysis |
|     |                     | choose R=2.799                    |
|     |                     | 2.799 Same as initial 0 2.799 OK  |        |     |          |
|     |                     | Family configuration not changed. |
|     |                     | Choose MF=2 and based on range analysis |
|     |                     | R=2.799                           |
|     |                     | 2.799 Same as initial 0 2.799 OK  |        |     |          |
|     |                     | Family configuration not changed. |
|     |                     | All families considered for reallocation |
|     |                     | STOP                               |
TABLE 9 d

Iteration Log for the Approximation Procedure

Number of Parts = 15
Random Starting Partition = 4

<table>
<thead>
<tr>
<th>Itr No.</th>
<th>INITIAL ALLOCATION</th>
<th>INTERMEDIATE OBJ AND FINAL FN. NEW ALLOCATIONS</th>
<th>Z(R,X) CDC Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1,5,8,10,15</td>
<td>* Based on range analysis choose R=3.30</td>
<td></td>
</tr>
<tr>
<td></td>
<td>[3,9,7,13,14]</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[2,4,6,11,12]</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MF=2</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CDC=3.44</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3.30</td>
<td>[1,5,8,10,15,  40.5  3.20  R high 7,9,13,14]</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Choose</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>[3]</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>[2,4,6,11,12]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3.20</td>
<td>Same as above</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>3.20</td>
<td></td>
</tr>
<tr>
<td></td>
<td>OK</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1,5,7,8,9,10,13,14,15</td>
<td>* Based on range analysis choose R = 3.05</td>
<td></td>
</tr>
<tr>
<td></td>
<td>[3]</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[2,4,6,11,12]</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MF=1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CDC=3.204</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3.05</td>
<td>[5,7,9,10,13, 15.7  3.105  Low, 14]</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Choose</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>[1,3,8,15]</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>[2,4,6,11,12]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3.105</td>
<td>Same as above</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.02</td>
<td>3.105</td>
<td></td>
</tr>
<tr>
<td></td>
<td>OK</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>5,7,9,10,13,14</td>
<td>* Based on range analysis choose 3.105</td>
<td></td>
</tr>
<tr>
<td></td>
<td>[1,3,8,15]</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[2,4,6,11,12]</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MF=1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CDC=3.104</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3.104</td>
<td>Same as initial 0.02 3.105 OK Family configuration not changed.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Choose the family with next highest D_k i.e., MF=3. Based on the range analysis R = 2.95</td>
<td></td>
</tr>
</tbody>
</table>

(Contd.)
<table>
<thead>
<tr>
<th>Itr No.</th>
<th>INITIAL ALLOCATION</th>
<th>INTERMEDIATE OBJ R AND FINAL FN. Z(R,X) CDC Comments</th>
<th>NEW ALLOCATIONS</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.95</td>
<td>[5, 7, 9, 10, 13, 14]</td>
<td>-31.7 2.846 High, Choose</td>
<td>2.846</td>
</tr>
<tr>
<td></td>
<td>[1, 3, 8, 15, 4]</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[2, 6, 11, 12]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.846</td>
<td>Same as above</td>
<td>-0.01 2.846 OK</td>
<td></td>
</tr>
</tbody>
</table>

Refer to Iteration 5 onwards in RSP #1
TABLE 9

Iteration Log for the Approximation Procedure

Number of Parts = 15
Random Starting Partition = 5

<table>
<thead>
<tr>
<th>ITR</th>
<th>INITIAL Allocation</th>
<th>INTERMEDIATE OBJ R AND FINAL FN.</th>
<th>NEW ALLOCATIONS 2(R,X)</th>
<th>CDC Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>[1, 3, 9, 12, 13]</td>
<td>* Based on range analysis choose R=3.35</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[4, 10, 11, 5, 7]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[2, 6, 8, 14, 15]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MF=2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CDC=3.665</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>3.35 [1, 3, 9, 12, 13, 4, 5, 7, 10, 11]</td>
<td>High, Choose</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>[2, 6, 8, 14, 15]</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>3.209 [1, 3, 9, 12, 13, 7, 10, 11, 4, 5]</td>
<td>High, Choose</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>[2, 6, 8, 14, 15]</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>3.197 Same as above 0 3.197 OK</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>[1, 3, 4, 5, 9, 12, 13]</td>
<td>* Based on range analysis choose R=3.10</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[7, 10, 11]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[2, 6, 8, 14, 15]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MF=1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CDC=3.197</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>3.10 [1, 3, 4, 5, 9, 13, 7, 10, 11, 12, 2, 6, 8, 14, 15]</td>
<td>Low, Choose</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>3.123 Same as above -0.01 3.123 OK</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>[1, 3, 4, 5, 9, 13]</td>
<td>* Based on range analysis choose R=2.95</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[7, 10, 11, 12]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>[2, 6, 8, 14, 15]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MF=3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CDC=3.665</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>2.95 [1, 3, 4, 5, 9, 13, 7, 10, 11, 12, 2, 6, 8, 14, 15]</td>
<td>OK, 2.95</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>[2, 6, 15]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(Contd.)
<table>
<thead>
<tr>
<th>ITR INITIAL NO.</th>
<th>ALLOCATION</th>
<th>INTEmediate OBJ R AND FINAL FN. NEW ALLOCATIONS Z(R,X) CDC Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>[1,3,4,5,8,9,13] [7,10,11,12,14] [2,6,15]</td>
<td>* Based on range analysis choose R=85</td>
<td>2.85 Same as initial 33 2.95 Low, Choose 2.95</td>
</tr>
<tr>
<td>MF=1 CDC=2.95</td>
<td>2.95 Same as above 0.03 2.95 OK Family configuration not changed. Choose family with the next highest Dk, i.e., MF=2. Based on range analysis Take R=2.85</td>
<td></td>
</tr>
<tr>
<td>[1,3,4,5,8,9,13] [7,10,14] [11,12,2,6,15]</td>
<td>2.85 [1,3,4,5,8,9,13] 52 2.86 Low, Choose 2.86</td>
<td></td>
</tr>
<tr>
<td>[1,3,4,5,8,9,13] [7,10,14] [11,12,2,6,15]</td>
<td>2.86 Same as above 0 2.86 OK</td>
<td></td>
</tr>
<tr>
<td>[1,3,4,5,8,9,13] [7,10,14] [11,12,2,6,15]</td>
<td>* Based on range analysis choose R=2.80</td>
<td>2.80 [1,3,4,8,13] 10.4 2.835 Low, Choose 2.835</td>
</tr>
<tr>
<td>4 MF=1 CDC=2.861</td>
<td>2.835 [1,3,4,5,8,13] -0.35 2.835 OK</td>
<td></td>
</tr>
<tr>
<td>[1,3,4,5,8,13] [7,10,14,9] [11,12,2,6,15]</td>
<td>* Based on the range analysis choose R = 2.775</td>
<td>2.775 [1,3,4,5,8,13] 7.6 2.799</td>
</tr>
<tr>
<td>5 MF=3 CDC=2.835</td>
<td>2.775 [7,9,10,14,2] [6,11,12,15]</td>
<td></td>
</tr>
</tbody>
</table>

Refer to iteration 6 of RSP #3
VITA AUCTORIS

1958  Born in Hosmat, India, on 14th of December.

1974  Completed higher secondary education from
      Government High School, Kadaba, India.

1976  Completed Pre-University Course from Sri.
      Bhuvanendra College, Karkala, India.

1981  Graduated from The National Institute of
      Engineering, Mysore, affiliated to the
      University Of Mysore, India with a Bachelor’s
      degree in Mechanical Engineering.

1981-84  Worked as an Industrial Engineer in the
         departments of Management Services and
         Industrial Engineering at Bharat Electronics
         Ltd., Bangalore, India.

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