



National Library  
of Canada

Acquisitions and  
Bibliographic Services Branch

395 Wellington Street  
Ottawa, Ontario  
K1A 0N4

Bibliothèque nationale  
du Canada

Direction des acquisitions et  
des services bibliographiques

395, rue Wellington  
Ottawa (Ontario)  
K1A 0N4

*Your file* *Votre référence*

*Our file* *Notre référence*

## NOTICE

The quality of this microform 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.

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

## AVIS

La qualité de cette microforme 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.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

Canada

# **STATIC & DYNAMIC CONSIDERATIONS OF PART FAMILY & MACHINE CELL FORMATIONS**

*A THESIS SUBMITTED TO  
THE FACULTY OF GRADUATE STUDIES AND RESEARCH  
IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE  
DEGREE OF MASTER OF APPLIED SCIENCE IN  
MECHANICAL ENGINEERING*

**BY  
TÜRKER ÜLGER**

Ottawa-Carleton Institute for  
Mechanical and Aeronautical Engineering  
University of Ottawa  
Ottawa, Ontario  
CANADA, K1N 6N5



National Library  
of Canada

Acquisitions and  
Bibliographic Services Branch

395 Wellington Street  
Ottawa, Ontario  
K1A 0N4

Bibliothèque nationale  
du Canada

Direction des acquisitions et  
des services bibliographiques

395, rue Wellington  
Ottawa (Ontario)  
K1A 0N4

*Your file* *Votre référence*

*Our file* *Nôtre référence*

The author has granted an irrevocable non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.

L'auteur a accordé une licence irrévocable et non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without his/her permission.

L'auteur conserve la propriété du droit d'auteur qui protège sa thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

ISBN 0-315-83843-4

Canada



UNIVERSITÉ D'OTTAWA  
UNIVERSITY OF OTTAWA

TO *MY FAMILY*

Suzan and Neşet ÜLGER

Fusun and Yunus ERDEM

Emre ÜLGER

## ABSTRACT

The dynamic structure of part-mix in cellular manufacturing systems brings out volume/capacity imbalance and consequently frequent reallocation problems, which cause loss of time, production and money. Consideration of part-mix and demands for multi-period according to forecasts and/or product life-cycles at the design stage of cellular manufacturing systems would reduce the costs of time and investment, as long as, there is no rapid unexpected fluctuations at part demand curves.

In this research, a new multi-period strategy is suggested for machine and part family formation process, and compared with two other old common strategies. Single-and multi-period mixed integer mathematical models for simultaneous machine/component grouping and assignment are developed as a tool of comparison. The models consider the trade-offs of cell configuration, machine procurement and salvage, part subcontracting, inter-cell movement, capital investment costs, which reflect the significance of real life planning aspects. The heuristic method of machine cell and part family formations under the criteria of maximum cell similarity and minimum machine number is proposed. This heuristic method is integrated with a mathematical program, which optimizes various cost aspects into a two-stage model. The heuristic method, as well as the mathematical program are modified to take into consideration the multi-period part family/machine cell configuration problem.

Possible advantages and disadvantages of various multi-period part/machine grouping strategies and their associated costs are investigated by using the developed heuristic method and mathematical programming models. Three and five-period example problems are illustrated to compare various planning strategies with varying cost values. Behaviour of models with changing cost values are also discussed. The importance of multi-period planning consideration during cell creation phase is verified with the proposed multi-period strategy.

## ACKNOWLEDGEMENT

I would like to express my sincere appreciation to my supervisor Dr. S.M. Taboun, and co-supervisor Dr. B. Dhillon, for their advice, criticisms, patience, and moral and financial support during the course of this research. My regards go to Dr. M. Liang for his generous support in terms of providing software and computer facilities.

I am grateful to the following people for their unbroken moral support (in alphabetic order): Mr. N. Alpan, Dr. N. Dai, Mr. S. Dai, M.D. A. Demircan, Dr. M. Dicleli, Mrs. B. Leitgeb, Mr. S. Kınıklıođlu, Mr. F. Kolahan, Mrs. K. Lorette, Mrs. N. Mandıralıođlu and Mr. C. Yalçın.

Lastly, my appreciations go out to Mr. D. Seaman, and the support staff of the Mechanical Engineering Department for their cooperation.

## TABLE OF CONTENTS

<b>ABSTRACT</b> .....	i
<b>ACKNOWLEDGEMENT</b> .....	ii
<b>TABLE OF CONTENTS</b> .....	iii
<b>LIST OF FIGURES</b> .....	vi
<b>LIST OF TABLES</b> .....	vii
<b>CHAPTER 1 INTRODUCTION</b> .....	1
<b>CHAPTER 2 LITERATURE REVIEW</b> .....	6
2.1 Classification and Coding Techniques .....	6
2.2 Production Flow Analysis .....	7
2.3 Graph Theoretic Models .....	12
2.4 Mathematical Programming Formulation .....	14
<b>CHAPTER 3 MODEL AND STRATEGY DEVELOPMENT</b> .....	20
3.1 General .....	20
3.2 Single Period PF/MCC .....	21

3.2.1	Assumptions .....	21
3.2.2	Notations .....	21
3.2.3	Developing objective function .....	25
3.2.4	Developing Constraint sets .....	29
3.3	Two-Stage Single-Period PF/MCC Model .....	33
3.3.1	Maximum Similarity-Minimum Machines (MS-MM) Method .....	35
3.3.2	Development of System Weighted Similarity Function.	42
3.3.3	Modified Single Period Model .....	44
3.4	Model Developments for Multi-Period Strategies .....	47
3.5	Multi-Period Pf/MCC Model .....	49
3.5.1	General assumptions .....	49
3.5.2	Additional notations .....	50
3.5.3	Developing objective function .....	52
3.5.4	Developing constraint sets .....	54
3.6	Two-Stage Multi-Period PF/MCC Model .....	56
3.6.1	Modified MS-MM Algorithm .....	57
3.7	Multi-Period Planning Strategies .....	61
3.7.1	Strategy I .....	61
3.7.2	Strategy II .....	63
3.7.3	Strategy III .....	64
 <b>CHAPTER 4 COMPUTATIONAL RESULTS .....</b>		<b>65</b>
4.1	General .....	65
4.2	Capability of MS-MM Method .....	65
4.3	Testing and Evaluation of Two-Stage Models .....	70
4.3.1	Two-Stage Single-Period PF/MCC model testing .....	70
4.3.2	Two-Stage Multi-Period PF/MCC model testing .....	73

4.4	Strategy Evaluation .....	75
4.4.1	Applications for 3-Period .....	76
4.4.2	Applications for 5-Period .....	84
4.5	Test of Two-Stage Multi-Period Pf/MCC Model .....	94
<b>CHAPTER 5 CONCLUDING REMARKS &amp; FUTURE RESEARCH .</b>		<b>100</b>
5.1	Concluding Remarks .....	100
5.2	Future Research .....	102
<b>REFERENCES .....</b>		<b>104</b>
<b>APPENDIX I</b>	<b>MS-MM METHOD .....</b>	<b>109</b>
<b>APPENDIX II</b>	<b>FIRST STAGE OF TWO-STAGE SINGLE-PERIOD PF/MCC MODEL .....</b>	<b>118</b>
<b>APPENDIX III</b>	<b>FIRST STAGE OF TWO-STAGE MULTI-PERIOD PF/MCC MODEL .....</b>	<b>130</b>
<b>APPENDIX IV</b>	<b>EXAMPLE LINDO INPUT FILES FOR SINGLE- PERIOD PF/MCC MODEL AND TWO-STAGE SINGLE-PERIOD PF/MCC MODEL .....</b>	<b>142</b>
<b>APPENDIX V</b>	<b>EXAMPLE LINDO INPUT FILES FOR MULTI- PERIOD PF/MCC MODEL AND TWO-STAGE MULTI-PERIOD PF/MCC MODEL .....</b>	<b>146</b>

## LIST OF FIGURES

<b>Figure 3.1</b>	<b>Two-Stage Single-Period PF/MCC Model .....</b>	<b>34</b>
<b>Figure 3.2</b>	<b>The main body of MS-MM Method .....</b>	<b>39</b>
<b>Figure 3.3</b>	<b>The Procedure A of MS-MM Method .....</b>	<b>40</b>
<b>Figure 3.4</b>	<b>The procedure B of MS-MM Method .....</b>	<b>41</b>
<b>Figure 3.5</b>	<b>Two Stage Multi-Period Algorithm .....</b>	<b>57</b>
<b>Figure 3.6</b>	<b>Planning Processes of Strategies .....</b>	<b>62</b>
<b>Figure 4.1</b>	<b>Evaluation of Strategies for 3-Period .....</b>	<b>83</b>
<b>Figure 4.2</b>	<b>Example Product Demands in Five Periods .....</b>	<b>86</b>
<b>Figure 4.3</b>	<b>Evaluations of Strategies for 5-Period .....</b>	<b>90</b>
<b>Figure 4.4</b>	<b>Impact of Strategies on Cell and System Utilizations .....</b>	<b>93</b>
<b>Figure 4.5</b>	<b>Impacts of different Inter-cell Movement Cost on system ....</b>	<b>96</b>
<b>Figure 4.6</b>	<b>Savings of Multi-Period Model .....</b>	<b>99</b>

## LIST OF TABLES

<b>Table 3.1</b>	<b>Model Sizes .....</b>	<b>60</b>
<b>Table 3.2</b>	<b>Numerical Example for Model Sizes .....</b>	<b>60</b>
<b>Table 4.1</b>	<b>Burbidge's Initial Matrix .....</b>	<b>68</b>
<b>Table 4.2</b>	<b>Comparisons of Burbidge's Manual Method and MS-MM Method .....</b>	<b>69</b>
<b>Table 4.3</b>	<b>Comparison of ROC and MS-MM SolUtions .....</b>	<b>69</b>
<b>Table 4.4</b>	<b>Part Demands and Processing Requirements .....</b>	<b>71</b>
<b>Table 4.5</b>	<b>Cost Parameters and Planning Horizon .....</b>	<b>71</b>
<b>Table 4.6</b>	<b>Results of MS-MM Method .....</b>	<b>72</b>
<b>Table 4.7</b>	<b>Results of Single-Period Models .....</b>	<b>72</b>
<b>Table 4.8</b>	<b>Comparison of Single-Period Models .....</b>	<b>72</b>
<b>Table 4.9</b>	<b>Unit Processing Times .....</b>	<b>73</b>
<b>Table 4.10</b>	<b>Cost Parameters and Planning Horizon .....</b>	<b>73</b>
<b>Table 4.11</b>	<b>Results of 2-Stage Algorithm and Multi-Period Model .....</b>	<b>74</b>
<b>Table 4.12</b>	<b>Results of Multi-Period Models .....</b>	<b>75</b>
<b>Table 4.13</b>	<b>Comparison of Multi-Period Models .....</b>	<b>75</b>
<b>Table 4.14</b>	<b>Outline of Applications .....</b>	<b>77</b>
<b>Table 4.15</b>	<b>Part Demands for 3 Periods .....</b>	<b>78</b>
<b>Table 4.16</b>	<b>Machine Requirements and Processing Times of Parts on Machines .....</b>	<b>78</b>
<b>Table 4.17</b>	<b>Results of MS-MM Algorithm for 3-Period Application .....</b>	<b>79</b>
<b>Table 4.18</b>	<b>Results of 1<sup>st</sup> Stage of 3-Period Application .....</b>	<b>79</b>
<b>Table 4.19</b>	<b>Results of STRATEGY I .....</b>	<b>80</b>

<b>Table 4.20</b>	<b>Results of STRATEGY II .....</b>	<b>80</b>
<b>Table 4.21</b>	<b>Results of Strategy III .....</b>	<b>81</b>
<b>Table 4.22</b>	<b>Strategy Comparisons for 3-Period Example .....</b>	<b>82</b>
<b>Table 4.23</b>	<b>Part Demands for 5 Periods .....</b>	<b>85</b>
<b>Table 4.24</b>	<b>Results of First Stage of Two-Stage Multi-Period PF/MCC Model .....</b>	<b>87</b>
<b>Table 4.25</b>	<b>Results of STRATEGY I .....</b>	<b>87</b>
<b>Table 4.26</b>	<b>Results of STRATEGY II .....</b>	<b>88</b>
<b>Table 4.27</b>	<b>Results of Strategy III .....</b>	<b>88</b>
<b>Table 4.28</b>	<b>Strategy Comparison for 5-Period Applications .....</b>	<b>89</b>
<b>Table 4.29</b>	<b>Evaluation of Strategies for 5-Period Applications .....</b>	<b>92</b>
<b>Table 4.30</b>	<b>Results of Applications with IMC 1/5 - 1/10 - 1/20 .....</b>	<b>95</b>
<b>Table 4.31</b>	<b>Savings of Multi-Period PF/MCC Model .....</b>	<b>99</b>

# **CHAPTER 1**

## **INTRODUCTION**

Cellular Manufacturing System (CMS) is the result of direct application of Group Technology (GT) philosophy. Parts that have similar processing requirements, such as machines, tools, routes and/or geometrical shapes are classified into part families. Machine cells contain groups of functionally dissimilar machine types, and at least one part family is assigned to a machine cell. Reorganization of the functional shop into manufacturing cells leads to several benefits which reduce the production costs of manufacturing parts. It also reduces the set-up and throughput times, work-in-process inventories, materials handling, and improves machine utilization.

Part families and machine cell formations are the backbone of the structural design phase of manufacturing cells. Major steps in this phase include the identification of parts and their process populations, machines, tools, fixtures, pallets and material

handling equipment. Other key variables in designing cellular manufacturing systems may include number of cells, machine types, different part types, number of operations per part, alternative process plans and routings for each part, cell size and composition, and part characteristics. In many situations, it is necessary to trade-off certain objectives related to structural design parameters and system performance variables (Wemmerlow and Hyer, 1986).

A fundamental problem associated with cellular manufacturing is the determination of part families and machine cells, such that one or more families can be fully processed within a single cell, which is designated as "machine-component grouping". A significant effort has been directed toward part family and cell formation problems, and several approaches were utilized. Part families and machine cells were formed either sequentially or simultaneously while considering various manufacturing costs directly or indirectly and system constraints. Although each approach has its advantages and disadvantages, an effective plan must be determined prior to selecting any technique.

Many of the existing design and planning strategies do not take into consideration the possibility of changing workloads allocated to each machine cell as well as changes in the part-mix of future production plans. In other words, existing design and planning strategies lack the flexibility of absorbing fluctuations in the production workloads and future part-mix changes. This leads to a greater imbalance in the

machines and machine cells workloads, and reduces their utilizations. Furthermore, inter-cell material handling problems would arise, which might lead to a greater work-in-process, production delays and longer throughput times. The most common pitfalls during the planning process were summarized as follows (Stephanou and Spiegl, 1992):

- The short-range view and concern of top management with the immediate profit and loss picture, preventing adequate funding and consideration of the longer range requirements; goes hand in hand with sacrificing long-term economic gains and product viability for short term gains.
- Failure to take even small risks and the "wait-and-see" attitude that invariably places the company in a catch-up mode relative to more versatile and mobile completion.
- Lack of follow-up in the form of actions and a meaningful control process throughout the product cycle.
- Capacity planning is the activity whose primary goal is to ensure sufficient processing capability within an organization to accomplish the required production within the desired time.

"A five-year plan must be developed for products, volumes, and variations, key business drivers, such as reducing product costs must be identified; and these goals should mesh with corporate strategy" (Baran, 1991). Producing 3-year or 5-year plans simply by extrapolating or pushing forward the current operations and stressing

the numbers rather than thoughtfulness in planning are also recommended by Canada and Sullivan (1989).

As mentioned by J. Baran (1991), product strategy is an important part of the action's plan to cover as well as, quality demands, technology assessment, business goals. Present and future products and their expected volumes and variations regarding product life cycles must be considered for preplanning of GT cells to prevent or reduce the problems as much as possible. In most cases, products tend toward a life cycle consisting of introduction, growth, maturity and decline stages. Sales grow slowly during the introduction stage, rapidly during the growth stage, and level off in maturity stage. In the decline stage, sales of the product fall. The length of the life cycle and the length of the individual phases vary for different products. The shape of the curve is extremely important in major business decisions concerning current products and also development of new product lines (Canada and Sullivan, 1989).

Designing manufacturing cells while considering the possible highest demand of parts can not be a good solution, since system utilization would decrease remarkably, even if it increases the flexibility of system regarding part routings. Consideration of multi-period planning horizon would make the models and algorithms more sensitive to fluctuating part demand curves, and reduce the costs resulting from the problems mentioned above.

The objectives of this study are four-fold:

- i. To develop single and multi-period mathematical programming models which simultaneously form the optimal part family and machine cell configurations.
- ii. To develop a two-stage model using single-period planning data to determine the optimal part-family and machine cell configurations.
- iii. To develop a two-stage model using multi-period planning data to determine the optimal part-family and machine cell configurations.
- iv. To evaluate the effect of static, dynamic, and mixed planning strategies (based on single and multi-period models in ii and iii) on part family and machine cell configurations considering various cost trade-off techniques.

Following this chapter the thesis is organized as follows: Literature review on various aspects of design and planning approach in cellular manufacturing systems is presented in Chapter 2. The development of single and multi-period mathematical programming as well as two-stage models are presented in Chapter 3. Also the behaviour of proposed heuristic and mathematical models are examined. In Chapter 4, illustrative applications using various models with 3 and 5 planning periods are solved to observe the influence of multi-period planning strategy on cell configuration. Discussion and comparison of planning strategies regarding arising costs and system utilization are also presented. Conclusions and recommendations for future research are presented in Chapter 5.

# **CHAPTER 2**

## **LITERATURE REVIEW**

Part family and machine cell formations can be determined using various strategies and techniques. As indicated earlier, parts and machines are clustered into groups either sequentially or simultaneously. All of the methods and models presented in this chapter and literature consider only one period as a planning horizon for machine / component grouping and assignment problem. The review of literature of various clustering approaches of manufacturing cells is divided into four categories presented in the subsequent sections. It includes classification and coding, production flow analysis, graph theoretic models and mathematical programming formulations.

### **2.1 Classification and Coding Techniques**

The classification and coding methods, group parts into part families according to their design and geometric features. One of the simplest techniques is the visual

classification method. In this case, parts are grouped according to the similarity of their geometric shapes such as rotational and prismatic parts. A significant variation may result from employing this method, because it is highly subjective and depends on personal selection. As the number of manufacturing part increases, it becomes very difficult to handle them using this technique. Obviously, this is one of the major drawbacks of this technique.

Geometric shape and complexity, dimensions, type of material, shape of raw material and required accuracy of the finished parts are the features of the classification and coding methods of manufacturing parts. A numerical or alphabetical code is assigned to each part. Commercial coding schemes such as BRISCH, MICLASS, CODE are widely available as well as non-proprietary coding schemes such as KC-1 and OPITZ. One of the major disadvantages of these schemes is their inability to group parts which are dissimilar in design features but are processed on the same set of machines (Burbidge, 1979). Although the classification and coding techniques have several advantages, a significant amount of data input is required, which makes it undesirable in many practical situations.

## **2.2 Production Flow Analysis**

Burbidge (1971), developed Production Flow Analysis (PFA) to form the production

cells in such a way that each part is completely processed within a production cell. PFA was extended to consider cellular subsystems formation for assembly countenance of production process (De Beer and de Witte, 1978). The method was called Production Flow Synthesis (PFS) and it works well for small workshops perform a large number of operations (Wemmerlow, 1984). The Flexible Production Cells (FPC) was developed by Tilsley and Lewis (1977), considers the demand fluctuations of parts. The cells are formed ensuring that the highest demand of each part can be performed within a cell. This consideration brings a high degree of scheduling flexibility with low utilization. El-Essawy (1972) proposed Component Flow Analysis (CFA), which is similar to PFA in some aspects.

Production flow analysis has been successfully used to develop models and algorithms using information from the machine-component incidence matrix. Various similarity measures were used in the development of heuristic models to form part families and machine cells. The similarity coefficient indices plays an important role in increasing the models capability to improve the configurations of manufacturing cells.

One of the early similarity indices used to group machines is the single linkage cluster analysis (SLCA) introduced by McAuley (1972), which is based on Jaccard's similarity coefficient. The index uses pairwise comparisons to calculate similarities between machines. Moiser (1989) indicates the McAuley's coefficient to be one of the preferable coefficient with the modified multiplicative weighted similarity

coefficient. Carrie (1975) used a similar coefficient as McAuley (1972) used, except capturing the similarity between parts instead of machines. SLCA method does not realize the chaining problem resulted from the duplication of bottleneck machines.

Seifoddini and Wolfe (1986) developed the average linkage clustering (ALC) algorithm to overcome the chaining problems. The similarity coefficient between any two clusters is defined as average of the similarity coefficient between all elements of the two clusters. The machine-part incidence matrix was represented by using a binary machine code. Weighted Average Linkage (WLINK) is an extension of ALC. The average similarity coefficient of ALC is weighted over by the sizes of the two member clusters.

A number of similarity measures based on machine-component matrix information were introduced to develop clustering algorithms such as absolute similarity coefficient, mutual similarity coefficient and absolute similarity coefficient (De Witte, 1980), processing sequence similarity coefficient (Vakharia and Wemmerlow, 1985), the pairwise proximity measure based on manufacturing operations (Choobineh, 1988). Production data based similarity coefficient using part routing sequence, unit operation time and part production volume, as a measure of association between machines (Gupta and Seifoddini, 1989).

McCormick et al. (1972) introduced an interchange clustering algorithm, the bond-

energy algorithm (BEA). The BEA seeks to form a block diagonal form by maximizing the measure of effectiveness (ME). Slagle et al. (1975) developed a clustering algorithm based on BEA and the shortest spanning path (SPS). Bhat and Haupt (1976) extended the idea of Slagle et al. (1975). Chu and Tsai (1990) mentioned that BAE algorithm performs better than ROC and DCA, especially while dealing with exceptional elements.

King (1980) developed the Rank Order Clustering (ROC) algorithm, where clusters are identified visually in the final generated matrix, and the result is dependent on the initial nature of the machine/part matrix and usage of binary value which is a power of 2 limits the size of the problem technically. To overcome these disadvantages, King and Nakornchai (1982) developed ROC2 and then Rajagopalan and Chandrasekharan (1986) extended ROC algorithm by integrating "block and slice" method and the hierarchical clustering method, which is called MODROC.

Chan and Milner (1982) developed the direct cluster algorithm (DCA) based on sorting. DCA algorithm rearranges the rows and columns. Wemmerlow (1984), revised the algorithm to handle large size problems. DCA counts the number of positive cells which makes result more sensitive and independent from initial matrix.

Kusiak and Chow (1987a) developed the cluster identification (CI) algorithm, based on the concept presented in Iri (1968). Kusiak and Chow (1987b) extended the CI

algorithm to solve GT problem with known subcontracting cost, which is called the cost analysis algorithm. The algorithm minimize the total sum of subcontracting costs, while the overlapping parts removed from the incidence matrix. Kusiak and Chenk (1990) introduced a branch-and-bound algorithm for solving group technology problem, with bottleneck parts and machines. The algorithm is based on removing parts from a machine-part incidence matrix.

Peihua Gu (1991) proposed process-based machine grouping method for cellular manufacturing systems. The approach of Peihua Gu is in two stages. A clustering algorithm for part families grouping based on process similarities, and cells formation based on the similarities of the machine's operational functionalities, and formed part families are the two stages of Peihua Gu's approach. Two 0-1 incidence matrices, one of which contains information of components and process relations and the other machine-component relations.

Logendran (1990) and Logendran et al.(1990) developed algorithms for part-machine grouping, while minimizing total inter-cell and intra-cell movements. Minis et al.(1990) introduced a method considering set-up and run times for the evaluation of the capacity requirements, and uses pallet traffic as opposed to individual part traffic in the minimization criterion. Hilger (1991) presented an approach with the objective of minimizing inter-cell traffic while forming manufacturing cells. The approach aims to concentrate as many operations related to a part family as possible in the same cell.

Okokbaa et al. (1992) developed a new inter-cell flow reduction heuristic, in which the identification of part families, identification of cell equipment, and the allocation of families to equipments are involved.

Nagi et al. (1990) introduced a new algorithm considering both the manufacturing system as well as projected production, and distributing the demand among alternate routings to obtain better manufacturing cell design. Routing selection and cell formation are the problems simultaneously addressed in the algorithm with the common objective of minimizing inter-cell traffic in the system.

### **2.3 Graph Theoretic Models**

Rajagopalan and Batra (1975) presented a graph-partitioning approach for forming cells. The situation is represented in the form of a graph whose vertices correspond to the machines and whose edges represent the relationships created between the machines by the components using them. The method uses the similarity coefficient of McAuley (1972).

Bhat and Haupt (1975) introduced a method using Matching Algorithm and clustering model similar to the Bond Energy Algorithm (BAE). Vohra et al. (1990), presented a non-heuristic network approach to form manufacturing cells with

minimum inter-cell movements. The method uses Gomory-Hu algorithm (Gomory and Hu, 1971) to find a minimum intercellular interaction.

Boe and Cheng (1991) developed a close neighbour algorithm to overcome the deficiencies of clustering, data organization and array sorting methods, such as difficulty of visual identification of machine groups. The method always gives a block diagonal solution matrix. It can deal with exceptional elements without identification.

Lee and Hwang (1991) proposed a hierarchical divisive clustering method, based on graph theory for machine-component grouping problem. The method increases the degree of interrelations between components in the same cell while decreasing for components of different part families. The method uses Minimum Spanning Tree by applying Prim's Algorithm.

Askin et al. (1991) developed a Hamiltonian path approach to reordering the part-machine matrix. Pairwise comparisons are used to compute similarity coefficients and form a distance measure for machines and parts. The authors indicated that this technique overcomes many of the shortcomings of ROC2 Method, which are related to solution convergence and exceptional elements; hence, it leads to improved solutions.

Zhang and Wang (1991) used fuzzy set theory for concurrent formation of part families and machine cells.

## **2.4 Mathematical Programming Formulation**

Many researchers have formulated the part family and machine cell problem as 0/1 integer mathematical program. Although 0/1 mathematical programming technique has some disadvantages in terms of computational times, advances in computer technology and mathematical programming software eliminate some of these disadvantages. Literature reveals that various objective functions are formulated based on similarity measures, cost functions, and other performance measures.

Although there exists some limitations to solve large scale problem of machine cell formation, mathematical programming techniques provide a useful tool to obtain optimal solutions.

Kusiak (1985) considered an integer programming formulation of the clustering problem, known as p-median model. The number of part families (clusters) are specified in p-median formulation with the objective of maximizing total part similarities as determined from pairwise comparison of 0/1 vectors of the incidence matrix. The p-median model was extended to deal with alternative process plans for each part type (Kusiak, 1987). The objective is to minimize the total sum of distance

measures and production costs. The problem was also formulated as a 0-1 quadratic programming model to deal with the restricted number of clusters and cluster sizes (Kusiak et al., 1986). The model was solved by an eigenvector-based algorithm.

Choobineh (1988) proposed a two-stage procedure. The first stage forms the part families by using clustering techniques. The proximity measure uses the manufacturing operations and operations sequences. The second stage forms the machine cells with an integer programming model. The objective function is the sum of production costs of acquiring and maintaining the machine tools. The model considers cell capacity , budget restrictions.

Co et al. (1988) presented three stage procedure for cellular manufacturing system configuration. At the first stage, the mathematical model minimizes the deviation between available capacity and the workload assigned to each machine. There is no constraint except available machining time. Second stage is extended King's algorithm to group machines based on similarities of operations, and third stage is a direct-search algorithm for defining the composition of manufacturing cells.

Gunasingh at al.(1989) presented two 0-1 integer programs with a sequential modelling approach to cell formation problem. The machines are grouped into cells based on their similarity in parts processing, then the parts are allocated to appropriate machine groups based on processing requirements.

Sundaram (1989) considered alternate routing for designing cellular manufacturing systems. The method selects the machines in which a particular operation will be performed to minimize the total capital investment costs on machining centers. Part demands and available machining times are under consideration. The limitation of the model is only two cell consideration.

Sankaran et al. (1990) developed two mathematical models for cell formation and routing selection. The cells are formed in order to maximize the routing flexibility of the system, and to minimize the annual machine usage and processing costs. In The problem was also formulated as a goal programming model, where the objective function is the sum of linear cost functions, including, sum of the capital costs for machines, machining costs, total tool usage costs, all other processing costs, material handling cost (Sankaran, 1990). It also considers process and tool similarity between parts with the limitations of available machining time, part movement and the number of parts assigned to each cell.

Rajamani et al. (1990) proposed three mathematical models to analyze how alternative process plans influence the resource utilization when the part families and machine groups are formed simultaneously. Model 1 assigns machines to parts. The objective function is to minimize the total investment under the machine capacity and budgeted constraint. Model 2 assumes that part families are known. It selects a process plan for each part, machine type for each operation and number of machines of each type in

different cells. The objective function is to minimize the total investment on machines of different types assigned to all the cells under the same constraints as model 2. Model 3 identifies part families and machine groups simultaneously. The objective function and limitations are basically the same.

Srinivasan et al. (1990) developed an assignment model for part-families problem, with the objective of minimizing exceptional parts and machine idling. The similarity coefficient matrices for machines and components with identified preliminary machine cells and part families are used as inputs, to merge and get final solution.

Logendran (1991) observed effect of the identification of key machines in the cell formation problem of cellular manufacturing systems, and proposes one mathematical model and two algorithms. Mathematical model minimizes the total number of inter- and intra-cell moves and considers just available capacity of machines.

Jain et al.(1991) proposed a 0-1 integer programming model for the combined problem of forming cells and providing tools in a flexible manufacturing system. The objective of the problem is to minimize the overall system cost defined as the sum of annual cost of processing the parts, cost of tools and annualized cost of machines. Tooling requirements, tool lives, available processing time on a machine are considered in this model.

Gunasingh et al. (1991) proposed two non-linear 0-1 integer programming formulations to group machines and parts in cellular manufacturing systems simultaneously based on the tooling requirements of the parts, tools available on the machines and processing times. The first model forms machine-part groups according to the compatibility of the parts with the machines. The second one does the same work, while seeking a trade-off between the cost of duplicating the machines and the cost of inter-cell movement. The formulations consider the limitations on the number of parts and machines in a group as well as number of machine types available.

Taboun et al. (1991) introduced a weighted similarity index and incorporated in mathematical programming model to form part families and machine cells. It is a mixed (0-1) non-linear program with a quadratic objective function. The three similarity measures suggested by K. Richardson et al. (1990), based on processing machines, sequence of operations and tools required are considered. The proposed model considers just available resources such as the available machining capacity in total.

Liang and Taboun (1992) presented a model to address the part selection and assignment problems simultaneously for existing layouts, while dealing with the exceptional parts and inter-cell movements. The objective function is maximizing total profit while minimizing inter-cell movement, machining and high penalty costs toward the unsatisfied due dates. The model considers part demands, machine

capacities.

Shafer et al. (1992) presented a mathematical programming model that deals with exceptional elements which are bottle neck machines and exceptional parts. The model considers intercellular transfer, machine duplication, and subcontracting costs. Set of exceptional parts and bottleneck machines required in cells annual demand of parts and annual capacities of machines are known in advance.

# **CHAPTER 3**

## **MODEL AND STRATEGY DEVELOPMENT**

### **3.1 General**

In this chapter, two mathematical models are developed. The single-period planning problem of simultaneous part family and machine formulation is formulated as a mixed 0/1 integer program. Similarly, the multi-period planning case is formulated as a mixed 0/1 integer program. In addition, two-stage models are developed to deal with large scale problem of both cases. In both case, the first stage consists of partially modified mathematical program, which interfaced with the heuristic algorithm. All models are described in subsequent sections. Various planning strategies are developed using the single and multi-period models. Comparisons of these strategies are presented at the end of this chapter.

## 3.2 Single-Period PF/MCC Model

This model is developed to form machine groups and part families simultaneously in a cellular manufacturing system while considering main cost groups regarding design and implementation stages of the system for a single period. The following assumptions and notations are used to develop the model.

### 3.2.1 Assumptions

- i. Production information of all parts, such as part demand, processing times, required machines, available machining capacities and various costs are known.
- ii. Inter-cell movements are allowed.
- iii. Subcontracting of any part type is possible, but the whole demand of a particular part must be subcontracted.
- iv. There is no alternative process plans.

### 3.2.2 Notations

#### i. Indices

i                    Part type index,                     $i=1,2,\dots,N$

<b>j</b>	Part family / cell index,	$j=1,2,\dots,N$
<b>k</b>	Machine index,	$k=1,2,\dots,K$
<b>l</b>	Similarity index,	$l=1,2,\dots,L$

## ii. Parameters

<b>AC<sub>k</sub></b>	Average inter-cell movement cost per unit processing time on machine type k.
<b>APT<sub>k</sub></b>	Average processing time per part on machine type k.
<b>CC</b>	Cell configuration cost, including the cost of lost production incurred during cell configuration.
<b>CIM</b>	Average inter-cell movement cost.
<b>D<sub>i</sub></b>	Annual demand of part type i.
<b>EM<sub>k</sub></b>	Existing number of machine type k in whole system.
<b>F<sub>i</sub></b>	Set of families that part i may belong to.
<b>F<sub>j</sub></b>	Set of machines that part family j may require.
<b>F<sub>k</sub></b>	Set of parts that require machine type k.
<b>IC<sub>k</sub></b>	Idle time cost for machine type k per unit time.
<b>MAXC</b>	Maximum number of part families allowed in the system.
<b>MAXP</b>	Maximum number of parts can be assigned to a part family.
<b>MC<sub>k</sub></b>	Annual capital cost for machine type k.

<b>MINC</b>	Minimum number of part families/cells allowed in the system.
<b>MINP</b>	Minimum number of parts can be assigns to a part family.
<b><math>NPM_k</math></b>	Average number of parts per movement to machine type k.
<b>NC</b>	Determined number of cells in the system.
<b><math>PC_k</math></b>	Procurement cost associated with buying one machine of type k, excluding purchasing price.
<b><math>PT_{ik}</math></b>	Total processing time required on machine type k for part type i.
<b><math>SC_k</math></b>	Cost associated with selling machine type k excluding selling price.
<b><math>S_{ij}</math></b>	Similarity coefficient between part i and j.
<b>SL</b>	Required minimum average similarity level (cut-off level).
<b><math>SUC_i</math></b>	Cost of subcontracting part i.
<b><math>TMC_{kj}</math></b>	Total machining capacity for machine type k in cell j.
<b>Z</b>	Total cost.
<b><math>Z_1</math></b>	Total cell configuration cost.
<b><math>Z_2</math></b>	Total annual machine capital investment cost.
<b><math>Z_3</math></b>	Total cost associated with procurement and sales of machines.
<b><math>Z_4</math></b>	Total idle time cost.
<b><math>Z_5</math></b>	Total inter-cell movement cost.
<b><math>Z_6</math></b>	Total subcontracting cost.

### iii. Decision variables

#### a. 0/1 Integer variables

$$X_{ij} = \begin{cases} 1 & \text{if part } i \text{ belongs to part family } j, \\ 0 & \text{otherwise.} \end{cases}$$

$$S_i = \begin{cases} 1 & \text{if part } i \text{ is subcontracted,} \\ 0 & \text{otherwise.} \end{cases}$$

#### b. General integer variables

**CN**            Number of formed part families/machine cells.

**NUM<sub>k</sub>**        Number of machine type k will exist.

**PUR<sub>k</sub>**        Number of new machine type k.

**SEL<sub>k</sub>**        Number of machine type k to sell.

**Y<sub>kj</sub>**        Number of machine type k required in cell j.

#### c. General variables

**AMC<sub>kj</sub>**        Available machining capacity for machine type k in cell j, excluding the processing capacity needed for part family j.

**ID<sub>k</sub>**        Total idle time for machine k(s).

**M<sub>kj</sub>**        Process time allocated out of dedicated cell j for machine k.

### 3.2.3 Developing objective function

The objective of the single-period planning model is to minimize various cost components associated with the configuration design of manufacturing cells. There are six cost components considered in the objective function of the single-period model.

The first cost component is the total cell configuration cost. This cost includes the cost of installing and/or relocating various machines, handling systems, extensive air, oil and water lines to form manufacturing cells. The cost also associated with production loss due to cell reconfiguration and production interruption due to installation activities. Although manufacturing cells may be of different sizes, it is assumed that a reasonable average cost of cell configuration, including above listed costs can be determined. Therefore, the total configuration cost of manufacturing cell,  $Z_1$  is as follows:

$$Z_1 - CC \times CN \tag{3.1}$$

The second cost component in the objective function is the cost of capital investment associated with various machines in the system. The capital investment cost is the difference between purchase price and salvage value plus the accumulated interest on the capital investment. In other words, it is the cost of depreciating the machine or

equipment during the period of interest. This cost functions  $Z_2$  is as follows:

$$Z_2 = \sum_{k=1}^K \sum_{j=1}^N MC_{kj} \times Y_{kj} \quad (3.2)$$

Purchasing a new machine would bring out some additional costs including planning, designing, ordering, transportation, set-up, and integration with the system costs. Also, there is a cost associated with the obsolete machines, which are no longer needed for the planning period. If the machine is not sold, there would be storage costs associated with it. The total procurement and salvage cost function  $Z_3$  is as following:

$$Z_3 = \sum_{k=1}^K [(PC_k \times PUR_k) + (SC_k \times SEL_k)] \quad (3.3)$$

Idle time which is unutilized machine processing time is the concept of fourth cost function in the model. Each machine has a turnover ratio for unit processing time. If a machine is idle, it does not bring that turnover, which reduces the profit. This time should be minimized to increase the efficiency of the system and investment. Moreover, the cost function would balance the utilization of machines. ( The cost of idle machining time for same type of machines is assumed to be same in the model without considering the parts that will be processed on them and their profit margins per unit machining time. ) The cost function  $Z_4$  is as following:

$$Z_4 = \sum_{k=1}^K IC_k \times ID_k \quad (3.4)$$

Inter-cell movements causes extra costs other than cost of intra-cell material handling. These costs could be extra and special handling equipments, part set-up cost regarding extra fixtures, additional work-in-process, monitoring costs. Quality control, scheduling, information flow problems might also arise which would require extra man power cost. Although inter-cell movement is inappropriate for cellular manufacturing philosophy, this is the more suitable way to deal with exceptional parts in most of cases, to increase utilization and to avoid from subcontracting. The cost of inter-cell movement per unit processing time out of original cell is the coefficient of the cost function. This value is different from each type of machine and it can be found approximately from this equation:

$$AC_k = \frac{CIM}{NPM_k \times APT_k} \quad (3.5)$$

The decision variable is going to be the processing time which will be utilized in another cell for type machine k. The cost function  $Z_5$  would be as following:

$$Z_5 = \sum_{k=1}^K \sum_{j=1}^N AC_k \times M_{kj} \quad (3.6)$$

In the model processing cost is not considered, and it is assumed that there is no alternative machine for each operation of a part. This makes the processing cost constant. Therefore, the sixth cost type of the model considers the additional cost of subcontracting parts rather than producing in the system. These values could be estimated by subtracting the raw material, operating and overhead cost of a part from subcontracting cost. Non-utilization of machines because of this subcontracting has already been considered in idle time cost function. So there is no need to add this term to resulting value. This cost type may also be used as opportunity cost of not producing the part in the system, if the part can not be subcontracted. The total subcontracting cost function  $Z_6$  would be as following, considering each part type separately, and as whole demand.

$$Z_6 = \sum_{i=1}^N SUC_i \times S_i \quad (3.7)$$

The objective function of this model will be to minimize the total cost which is summation of the six cost functions presented above. The model does not aim to minimize a specific cost, but the total cost arising from trade-offs between cost functions. The objective function can be written as following:

$$MIN Z \quad ; \quad Z = Z_1 + Z_2 + Z_3 + Z_4 + Z_5 + Z_6 \quad (3.8)$$

### 3.2.4 Developing constraint sets

Each part must be either assigned to a part family or subcontracted. A part can not be in more than one part families. There is no possibility not to produce or subcontract a part, as long as, the opportunity cost of not providing the part is compensated. As mentioned in the objective function development section  $S_i$  variable can be used for both subcontracting cost and opportunity cost of not producing a part. There is no need to use an extra variable and objective function. The constraint set is as following:

$$\sum_{j=1}^N X_{ij} + S_i = 1 \quad \forall i=1,2,\dots,N \quad (3.9)$$

The number of part families might already be planned or the existing layout of the facility might allow a number of cells. In this case, the following constraint ensures the number of cells/part families. This number must be assigned while modelling a system.

$$\sum_{j=1}^N X_{ij} - CN = 0 \quad (3.10)$$

If the existing layout is flexible and available for any number of cells in a range, then any number would be assigned to CN, and additional two constraints would be used to put the limitation of range.

$$MINC \leq CN \leq MAXC \quad (3.11)$$

A part can belong to a part family if it is formed, otherwise, it can not be assigned. To bring an upper and/or a lower limitation for number of parts to be assigned to a part family would also be useful to balance the utilization of cells. This set of constraints would also decrease the time to get the solution. The following set of constraints are more useful for the balancing cell sizes, when the utilization of cells is low:

$$MINP \times X_{jj} \leq \sum_{i=1}^N X_{ij} \leq MAXP \times X_{jj} \quad \forall j=1,2,\dots,N \quad (3.12)$$

If the utilization of the cells is high enough, these two constraints are not important anymore. Moreover, they may increase the solution time and cause to get a sub-optimal solution. In these cases, the following constraint set would be used for the first purpose of the set, which is to assign parts to the part families.

$$\sum_{i=1}^N X_{ij} - N \times X_{jj} \leq 0 \quad \forall j=1,2,\dots,N \quad (3.13)$$

For each machine type in the system there is available machining time limitation. As the required number of machines from each type are not known, an exact value can not be used. In the following set of constraints, there are basically four parts to state an equilibrium. Total processing time required on a machine type, available machining time depends on number of machines from same type, possible idle time for this machine(s), and required overtime for this machine(s), which means some of the processes should be accomplished out of cell, if it is possible. This relation could be presented as following:

$$\sum_{i=1}^N (D_i \times PT_{ik}) \times X_{ij} - TMC_{kj} \times Y_{kj} + AMC_{kj} - M_{kj} = 0 \quad \forall j=1,2,\dots,N, \quad \forall k=1,\dots,K \quad (3.14)$$

For the same type of machines, summation of each machine's "overtime" and total idle time of a same type of machines must give the summation of each machine's idle time. Idle time and "overtime" can not exist together. If idle time of some machines are equal to overtime of other machines, then there is no idle time for that type of machines. The following constraint set states this integrality:

$$\sum_{j=1}^N (AMC_{kj} - M_{kj}) + ID_k = 0 \quad \forall k=1,2,\dots,K \quad (3.15)$$

To find out required number of machines from each type to purchase and to sell, the following constraint set will be used. If there is a need to purchase a type of

machine, it means there is no possibility to sell from the same type of machine. Since variables can not get negative value, one of the terms may get a positive value. This constraint set provides the value of variables for responding cost function.

$$\sum_{j=1}^N Y_{ij} - EM_k - PUR_k + SEL_k = 0 \quad \forall k=1,2,\dots,K \quad (3.16)$$

Intra-cell handling and tooling cost must be considered, as well as other costs included in objective function. Since this and following models has been developed for specific purposes, these two costs are not considered in the calculations, even if they affect the results of the model. To keep these two type of costs at a level and to ensure the average similarity level to be higher than a level, the following constraint would be used. This constraint type would bring a lower bound to model, and make the model find out optimum in less time.

$$\sum_{i=1}^N \sum_{j=1}^N S_{ij} \times X_{ij} \geq SL^l \times \left( N - \sum_{i=1}^N S_i \right) \quad \forall l=1,\dots,L \quad (3.17)$$

If machine similarity is less important than other similarity types, the above constraint type must be used to optimize the system; otherwise the model could automatically keep the machine similarity high enough to minimize the cost types based on machines.

The following sets of constraints would be used to ensure the integrality of the model.

$$X_{ij} \in (0,1) \quad \forall i=1,2,\dots,N \quad \forall j=1,2,\dots,N \quad (3.18)$$

$$\text{integer } Y_{kj}, \quad \forall k=1,2,\dots,K \quad \forall j=1,2,\dots,N \quad (3.19)$$

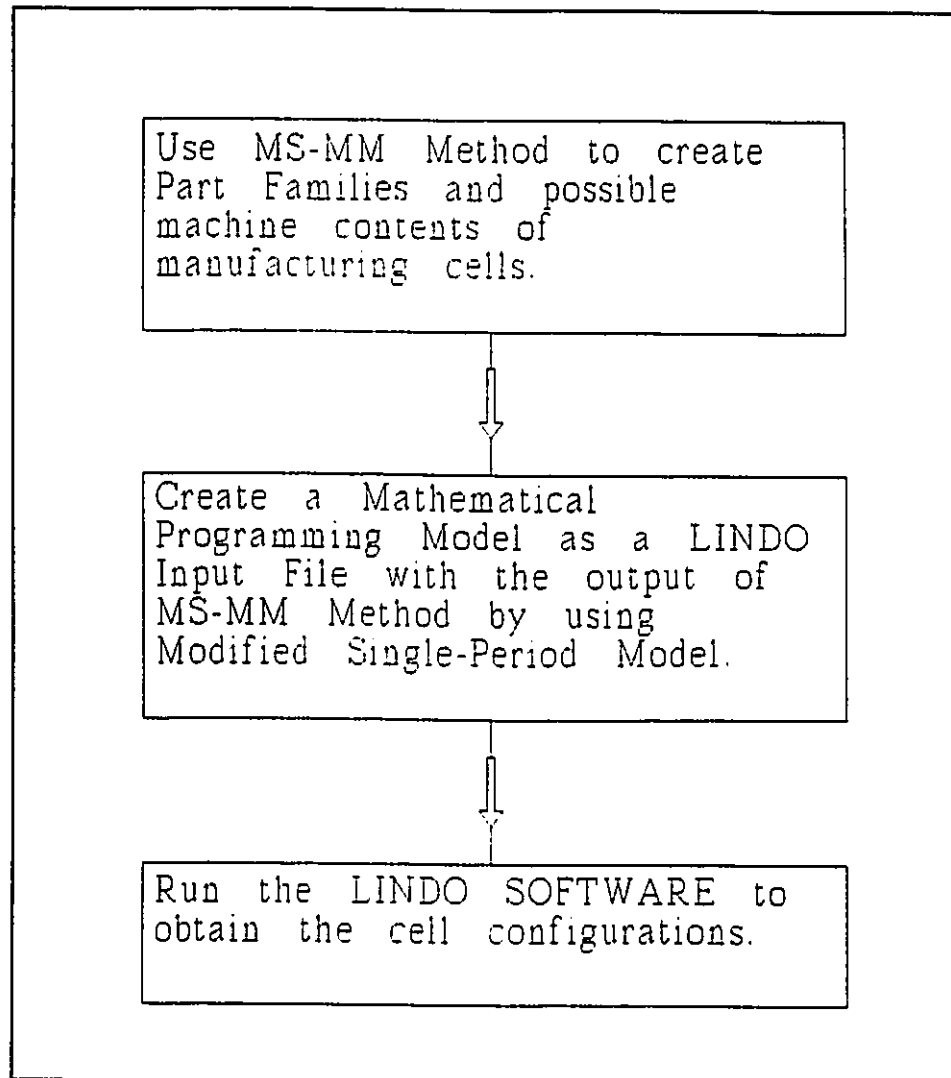
The above presented model would have  $N \times (N+K+1)$ , integer variables; where  $N$  is the number of part types and  $K$  is the number of machine types. However, stating  $N \times (N+K)$  variables would be enough, since first constraint set (eq. 3.9) would make  $S_i$  values to get integer values. For small size cases, this model could be efficient. However, for mid and large size problems, number of integer variables would increase the computational time dramatically.

### 3.3 Two-Stage Single-Period PF/MCC Model

As indicated in previous section, large-scale problems require significant amount of computational times to obtain optimal solutions. This is due to the fact that there are several integer variables, which increase exponentially with the increase in the number of parts and machines. Therefore, it is warranted to develop an heuristic technique to reduce the amount of computation. In this section a two-stage model is developed. In the first stage an heuristic model is used to form part families and to generate possible machines that may be included in the cell. The second stage of the

generate possible machines that may be included in the cell. The second stage of the model optimizes the total cost associated with machine cell configuration. It provides the answers of questions such as, if a part type will be produced or subcontracted, if a machine type will exist in the cell or not, and how many of them will exist in a cell.

The basic flowchart of the two-stage algorithm is in Figure 3.1.



**Figure 3.1** Two-Stage Single-Period PF/MCC Model.

### **3.3.1 Maximum Similarity-Minimum Machines (MS-MM) Method**

In the single-period model,  $X_{ij}$  and  $S_i$  are 0/1 integer variables, and  $Y_{kj}$  and CN are the integer variables. Since the number of  $S_i$  and CN variables cannot be reduced, there is no need to define them as integers because of the model structure. Therefore, the only integer variables left are  $X_{ij}$  and  $Y_{kj}$ . Reducing the number of integer variables would also reduce the total number of constraints in the model as well as the number of computational steps.

The proposed algorithm is based on the criteria of Maximum Similarity-Minimum Machines (MS-MM). The MS-MM algorithm maybe considered as Max-Min, which reduces the number of variables and minimizes the range between maximum and minimum number of variables. The heuristic model is divided into a main algorithms and two sub-routines. In the first sub-routine (Procedure A), part families are created, by using a similarity level, and without considering minimum part number in a cell. In the second sub-routine (Procedure B), the members of part families whose sizes are less than a pre-stated level, are placed to the families, which have highest similarities with them. The computer code of the method is in Appendix I. The flowchart of the main algorithm is in Figure 3.2. The major steps of the main algorithm are the following:

- **MAIN ALGORITHM**

- S1. Set the desired number of cells (CN), and search ranges for minimum part family size (PL), and lower similarity level SL  $[SL_1, SL_2, \dots, SL_n]$  in the system; MAX=1 and MIN=LARGE NUMBER.
- S2. Take the first element of SL range set.
- S3. CALL the PART A of Heuristic (Creating Part Families including part families (PF-LT-PLs), whose sizes are less than PL and part families (PF-GE-PLs), whose sizes are greater than and equal to PL).
- S4. Calculate the number of PF-GE-PLs (NPL), whose part sizes are greater and equal to PL.
- S5. If ( NPL = CN ) goto step 7, otherwise step 6.
- S6. Take the next element of SL range set, if there is, and goto step 3; otherwise goto step 14.
- S7. CALL the PART B of Heuristic (Modifying Part Families).
- S8. Calculate the similarity level (MSL) in the system.
- S9. If  $MSL > MAX$ , then goto step 10, otherwise goto step 11.
- S10. Update MAX (MAX = MSL), Update set of solution for maximum similarity.
- S11. Calculate the required number of machines (RNM).
- S12. If  $RNM < MIN$ , then goto step 13, otherwise goto step 15.
- S13. Update MIN (MAX = MSL), Update set of solution for minimum machine number, and goto step 6.

- S14. Calculate similarities per machine for Maximum Similarity-MS and Minimum Machines-MM Criteria.
- S15. TERMINATE.

The flowchart of Procedure A is in Figure 3.3, and major steps of the heuristic algorithm in Procedure A are the following:

- **PROCEDURE A**
- S1. Calculate similarity coefficients between each part,  $S_{ij}$ 's.
- S2. Create a PART Set =  $\{P_i; i=1,2,\dots,N\}$ .
- S3. Open an empty set,  $FAMILY_j^{SL}$ , and set  $i=1$ ; and add the first part  $P_i$ , in the set PART.
- S4. Search for parts, which have similarity with  $P_i$  more than and equal to  $SL$ , and place them in set  $FAMILY_j^{SL}$ ; and erase from set PART.
- S5. Take the next element of set  $FAMILY_j^{SL}$  and  $P_i$ , where  $(i=i+1)$ ;
- S6. If  $P_i$  is not  $\phi$ , goto Step 4; otherwise goto step 8.
- S7. if set PART does not have an element (  $PART = \{\phi\}$  ) goto step 8; otherwise increase the value of  $j$  by 1,  $(j=j+1)$  and update  $J$  ( number of family  $J=j$ );and goto step 3.
- S8. TERMINATE.

The flowchart of Procedure B is in Figure 3.4, and major steps of the heuristic algorithm in Procedure A are the following:

- **PROCEDURE B**

- S1. Calculate family sizes  $PN_i$  ( $i=1,2,\dots,J$ )
- S2. Determine the machine requirements of each part family (PF-GE-PL), whose size is greater than and equal to PL (Minimum Part Family Size).
- S3. Take one of the part families (PF-LT-PLs), whose sizes are less than PL.
- S4. Take an element (REP-EL) of the PF-LT-PL.
- S5. Calculate the machine similarities between REP-EL and PF-GE-PLs.
- S6. Choose the PF-GE-PL, which has the highest machine similarity with the element REP-EL. Remove the element REP-EL from its PF-LT-PL, and place it to the PF-GE-PL, with the highest value. Update the machine requirement of that PF-GE-PL.
- S7. If there is any element left in the PF-LT-PL, then goto step 4; otherwise goto step 8.
- S8. If there is any part family PF-LT-PL, whose size is less than PL left, then goto step 3; otherwise goto step 9.
- S8. TERMINATE

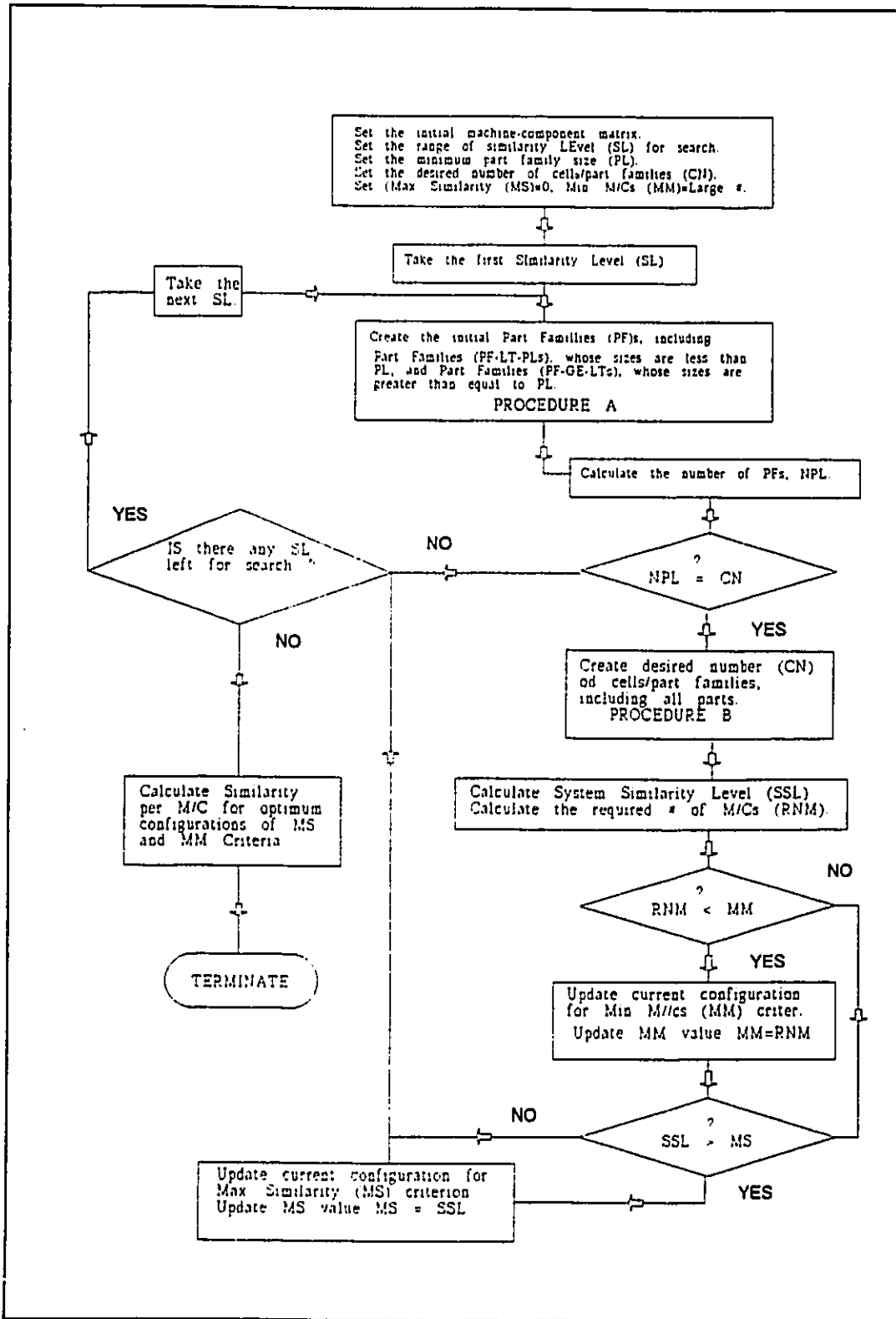


Figure 3.2 The main body of MS-MM Method.

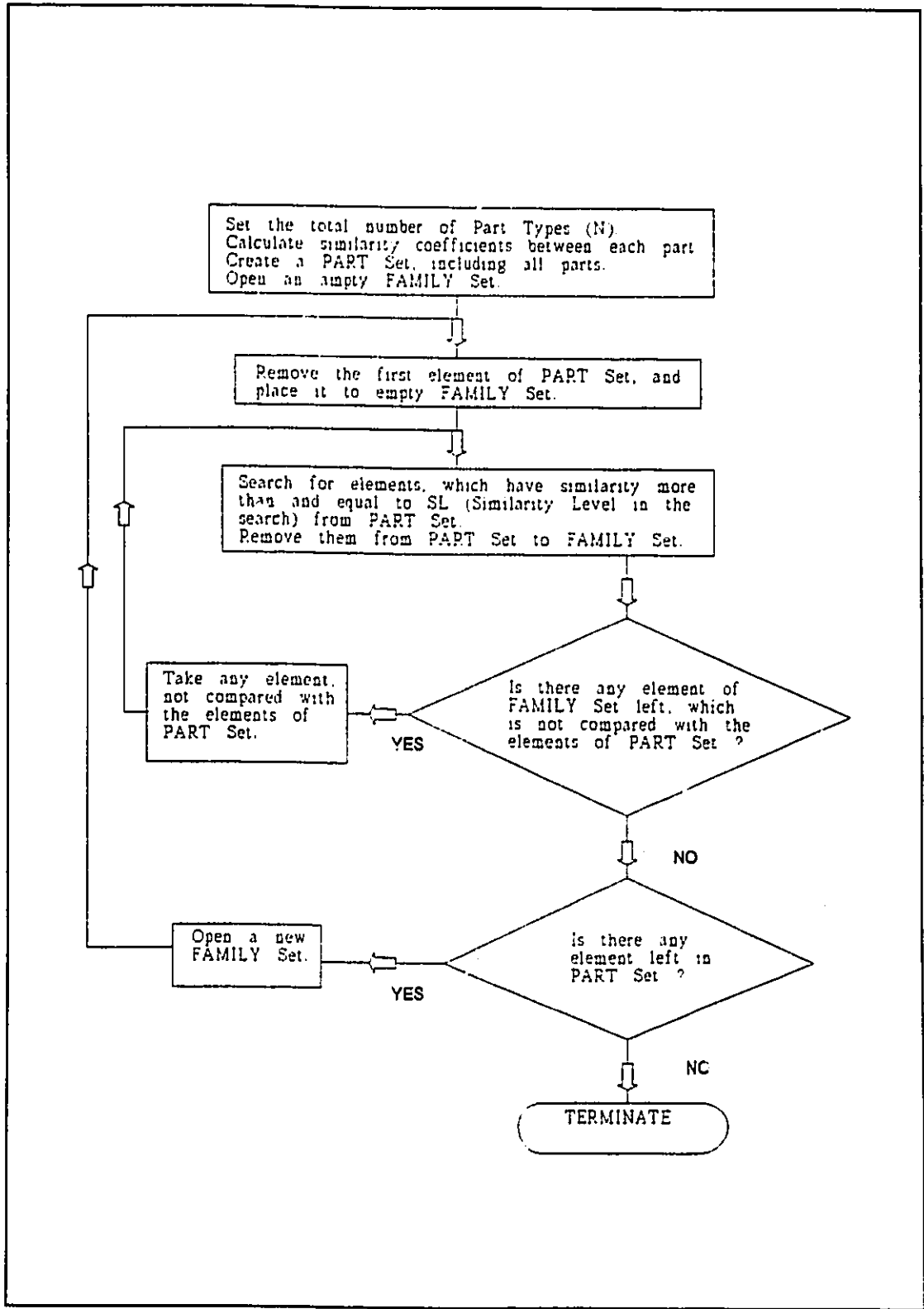


Figure 3.3 The Procedure A of MS-MM Method.

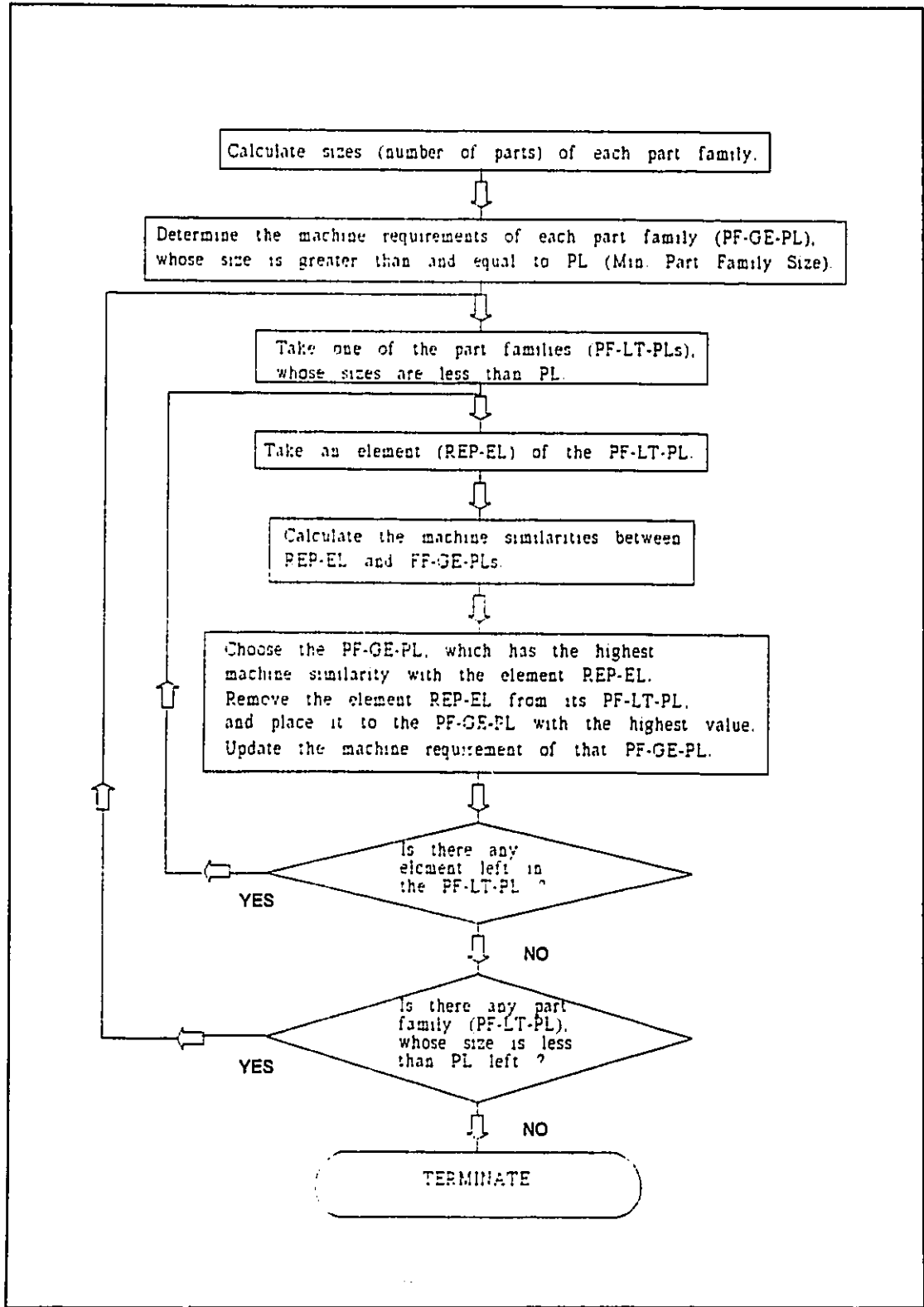


Figure 3.4 The Procedure B of MS-MM Method.

### 3.3.2 Development of weighted system similarity function

In this section the weighted system similarity function that has been used as the maximum similarity criterion in the MS-MM heuristic will be developed. The similarity level for a specific type such as, part similarity, in a created cell would be calculated by using following equation:

$$CSIM_c = \frac{\sum_{i=1}^{N_c} \sum_{j=i}^{N_c} S_{ij}}{\frac{(N_c - 1) \times N_c}{2}} \quad (3.20)$$

where;

$CSIM_c^l$       Similarity level of type l in cell c.

$N_c$           Number of parts in family c.

While calculating overall similarity in the system, weight concept would be used, since contribution of each cell is different based on part family sizes. The overall similarity level in the system would be:

$$OSIM^l = \sum_{c=1}^{CN} w_c \times CSIM_c^l \quad (3.21)$$

where;

**OSIM<sup>l</sup>** Overall l type similarity level in the system.

and,

$$w_c = \frac{N_c}{N}$$

The weighted similarity level of the system could be as following:

$$GSIM = \sum_{l \in SS} W_l \times OSIM^l \quad (3.23)$$

where;

**GSIM** Weighted similarity level in the system.

**SS** Set of similarity types { 1 (machine), 2 (tool), 3 (sequence),...}

**W<sub>l</sub>** Weight of similarity type of l.

and,

$$\sum_{l \in SS} W_l = 1 \quad (3.24)$$

The computer code of the algorithms are in Appendix I. The objective function in the mathematical model of the algorithm would also change, while constraint sets have some modifications.

### 3.3.3 Modified Single-Period Model

- **OBJECTIVE FUNCTION**

The objective function of the modified Single-Period Model will remain same, except Cell Configuration Cost Function, since number of cells are determined in the first stage, which will be as follows:

$$MIN Z \quad ; \quad Z = Z_1 + Z_2 + Z_3 + Z_4 + Z_5 + Z_6 \quad (3.25)$$

i. Cell Configuration Cost Function.

$$Z_1 = CC \times NC \quad (3.26)$$

ii. Machine Capital Investment Cost Function.

$$Z_2 = \sum_{j=1}^{CN} \sum_{k \in F_j} MC_k \times Y_{kj} \quad (3.27)$$

iii. Machine Procurement and Salvage Cost Function.

$$Z_3 = \sum_{k=1}^K [(PC_k \times PUR_k) + (SC_k \times SEL_k)] \quad (3.28)$$

iv. Idle Time Cost Function

$$Z_4 = \sum_{k=1}^K IC_k \times ID_k \quad (3.29)$$

v. Inter-cell Movement Cost Function.

$$Z_5 = \sum_{j=1}^{CN} \sum_{k \in F_j} AC_k \times M_{kj} \quad (3.30)$$

vi. Part Subcontracting Cost Function.

$$Z_6 = \sum_{i=1}^N SUC_i \times S_i \quad (3.31)$$

● **CONSTRAINT SETS**

Some of the constraint sets that were used in the Single-Period Model will not exist or the size of the some constraint sets will change in this model. There is no need to use part family forming constraints, since they have been already formed in the first stage.

The number of required constraints to create a model will also decrease, since

possible machine types that will exist in the cells have been already determined. The constraint sets will be as follows:

i. Each part must be produced in pre-determined cell or subcontracted.

$$X_{ij} + S_i = 1 \quad \forall i=1,2,\dots,N \quad j=1,2,\dots,CN \quad (3.32)$$

ii. Capacity constraint set determines the number of machines from each type in the cells. Possible machine types that may exist in the cells are determined in the first stage.

$$\sum_{i \in Z_k} (D_i \times PT_{ik}) \times X_{ij} - TMC_{kj} \times Y_{kj} + AMC_{kj} - M_{kj} = 0 \quad \forall j=1,2,\dots,CN, \forall k \in F_j \quad (3.33)$$

iii. Idle time constraint set keeps the integrity of second constraint set.

$$\sum_{j \in Z_k} (AMC_{kj} - M_{kj}) - ID_k = 0 \quad \forall k=1,2,\dots,K \quad (3.34)$$

iv. The constraint set provides the number of machines for procurement and sale.

$$\sum_{j \in Z_k} Y_{kj} - PUR_k + SEL_k - EM_k = 0 \quad \forall k=1,2,\dots,K \quad (3.35)$$

v. The following sets of constraints ensure the integrality of the model.

$$X_{ij} \in (0,1) \quad \forall i=1,2,\dots,N \quad j=1 \vee 2 \vee \dots \vee CN \quad (3.39)$$

$$\text{integer } Y_{kj}, \quad \forall j=1,2,\dots,CN \quad \forall k \in F_j \quad (3.40)$$

The computer code of first stage of Two-Stage Single-Period Model and LINDO input file generator are in Appendix II.

### 3.4 Model Developments for Multi-Period Strategies

In the literature, there are mainly two approaches to deal with product demands, while forming machine cells. One of them and the most common one is to consider expected part demands of only one period. This type of models considers just one period, and does not realize the part demand fluctuations and possible new parts to be produced in the following periods. The other approach is to consider the predicted maximum demand of products. This type of models mostly does not consider new parts, and is not sensitive to fluctuations. The only benefit of these models is flexibility of the system. However, they do not consider the arising costs of the system. Since the maximum demand values would be under consideration for each period, machine numbers would not change at each period. If high flexibility of the system is considered because of the possible high capacity, need for a relocation

would be a small possibility. This approach causes considerable low utilization for cells. When idle time and capital investment costs are relatively lower than the other costs, this approach might be reasonable. In real life cases, since utilization of the system and capital investment are two key goals, this approach would not be tested in this study. The proposed multi-period planning approach realizes part demand fluctuations, and stables system configuration for the periods under consideration, while considering new parts in the next periods.

Three main multi-period strategies will be determined, and they will be tested. The first one is to create a system considering the first period, and push the system without modifications for periods. The second one is to create a system considering the first period, and change the system configuration according to the part-mix of the planning period. The third strategy, which is the purpose of this study is to create a system, while considering more than one period and to keep the configuration stable during those periods.

A number of predictable periods regarding average part and cell lives which could be limited within three to five years, would be under consideration, in order to investigate the arising costs, and utilization level of the system, and required capital investment in three strategies. Three models will be developed to examine the strategies.

### **3.5 Multi-Period PF/MCC Model**

This model is developed to form machine groups and part families simultaneously in a cellular manufacturing system, while considering main cost groups regarding design implementation stages of the system. The model considers more than one period, while forming part families and machine cells. The machine cells will be formed by considering future part-mix. Part families may change throughout the periods under consideration. The following general assumptions and additional notations are used to develop the model.

#### **3.5.1 General assumptions**

- i. Part-mix may change from period to another throughout the planning horizon.
- ii. Some parts can be subcontracted. However, those parts must be either completely subcontracted or processed in the system.
- iii. Selling and buying machines are allowed, if the machine cell is reconfigured.
- iv. Inter-cell movements are allowed in the system.
- v. Duration of each planning period, and corresponding part-mix and production levels are known.
- vi. Production information such as relevant costs, and available machining capacities is known in advance.

vii. Inventory carrying to the next periods is not allowed.

### 3.5.2 Additional notations

#### i. Indices

$t$  Period index.

#### ii. Parameters

$N_t$  Number of parts to be produced in period  $t$ .

$K_t$  Number of machine types required in period  $t$ .

$TMC_{kj,t}$  Total machining capacity for machine type  $k$  in cell  $j$  in period  $t$ .

$D_{it}$  Annual demand of part type  $i$  in period  $t$ .

$Q_t$  Set of parts under consideration in period  $t$ .

$F_{it}$  The set of machines that part  $i$  requires in period  $t$ .

$F_j$  The set of machines that part family  $j$  requires.

$F_{kt}$  The set of parts that require machine type  $k$  in period  $t$ .

$Z_{1t}$  Cell configuration cost at the beginning of period  $t$ .

$Z_{2t}$  Annual capital investment cost for machines in period  $t$ .

$Z_{3t}$  Fixed selling and purchasing cost in period  $t$ .

$Z_{4t}$	Inter-cell movement cost in period t.
$Z_{5t}$	Idle time cost / work unbalance cost in period t.
$Z_{6t}$	Subcontracting cost in period t.

### iii. Decision Variables

#### a. 0/1 Integer Variables

$$X_{ijt} = \begin{cases} 1 & \text{if part } i \text{ belongs to part family } j \text{ in period } t, \\ 0 & \text{otherwise.} \end{cases}$$

$$S_{it} = \begin{cases} 1 & \text{if part } i \text{ is subcontracted in period } t, \\ 0 & \text{otherwise.} \end{cases}$$

#### b. General Integer Variables

$NUM_k$	Number of machine type k will exist in system.
$PUR_{k1}$	Number of new machine type k in period 1.
$SEL_{k1}$	Number of machine type k to sell in period 1.

#### c. General Variables

$AMC_{kjt}$	Available machining capacity for machine type k in cell j, excluding the processing capacity needed for part family j in period t.
$ID_{kt}$	Total idle time for machine type k in the system in period t.
$M_{kjt}$	Process time allocated out of cell j for machine k in period t.

### 3.5.3 Developing objective function

The objective of multi-period planning model is to minimize various cost components associated with the configuration design of manufacturing cells in multi-period planning horizon. The cost components of the model are similar to the cost components of Single Period PF/MCC Model. All of the cost components will exist for the number of periods times, except cell configuration and machine procurement and salvage cost, since configuration of the system is allowed once at the beginning of planning horizon. The objective function of the model would be as following:

$$MIN Z ; Z = Z_{11} + Z_{31} + \sum_{t=1}^T [ Z_{2t} + Z_{4t} + Z_{5t} + Z_{6t} ] \quad (3.41)$$

The components of the objective function are as following:

i. Cell Configuration Cost Function.

$$Z_{11} = CC \times CN \quad (3.42)$$

ii. Machine Capital Investment Cost Function.

$$Z_{2t} = \sum_{j=1}^N \sum_{k=1}^K MC_k \times Y_{kj} \quad (3.43)$$

iii. Machine Procurement and Salvage Cost Function.

$$Z_{3t} = \sum_{k=1}^K [(PC_k \times PUR_{kt}) + (SC_k \times SEL_{kt})] \quad (3.44)$$

iv. Idle Time Cost Function for one period.

$$Z_{4t} = \sum_{k=1}^K AC_k \times M_{kt} \quad (3.45)$$

v. Inter-cell Movement Cost Function for one period.

$$Z_{5t} = \sum_{j=1}^N \sum_{k=1}^K IC_k \times ID_{jt} \quad (3.46)$$

vi. Part Subcontracting Cost Function for one period.

$$Z_{6t} = \sum_{i=1}^{N_t} SUC_i \times S_{it} \quad (3.47)$$

### 3.5.4 Developing constraint sets

i. The constraint sets of multi-period model have similar content as the constraint sets

of Single-Period PF/MCC Model. The only difference is multi-period consideration. Therefore, most of the constraint sets include time parameter, and number of constraints in the sets is increasing by the effect of time parameter. The constraint sets of the Multi-Period PF/MCC Model are as following:

i. Each part must belong to a family to be produced or it must be subcontracted.

This constraint set is written for t periods.

$$\sum_{j=1}^{N_t} X_{ijt} + S_{it} - 1 \quad \forall i \in Q_t \quad \forall t=1,2,\dots,T \quad (3.48)$$

ii. This constraint ensures the number of machine cells or part families for each period.

$$\sum_{j \in Q_t} X_{ijt} - CN = 0 \quad \forall t=1,2,\dots,T \quad (3.49)$$

iii. A part can not belong to a part family, which is not created. This constraint set is necessary for t periods.

$$\sum_{i=1}^N X_{ijt} \leq N_t \times X_{jrt} \quad \forall j=1,2,\dots,N \quad \forall t=1,2,\dots,T \quad (3.50)$$

iv. Capacity constraint set determines the number of machines from each type in the cells. The production time of each period and processing times of parts on each machine are the input of this constraint set. This constraint set is also written for  $t$  periods.

$$\sum_{i=1}^{N_t} (D_{it} \times PT_{ik}) \times X_{ijt} - TMC_{kjt} \times Y_{jt} + AMC_{kjt} - M_{kjt} = 0 \quad \forall j \in Q, \forall k=1, \dots, K, \forall t=1, \dots, T \quad (3.51)$$

v. Idle time constraint set keeps the integrity of the fourth constraint set for  $t$  periods.

$$\sum_{j=1}^{N_t} (AMC_{kjt} - M_{kjt}) + ID_{kt} = 0 \quad \forall k=1, 2, \dots, K \quad \forall t=1, 2, \dots, T \quad (3.52)$$

vi. This constraint set provides the number of machines to procure and to sell at the beginning of planning horizon, and this constraint set is only for the first period.

$$\sum_{j=1}^N Y_{jt} - EM_k - PUR_{kt} + SEL_{kt} = 0 \quad \forall k=1, 2, \dots, K \quad (3.53)$$

vii. This constraint set bounds the part similarity of the system at a level. This constraint set would not be used, if only machine similarity is under consideration.

$$\sum_{i \in Q_t} \sum_{j \in Q_t} S_{ijt} \times X_{ijt} \geq SL \times \left( N_t - \sum_{i=1}^{N_t} S_{it} \right) \quad \forall t=1,2,..T \quad (3.54)$$

viii. The following sets of constraints are to ensure the integrality of the model.

$$X_{ijt} \in (0,1) \quad \forall i \in Q_t, \forall j \in Q_t, \forall t=1,2,..,T \quad (3.55)$$

$$\text{integer } Y_{kj} \quad \forall j=1,2,..,N, \quad \forall k=1,2,..,K \quad (3.56)$$

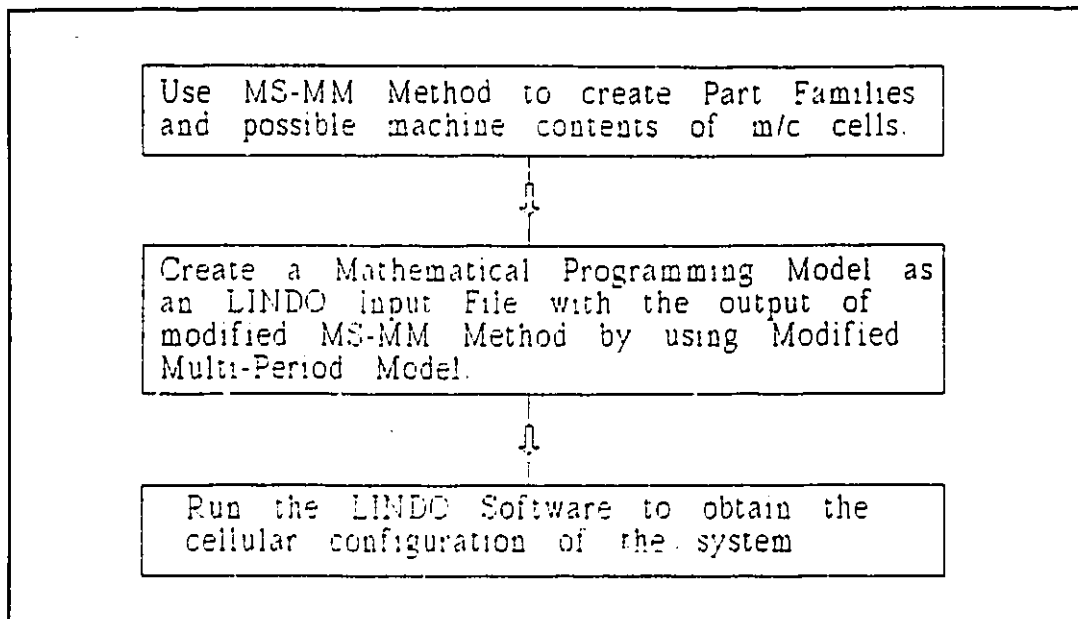
### 3.6 Two-Stage Multi-Period PF/MCC Model

It is obvious that, with the increase in the number of parts and machines, the above presented model would have a greater number of variables and constraints. This leads to longer computational times to obtain the optimal solution. To eliminate those ones, the model below proposed would be effective, especially for large size problems. To reduce the complexity of large scale problems a two-stage model will be proposed. In the first stage of model part families will be formed and possible machine contents of cells will be determined by using modified MS-MM Method. In the second stage the machine cells will be created by using modified Multi-Period PF/MCC Model, while determining parts to be produced, and required number of machines to procure and sell and amount of inter-cell movements.

### 3.6.1 Modified MS-MM Algorithm

- S1. Apply MS-MM Method to determine part families and possible machine contents of cells.
- S2. Determine the parts to be produced in cells for each period.
- S3. Determine possible machine types to exist in cells.
- S4. Apply Modified Multi-Period PF/MCC Model to determine machine requirements of cells and parts to be produced or subcontracted.

Computer code of the algorithm is in Appendix III, and the basic flowchart of the algorithm is in Figure 3.5.



**Figure 3.5 Two-Stage Multi-Period PF/MCC Algorithm**

### 3.6.2 Modified Multi-Period PF/MCC Model

Modified Multi-Period PF/MCC Model has the same characteristics as Modified Single-Period PF/MCC Model. The constraints associated with part family formation would not take place. The objective function and constraint sets of the model are as following:

- **OBJECTIVE FUNCTION**

Min Z

$$Z = Z_{11} + Z_{31} + \sum_{t=1}^T [ Z_{2t} + Z_{4t} + Z_{5t} + Z_{6t} ] \quad (3.57)$$

$$Z_{11} = CC \times NC \quad (3.58)$$

$$Z_{2t} = \sum_{j=1}^{CN} \sum_{k \in F_j} MC_{kj} \times Y_{kj} \quad (3.59)$$

$$Z_{31} = \sum_{k=1}^K [(PC_k \times PUR_{kl}) + (SC_k \times SEL_{kl})] \quad (3.60)$$

$$Z_{4t} = \sum_{j=1}^{CN} \sum_{k \in F_j} AC_k \times M_{kjt} \quad (3.61)$$

$$Z_{5t} = \sum_{j=1}^{CN} \sum_{k \in F_j} IC_k \times ID_{kt} \quad (3.62)$$

$$Z_{6t} = \sum_{i=1}^{N_t} SUC_i \times S_{it} \quad (3.63)$$

• **CONSTRAINT SETS**

$$X_{ijt} + S_{it} = 1 \quad \forall i \in Q_t, \forall j=1 \vee 2 \vee \dots \vee CN, \forall t=1,2,\dots,T \quad (3.64)$$

$$\sum_{i \in Z_{kt}} (D_{it} \times PT_{ikt}) \times X_{ijt} - AMT_{kt} \times Y_{kj} + I_{kjt} - M_{kjt} = 0 \quad \forall j=1,2,\dots,CN, \forall k \in F_j, t=1,2,\dots,T \quad (3.65)$$

$$\sum_{j \in Z_{kt}} (AMC_{kjt} - M_{kjt}) + ID_{kt} = 0 \quad \forall k=1,2,\dots,K \quad \forall t=1,2,\dots,T \quad (3.66)$$

$$\sum_{j \in Z_k} Y_{kj} - EM_k - PUR_{kl} + SEL_{kl} = 0 \quad \forall k=1,2,\dots,K \quad (3.67)$$

$$X_{ijt} \in (0,1) \quad \forall i \in Q_t, \forall j=1 \vee 2 \vee \dots \vee CN, \forall t=1,2,\dots,T \quad (3.68)$$

$$\text{integer } Y_{kj} \quad \forall j=1,2,\dots,CN, \forall k \in F_j \quad (3.69)$$

The maximum and minimum numbers of 0/1 and general integer variables, and constraints that may exist in the four models that have been presented are in Table 3.1

An illustrating numerical example is also presented in Table 3.2.

**Table 3.1 Model Sizes**

DESCRIPTION		# of 0/1 Integer Variables		# of General Integer Variables		Number of Constraints	
		Min	Max	Min	Max	Min	Max
Single Period	Without MS-MM	$N^2$	$N^2$	$NK$	$NK$	$2N+K*(N+2)+1$	$2N+K*(N+2)+1$
	With MS-MM	$N$	$N$	$K$	$CN*K$	$N+3K$	$N+K*(CN+2)$
Multi Period	Without MS-MM	$P*(N/P)^2$	$PN^2$	$NK$	$NK$	$2PN+PK*(N+1)+P$	$2PN+PK*(N+1)+P$
	With MS-MM	$N$	$PN$	$K$	$CN*K$	$PN+2PK+K$	$PN+PK*(CN+1)+K$

where;  $N$  Number of Part Types in the system  
 $K$  Number of Machine Types in the system  
 $CN$  Number of Cells in the system  
 $P$  Number of Periods under consideration

**Table 3.2 Numerical Example for Model Sizes**

DESCRIPTION		# of 0/1 Integer Variables		# of General Integer Variables		# of Constraints	
		Min	Max	Min	Max	Min	Max
Single Period	With MS-MM	729	729	324	324	403	403
	Without MS-MM	27	27	12	36	63	87
Multi Period	Without MS-MM	243	2187	324	324	1173	1173
	With MS-MM	27	81	12	36	165	237

where; 27 Number of Part Types in the system  
12 Number of Machine Types in the system  
3 Number of Cells in the system  
3 Number of Periods under consideration

## **3.7 Multi-Period Planning Strategies**

In this section, three multi-period planning strategies are presented. Strategy I and Strategy II are the most common strategies with single-period consideration. Strategy III is the proposed strategy, to overcome the shortcomings of the other two strategies. The schematic presentations of Strategies are in Figure 3.6.

### **3.7.1 Strategy I**

One of the main reason of low utilization of machines or not enough capacity is short term planning. High cost of reconfiguration prevents frequent reallocation of cells in real life systems, which is considered in this case. Therefore, configuration will be done at the beginning of first period. There will be no more reconfiguration allowed, until the end of multi-period under consideration, since one of the strategies in the companies is to push the existing system for some periods, until reallocation of the system is unavoidable.

In this case, Two-Stage Single PERIOD PF/MCC Model will be applied for the first period. For the rest of the periods the same model will be applied with a small modification. There will be no cell configuration and machine procurement and salvage cost components in the objective function, since reconfiguration is not

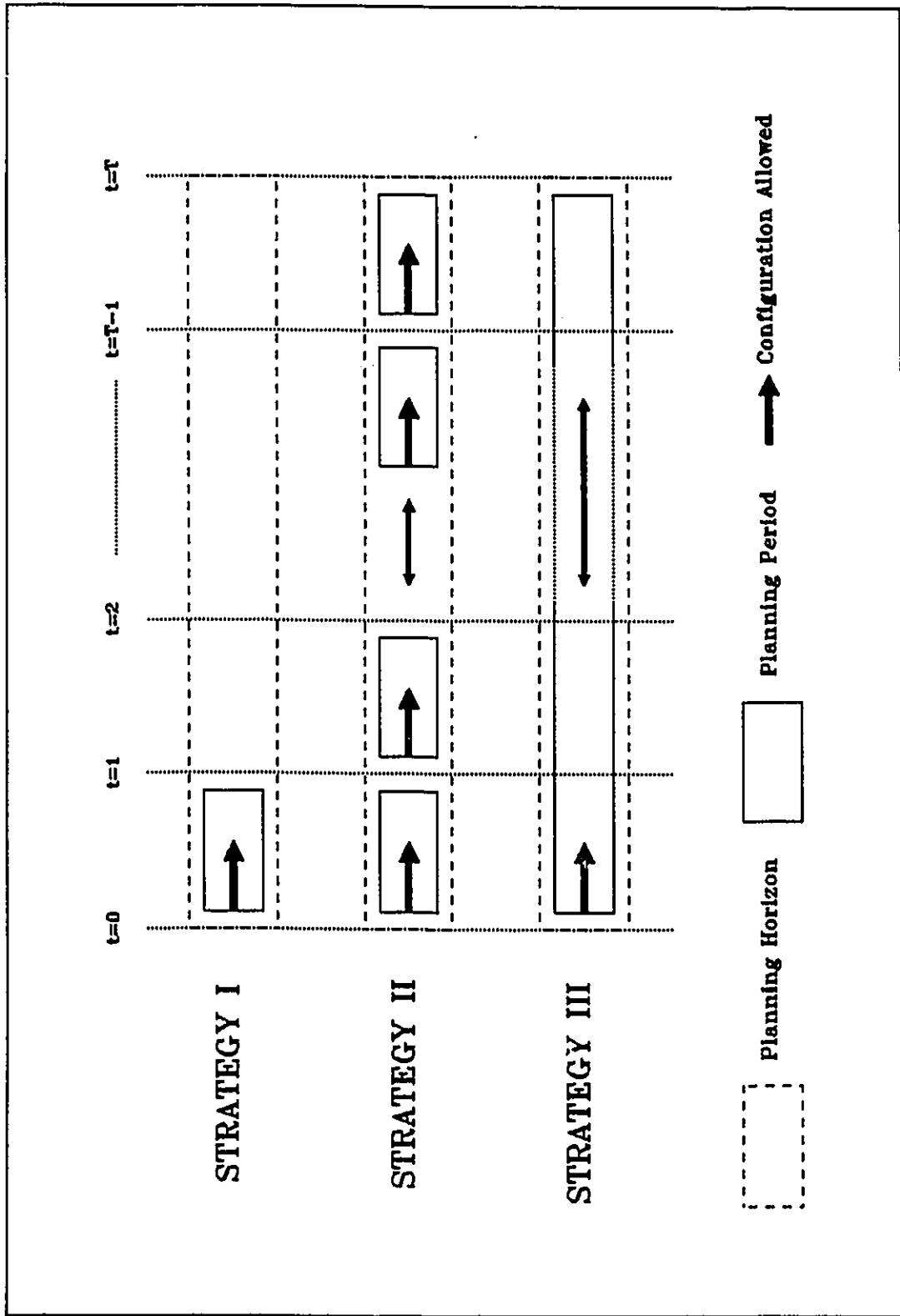


Figure 3.6 Planning Processes of Strategies

allowed. The values of  $Y_{kj}$  would be assigned in advance, and the constraint associated with machine procurement and salvage cost would not take place in the constraint sets. The summation of the total costs regarding the periods under consideration, would give the optimum value for this strategy.

### **3.7.2 Strategy II**

In this strategy, it is possible to reallocate machines, to purchase new machines or to sell machines, at the beginning of each period. The following questions should be answered, if one period is considered at each step:

- Existing system is enough to produce all demands ?
- Does the system need new machines ?
- Is it worth to purchase new machines ?
- What about subcontracting some of the parts in this period ?
- Does system have machines more than enough ?
- Is it worth to sell some of the machines ?
- Is reconfiguration of the system feasible ?

To provide answers of the above questions, the Two-Stage Single PERIOD PF/MCC Model will be applied for the first period , and the Two-Stage Single PERIOD PF/MCC Model would be applied without cell configuration cost function for the rest

of periods. In this case, if reallocation is required for periods, an additional set configuration cost would be added depending on new configurations, after optimum solutions are obtained for each period. Total cost of the strategy would be the summation of total costs of each period and total cell configuration cost.

### **3.7.3 Strategy III**

At the previous two strategies, each period was analyzed individually without considering all periods under consideration. In this strategy, planning will be done by considering the part demands, machine requirements and corresponding costs for each period of planning horizon. In this case, part families and cells will be created at the beginning of first period, and they will not change for the rest of the periods. The cost of cell configuration procurement and sale will take place only in the first period, since cell reconfiguration, and machine procurement and sale are allowed just in the first period. The proposed Two-Stage Multi-Period PF/MCC Model would be applied to seek the trade-offs between costs arising at each period.

# **CHAPTER 4**

## **COMPUTATIONAL RESULTS**

### **4.1. General**

The proposed MS-MM Method is tested by using the example of Burbidge (1975). Results of the MS-MM Method are compared with the results of Burbidge (1975), and King (1980). Proposed two-stage models are compared with the results of the mathematical models. Larger size problems are solved using the proposed two stage models. Furthermore, the proposed strategies discussed in the previous chapter are examined through the use of some examples. These strategies can be used as a tool of decision making to determine the best machine cell configuration. The behaviours of the model with different inter-cell movement cost values are also discussed at the end of this chapter.

### **4.2 Capability of MS-MM Method**

Burbidge (1975), presented a group technology problem with 14 machines and 43

components, which was solved by manual trial-and-error method. Table 4.1 shows the same initial machine-component matrix, which also used in a study by King (1980). The example data is also used in this study to test the proposed heuristic, and compared its results with those of Burbidge (1975) and King (1980).

Burbidge solution resulted in five cells and it has three exceptional parts that cannot be completely assigned to a cell. In order to produce all of the parts within machine cells the system must have 25 machines. In addition, the weighted similarity in the system, according to the weighted system similarity function developed in previous chapter is 0.615.

The proposed heuristic is able to find an optimum solution according to Minimum Machine-Maximum Similarity (MM-MS) criteria for a given cell size. When the example data is used with the proposed heuristic employing five machine cells, it gives similar results as Burbidge's Manual Solution of 25 machines and 43 part families. In addition, similar results of the weighted system similarities are obtained using both methods. Table 4.2 shows the solution results of both the proposed heuristic and Burbidge method. It is clear that both solution are identical using the example data.

The proposed algorithm is also used with the example data considering four machine cells. Results obtained in this case revealed a total of 22 machines are required to process all parts within the system, and 43 part families are formed. Furthermore, the

weighted system similarity is 0.539. When the ROC algorithm (King, 1980) is applied to the same example data using four machine cells, results revealed 2 exceptional parts and a total of 23 machines required to process all parts. The weighed system similarity is 0.523, which is less than the weighted similarity of MS-MM result.

Comparing both results of the proposed MM-MS and ROC algorithms it can be stated that the proposed MM-MS gives better solution than ROC. It is clear that the total required number of machines to process all parts of the MS-MM solution are less than the ROC solution. MS-MM Method gives a reasonable high weighted similarity, which is 0.601, with 23 machines under Maximum Similarity criterion. The comparison results of ROC and MS-MM solutions are presented in Table 4.3.

The above results show that MS-MM Method can be used efficiently in the other applications for the formation of perfect machine cells and part families in cellular manufacturing systems. The MS-MM Method would work for any determined number of cells, which is feasible to form in a system.

**Table 4.1 Burbidge's Initial Matrix**

Part Number	Machine Number															
	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1
	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
01	0	0	0	0	0	1	1	1	0	1	0	0	0	0	0	0
02	0	1	0	0	0	1	0	1	1	0	0	0	0	1	0	1
03	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0
04	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
05	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	1
06	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0
07	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	1
08	0	0	0	0	1	1	0	1	0	0	0	0	0	0	0	0
09	0	0	0	1	1	0	0	1	0	0	1	0	0	0	0	0
10	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1
11	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0
12	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0
13	0	0	0	0	1	1	0	0	1	0	0	0	0	0	0	0
14	0	0	0	1	1	1	0	0	0	0	0	0	0	0	1	0
15	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0
16	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
17	0	0	1	0	0	1	0	0	0	0	0	0	0	1	0	0
18	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1
19	0	0	0	1	1	1	0	1	0	0	0	0	0	0	1	0
20	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0
21	0	0	0	1	1	0	0	1	0	0	0	0	0	0	1	0
22	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
23	0	0	0	1	1	1	0	1	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	1	0	0	1	1	1	0	0	0
25	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
27	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0
28	0	1	0	0	0	0	0	1	1	0	0	0	0	0	0	0
29	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
31	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0
32	0	1	0	0	0	1	0	0	1	0	0	0	0	0	0	1
33	0	0	0	0	1	1	0	0	0	0	0	0	0	0	1	0
34	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0
35	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0
36	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
37	1	1	0	0	0	1	0	1	1	0	0	0	0	0	0	1
38	0	1	0	0	0	0	0	1	1	0	0	0	0	0	0	1
39	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0
40	0	1	0	0	0	1	0	0	1	0	0	0	0	0	0	0
41	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0
42	1	1	0	0	0	1	0	0	1	0	0	0	0	0	0	1
43	0	0	0	0	1	1	0	1	0	0	0	0	0	0	1	0

**Table 4.2 Comparisons of Burbidge's Manual Method and MS-MM Method**

Number of Cells	MS-MM SOLUTION		BURBIDGE SOLUTION	
	Part Families	M/C Groups	Part Families	M/C Groups
1	1,12,13,25,31,39,26 (7)	6,7,8,10 (4)	1,12,13,39,31,25,26 (7)	6,7,8,10 (4)
2	2,10,28,32,37,38,40 42,18,4 (10)	1,2,6,8,9 14,16 (8)	2,37,42,32,38 10,40,28,18,4 (10)	1,2,6,8 9,14,16 (7)
3	3,20,24,27,11,30,22 (7)	8,11,12,13 (4)	24,3,27,20,11,30,22 (7)	8,11,12,13 (4)
4	5,14,19,21,29,33,41 8,15,43,23,9,16 (13)	4,5,6,8,11 15 (6)	19,23,14,43,5,9,21 41,15,29,8,33,16 (13)	3,6,14,16 (4)
5	6,17,7,34,35,36 (6)	3,6,14,6 (4)	17,7,35,6,34,36 (6)	4,5,6,8,11,15 (6)
<b>TOTAL</b>	<b>43</b>	<b>25</b>	<b>43</b>	<b>25</b>
<b>Weighted Sim.</b>	<b>.615</b>		<b>.615</b>	

**Table 4.3 Comparison of ROC and MS-MM Solutions**

Number of Cells	MS-MM SOLUTION		ROC Solution	
	Part Families	M/C Groups	Part Families	M/C Groups
1	1,12,13,25,31,39,26 (7)	6,7,8,10 (4)	1,12,13,39,31,25,26 (7)	6,7,8,10 (4)
2	2,10,28,,32,37,38,40 42,18,36,35,347,17,6 (15)	1,2,3,6,8,9 14,16 (8)	2,37,42,32,28,38,10 40,18,4,76,34,17,35,36 (16)	1,2,3,6,89,4,16 (8)
3	3,20,24,27,11,30,22 0(7)	8,11,12,13 (4)	24,3,27,9,20,11,30,22 (8)	4,5,8,11,12,13 (6)
4	5,14,19,21,29,33,4,18 15,43,23,9,16 (14)	4,5,6,8,11 15 (6)	19,14,21,5,23,29,33 41,43,8,15,16 (12)	4,5,6,8,15 (5)
<b>TOTAL</b>	<b>43</b>	<b>22</b>	<b>43</b>	<b>23</b>
<b>Weighted Sim.</b>	<b>.539</b>		<b>.523</b>	

### **4.3 Testing and Evaluation of Two-Stage models**

The single and multi-period two-stage PF/MCC models are tested using a small size hypothetical example data in the single and multi-periods mathematical models developed in the previous chapter. The results of the single-period two-stage PF/MCC model are compared with those that are obtained from LINDO output of the Single-Period PF/MCC model. Similarly, the example data is modified to facilitate the requirements of the Multi-Period PF/MCC model. Again, the results of Two-Stage Single-Period PF/MCC model are compared with the those that are obtained from LINDO output of the Multi-Period PF/MCC model.

#### **4.3.1 Two-Stage Single-Period PF/MCC model testing**

To test the validity of the Two-Stage Single-Period PF/MCC model, a hypothetical data of seven parts and six machines is used. Part demands and processing requirements of the hypothetical example are given in Table 4.4. In addition, various cost parameters and planning horizon are given in Table 4.5. The data is used as an input to both models; the Two-Stage Single-Period PF/MCC and Single-Period PF/MCC models. In both cases, two machine cell formation is considered. The list of LINDO input files that have been used for the comparisons are in Appendix IV for single-period cases, and in Appendix V for multi-period cases.

**Table 4.4 Part Demands and Processing Requirements**

Part No	Machine Type						Part Demands
	01	02	03	04	05	06	
01	1.25	1.50	0.75	0	0	0	200
02	0	0	0	0.80	0.90	1.20	175
03	0.60	1.00	0	1.30	0	0	100
04	0	0	1.20	0	1.25	0.70	250
05	0	1.10	1.20	1.00	0	0	280
06	0	0	0	1.75	0	1.25	150
07	1.00	0	1.30	0.60	0	0	120

**Table 4.5 Cost Parameters and Planning Horizon**

Description	Unit Values
Planning Horizon	700 Unit Time
Unit Cell Configuration Cost	100 Unit Cost
Unit Machine Procurement Cost	10 Unit Cost
Unit Capital Investment Cost	15 Unit Cost
Unit Idle Time Cost	40 Unit Cost
Unit Inter-cell Movement Cost	10 Unit Cost
Unit Part Subcontracting Cost	80 Unit Cost

Results of the Single-Period PF/MCC model for two machine cells indicated that the first machine cell consists of machines 1,2,3 and 4, while the second machine cell consists of machines 3,5 and 6. On the other hand, their corresponding part families consist of parts 1,3,,5 and 7, and parts 2,4 and 6. These results are identical to those that are obtained from the Two-Stage Single-Period PF/MCC model. The results of both cases are summarized in Table 4.6. Also, various cost results of the optimal solution are given in Table 4.7.

Comparison of the model constraints and decision variables of both models and computational times associated with each one are given in Table 4.8. Furthermore, the Two-Stage Single-Period PF/MCC model is tested with other hypothetical problems. The results of all cases were near identical. It is clear that the proposed model gives results which are either optimal or near optimal and can be used to handle large size problems as will be shown in later sections.

**Table 4.6 Results of MS-MM Method**

Cell Number	Members of Part Family		Members of Machine Group	
	Part Number	Total	Machine Number	Total
1	1-3-5-7	4	1-2-3-4	4
2	2-4-6	3	3-5-6	3

**Table 4.7 Results of Single-Period Models**

Description	Cost Values
Cell Configuration Cost	200 Unit
Machine Purchasing Cost	70 Unit
Capital Investment Cost	105 Unit
Idle Time Cost	58 Unit
Inter-cell Movement Cost	41 Unit
Subcontracting Cost	0 Unit
<b>TOTAL COST</b>	<b>474 Unit</b>

**Table 4.8 Comparison of Single-Period Models**

Comparison Parameters	MODELS	
	Modified S-P Model	S-P Model
Number of 0/1 Integer Variable	7	49
Number of General Integer Variable	8	42
Number of Constraints	21	63
Number of iterations to reach optimum	319	54253
Required computer time to reach optimum	2 seconds	41 minutes

### 4.3.2 Two-Stage Multi-Period PF/MCC model testing

To test the validity of the Two-Stage Multi-Period PF/MCC model, a hypothetical data of eight parts, six machines and 3 period is used. Part demands and processing requirements of the hypothetical example are given in Table 4.9. In addition, various cost parameters and planning horizon are given in Table 4.10. The data is used as an input to both models; the Two-Stage Single-Period PF/MCC and Single-Period PF/MCC models. In both cases, two machine cell formation is considered.

**Table 4.9 Unit Processing Times**

Part No	Machine Type						Part Demands		
	01	02	03	04	05	06	Per 1	Per 2	Per 3
01	1.25	1.50	0.75	0	0	0	150	0	0
02	0	0	0	0.80	0.90	1.20	120	0	0
03	0.60	1.00	0	1.30	0	0	200	160	0
04	0	0	1.20	0	1.25	0.70	180	130	0
05	0	1.10	1.20	1.00	0	0	0	300	250
06	0	0	0	1.75	0	1.25	0	200	150
07	1.00	0	1.30	0.60	0	0	0	0	300
08	0	0	1.00	0	1.25	1.10	0	0	250

**Table 4.10 Cost Parameters and Planning Horizon**

Description	Unit Values
Planning Horizon	1500 Unit Time
Duration of a period	500 Unit Time
Unit Cell Configuration Cost	100 Unit Cost
Unit Machine Procurement Cost	10 Unit Cost
Unit Capital Investment Cost	15 Unit Cost
Unit Idle Time Cost	20 Unit Cost
Unit Inter-cell Movement Cost	10 Unit Cost
Unit Part Subcontracting Cost	80 Unit Cost

Results of the Multi-Period PF/MCC model for two machine cells indicated that the first machine cell consists of machines 1,2,3 and 4, while the second machine cell consists of machines 3,4,5 and 6. On the other hand, their corresponding part families consist of parts 1,3,5 and 7, and parts 2,4,6 and 8. These results are identical to those that are obtained from the Two-Stage Multi-Period PF/MCC model. The results of both cases are summarized in Table 4.11. Also, various cost results of the optimal solution are given in Table 4.12.

Comparison of the model constraints and decision variables of both models and computational times associated with each one are given in Table 4.13. Furthermore, the Two-Stage Multi-Period PF/MCC model is tested with other hypothetical problems. The results of all cases were near identical. It is clear that the proposed model gives reliable results which are either optimal or near optimal and can be used to handle large size problems as it will be shown in later sections.

**Table 4.11 Results of Multi-Period Models**

Cell Number	Members of Part Family		Members of Machine Group	
	Part Number	Total	Machine Number	Total
1	1-3-5-7	4	1-2-3-4	4
2	2-4-6-8	3	3-4-5-6	4

**Table 4.12 Results of Multi-Period Models**

Description	Cost Values
Cell Configuration Cost	200 Unit
Machine Purchasing Cost	80 Unit
Capital Investment Cost	120 Unit
Idle Time Cost	227 Unit
Inter-cell Movement Cost	20 Unit
Subcontracting Cost	0 Unit
<b>TOTAL COST</b>	<b>647 Unit</b>

**Table 4.13 Comparison of Multi-Period Models**

Comparison Parameters	MODELS	
	Modified S-P Model	S-P Model
Number of 0/1 Integer Variable	8	48
Number of General Integer Variable	12	48
Number of Constraints	55	118
Number of iterations to reach optimum	329	354527
Required computer time to reach optimum	2 seconds	230 minutes

#### 4.4 Strategy Evaluation

The presented strategies in the previous chapter are evaluated with an application of 3-period example in section 4.6.1, and an application for 5-period in section 4.6.2. A detailed analysis considering utilization, with-in cell utilization also exists for 5-period example. Two-Stage Single-Period PF/MCC Model and Two-Stage Multi-Period PF/MCC Model were used for applications, because of the time consideration.

Machine similarity between parts is under consideration for the applications, since proposed models are based on machine allocation in Cellular Manufacturing Systems. In addition, it is assumed that there is no existing machine in the system at beginning of first period for all applications in this chapter. The LINDO (Hyper version) software (run on IBM Compatible 486 SX Personal Computer) was used for the solutions of the linear programming models in applications. The source code of the computer programs are all in C language. The unit values of cost types that were used in the applications including in Section 4.5 are summarized in Table 4.14.

#### **4.4.1 Applications for 3-period**

In applications, 20 part types and 12 machine types are in consideration. Part demands for 3 periods are given at Table 4.15. Part demands are changing according to their product life cycles. Some of the parts are produced at three periods with changing quantities, such as, Part Number 8 and 9, not produced after first period such as Part Number 1 and 2, or after second period, such as, Part Number 5 and 6, introduced in second period, such as, Part Number 14 and 15, or on third period, such as, 19 and 20, and one of them which is Part Number 17 is produced only in period 2. Machine requirements of parts, and processing times of parts on these machines are given at Table 4.16.

**Table 4.14 Outline of Applications**

Parts of Studies			Sec. 4.3.1	Sec. 4.3.2	Sec. 4.3.3		
Applied Strategies			I-II-III	I-II-III	III		
Number of Periods			3	5	5		
Period Duration (Unit Time)			1200	1500	1500		
Cell Set-up Cost	per unit cell	CST	100	100	100	100	100
Machine Purchasing Cost	per unit machine	MPC	10	10	10	10	10
Machine Selling Cost	per unit machine	MSC	10	10	10	10	10
Capital Investment Cost	per unit machine	CIC	25	15	15	15	15
Idle Time Cost	per unit time	ITC	1/40	1/40	1/40	1/40	1/40
Inter-cell Movement Cost	per unit time	IMC	1/5	1/10	1/20	1/10	1/5
Subcontracting Cost	per unit part	SC	75	80	80	80	80

All the costs are in unit values, and the real dollar values can be found, by multiplying the arising costs with some constants. When the proposed Heuristic Method is applied to Machine-Component Matrix, the best solution is found with 3 part families/machine cells with both MS and MM Criteria. For MS Criterion similarity level is .766, and for MM Criteria, machine number is 23, with the same similarity found. The similarity per machine value is .033. The results of MS-MM Method are presented at Table 4.17, and the results of first stage of Two-Stage Multi-Period PF/MCC Model are presented in Table 4.18.

**Table 4.15 Part Demands for 3 Periods**

Part No	Part Demands (Unit)		
	Period 1	Period 2	Period 3
01	300	0	0
02	160	0	0
03	280	0	0
04	180	0	0
05	250	300	0
06	320	240	0
07	340	440	0
08	200	340	420
09	290	440	400
10	140	320	160
11	200	350	400
12	240	340	340
13	180	260	320
14	0	160	300
15	0	80	200
16	0	180	310
17	0	170	0
18	0	0	280
19	0	0	320
20	0	0	250

**Table 4.16 Machine Requirements and Processing Times of Parts Machines**

Part No	Machine Type											
	01	02	03	04	05	06	07	08	09	10	11	12
01	.4	0	0	.5	.6	0	0	.7	0	0	1.2	0
02	.5	0	.4	0	0	.9	0	0	0	0	0	.6
03	0	.75	.6	0	.5	0	0	0	.4	.7	0	0
04	.75	0	0	1.2	0	0	0	.8	.5	0	.4	0
05	0	.75	.6	0	0	0	.7	0	.9	.5	0	0
06	.4	.6	.75	0	0	0	.8	0	0	0	0	.75
07	0	0	0	.5	.4	0	0	.7	.8	0	.6	0
08	0	.6	0	0	.5	0	.4	0	1	.6	0	0
09	.9	0	.75	0	0	.6	.4	0	0	0	0	.5
10	0	.75	.7	0	.6	0	0	0	1	.4	0	0
11	.5	0	0	.8	0	0	0	1.2	.6	0	1.25	0
12	0	.6	.4	0	0	.5	.9	0	0	0	0	.7
13	0	.75	0	0	0	.8	.4	0	.8	.5	0	0
14	0	.6	0	.75	1	0	0	0	.4	.6	0	0
15	0	0	0	.7	.9	0	0	.5	0	0	1.2	0
16	0	.4	.75	0	0	0	1	0	0	.6	0	.75
17	0	.5	.8	0	0	.4	0	.7	0	0	0	.6
18	.6	0	0	.8	.5	0	0	.9	0	0	.4	0
19	0	0	.75	0	0	1.2	1.25	0	0	0	.5	.6
20	4	0	0	.7	.6	0	0	.6	0	0	.5	0

**Table 4.17 Results of MS-MM Algorithm for 3-Period Application**

Cell Number	Members of Part Family		Members of Machine Group	
	Part Number	Total	Machine Number	Total
1	1-4-7-11-15-18-20	7	1-4-5-8-9-11	6
2	2-6-9-12-16-17-19	7	1-2-3-6-7-8-10-11-12	9
3	3-5-8-10-13-14	6	2-3-4-5-6-7-9-10	8
TOTAL	20		23	

**Table 4.18 Results of first stage of 3-Period Application**

Cell Number	Members of Part Family		Members of Machine Group	
	Part Number	Total	Machine Number	Total
1	per1	1-4-7-11	1-4-5-8-9-11	6
	per2	7-11-15		
	per3	11-15-18-20		
2	per1	2-6-9-12	1-2-3-6-7-8-10-11-12	9
	per2	6-9-12-16-17		
	per3	9-12-16-19		
3	per1	3-5-8-10-13-14	2-3-4-5-6-7-9-10	8
	per2	5-8-10-13-14		
	per3	8-10-13-14		

The planning horizon is 3 periods, and each period is 1200 unit time. The strategies are evaluated with 75 unit of Subcontracting Cost and 1/5 unit of Inter-cell Movement Cost, 1/40 unit of Idle Time Cost, and 25 unit of Capital Investment Cost. The cellular configurations of each strategy are presented in Table 4.19, Table 4.20, and Table 4.21. The arising cost values from strategies are listed at Table 4.22, and graphical presentation of them are in Figure 4.1.

**Table 4.19 Results of STRATEGY I**

Cell Number		Members of Part Family		Members of Machine Group	
		Produced Parts	Subcontr. Parts	Machine #	Total
1	per1	-	1-4-7-11	-	0
	per2	-	7-11-15		
	per3	-	11-15-18-20		
2	per1	2-6-9-12	-	1-3-6-7-12	5
	per2	6-9-12-16	17		
	per3	9-12-16-19	-		
3	per1	3-5-8-10-13-14	-	2-3-5-9-10	5
	per2	5-8-10	13-14		
	per3	8-10-13	14		

**Table 4.20 Results of STRATEGY II**

Cell Number		Members of Part Family		Members of Machine Group	
		Produced Parts	Subcontr. Parts	Machine Number	Total
1	per1	-	1-4-7-11	-	0
	per2	7-11-15	-	4-8-9-11	4
	per3	11-15-18-20	-	1-4-5-8-11	5
2	per1	2-6-9-12	-	1-3-6-7-12	5
	per2	6-9-12-16-17	-	1-2-3-6-7-12	6
	per3	9-12-16-19	-	1-2-3-6-7-12	6
3	per1	3-5-8-10-13-14	-	2-3-5-9-10	5
	per2	5-8-10-13-14	-	2-3-5-7-9-10	6
	per3	8-10-13-14	-	2-5-7-9-10	5

**Table 4.21 Results of STRATEGY III**

Cell Number		Members of Part Family		Members of Machine Group	
		Produced Parts	Subcont. Parts	Machine Number	Total
1	per1	1-4-7-11	-	1-4-5-8-9-11	6
	per2	7-11-15	-		
	per3	11-15-18-20	-		
2	per1	2-6-9-12	-	1-2-3-6-7-12	6
	per2	6-9-12-16-17	-		
	per3	9-12-16-19	-		
3	per1	3-5-8-10-13-14	-	2-3-5-7-9-10	6
	per2	5-8-10-13-14	-		
	per3	8-10-13-14	-		

When Total Costs of Strategies are compared, Strategy III gives the best result.

Only two cells are established with Strategy I and II for the first period, while this number is three for Strategy III. Strategy II adds one more cell after first period because of dynamic structure of strategy, while Strategy I remains with the same number of cells. Strategy I has 10 machines in the first period, and it can not get more machines for the next periods. Strategy II has 10 machines for the first period, 16 machines for the second and third periods, which makes Strategy I totally infeasible with the current cost values. However, Strategy III has 17 machines for three periods, which is more than other strategies have. Strategy III provides the best solution, even with the highest number of cell sizes, since the planning horizon of the strategy includes three periods.

**Table 4.22 Strategy Comparisons for 3-Period Example**

Cost Type	Periods	Strategy I	Strategy II	Strategy III	Best Strategy
Cell Configuration Cost	Period 1	200	200	300	
	2	0	300	0	
	3	0	200	0	
	Total	200	700	300	I
Machine Procurement Cost	Period 1	100	100	170	
	2	0	60	0	
	3	0	20	0	
	Total	100	180	170	I
Machine Salvage Cost	Period 1	0	0	0	
	2	0	0	0	
	3	-	20	-	
	Total	0	20	0	I - III
Capital Investment Cost	Period 1	250	250	425	
	2	250	400	425	
	3	250	400	425	
	Total	750	1050	1275	I
Idle Time Cost	Period 1	139	139	259	
	2	134	179	209	
	3	143	164	194	
	Total	416	482	662	I
Inter-cell Movement Cost	Period 1	162	162	92	
	2	175	196	161	
	3	214	236	260	
	Total	551	594	513	III
Part Subcontracting Cost	Period 1	300	300	0	
	2	450	0	0	
	3	450	0	0	
	Total	1200	300	0	III
<b>TOTAL COST</b>	<b>Total</b>	<b>3217</b>	<b>3326</b>	<b>2920</b>	<b>III</b>

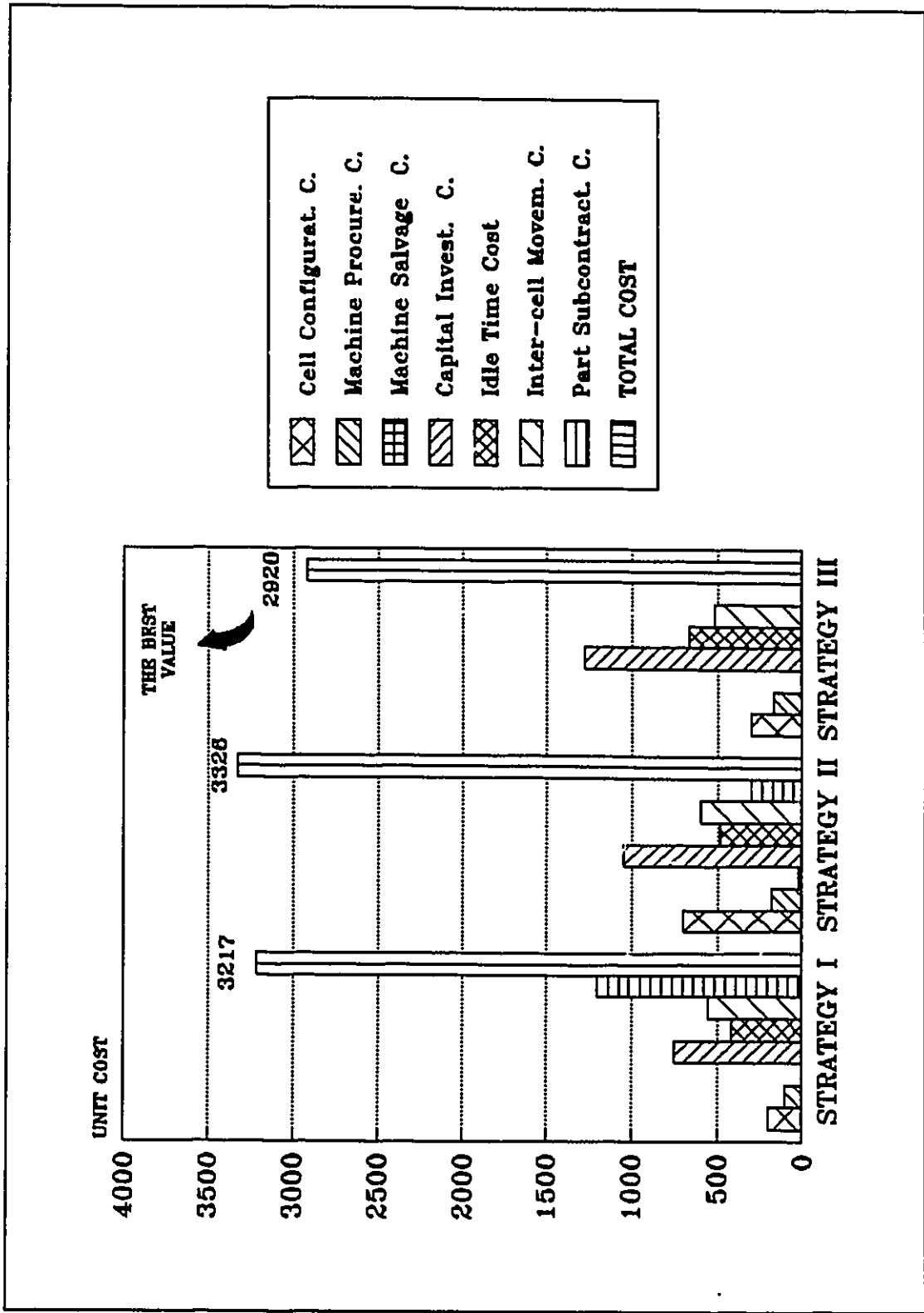


Figure 4.1 Evaluation of Strategies for 3-Period

High number of machines cause the highest Idle Time Cost, and lowest Inter-cell Movement Cost, and no Subcontracting Cost. Less number of machines causes so many subcontracted Parts, which is 16 parts in total for three periods, while this value is only four for Strategy II.

Strategy I, has the lowest value for Idle Time Cost. However, this value is still high, when machine numbers in each strategy are compared. That is one of the reason, why Strategy II gives better Total Cost than Strategy I does.

Strategy II has the highest Inter-cell Movement Cost, since the strategy optimizes each period separately. It also gives reasonable good value for Idle Time Cost. However, Total Cell Configuration Cost of Strategy II increases the Total Cost an important amount. Strategy II could be the best alternative, if unit cell configuration cost has a value from 0 up to 1.5 unit. This result shows that Strategy II can not be a good alternative for some cases, even where cell configuration cost is not reasonable high.

#### **4.4.2 Applications for 5-period**

In this section, the planning horizon is five periods. The strategies are evaluated with 80 unit of Subcontracting Cost and 1/10 unit of Inter-cell Movement Cost, 1/40 unit of Idle Time Cost, and 15 unit of Capital Investment Cost. There are 12 machine

types, and 20 parts, which have the same machine requirements, and processing times, as given in previous section. However, life cycles of parts are different. Product demands for 5-period are listed at Table 4.23, and some of the product life cycles in five periods are presented in Figure 4.2. In the third period, product-mix more variety than other periods. It is possible to know products to be introduced to the system in 3 periods advance, and no more variety is expected for the fourth and fifth periods. Average number of part types to be produced at each period is 13.6.

**Table 4.23 Part Demands for 5 Periods**

Part No	Period				
	1	2	3	4	5
01	400	0	0	0	0
02	260	0	0	0	0
03	380	200	0	0	0
04	280	120	0	0	0
05	350	200	190	0	0
06	320	240	160	0	0
07	440	440	400	0	0
08	600	540	460	340	200
09	480	440	400	280	150
10	0	320	360	400	0
11	300	400	400	360	280
12	600	640	640	560	440
13	460	560	720	760	720
14	0	160	300	400	440
15	0	80	200	360	600
16	0	160	310	480	800
17	0	170	250	420	750
18	0	0	180	360	620
19	0	0	150	320	580
20	0	0	250	500	950

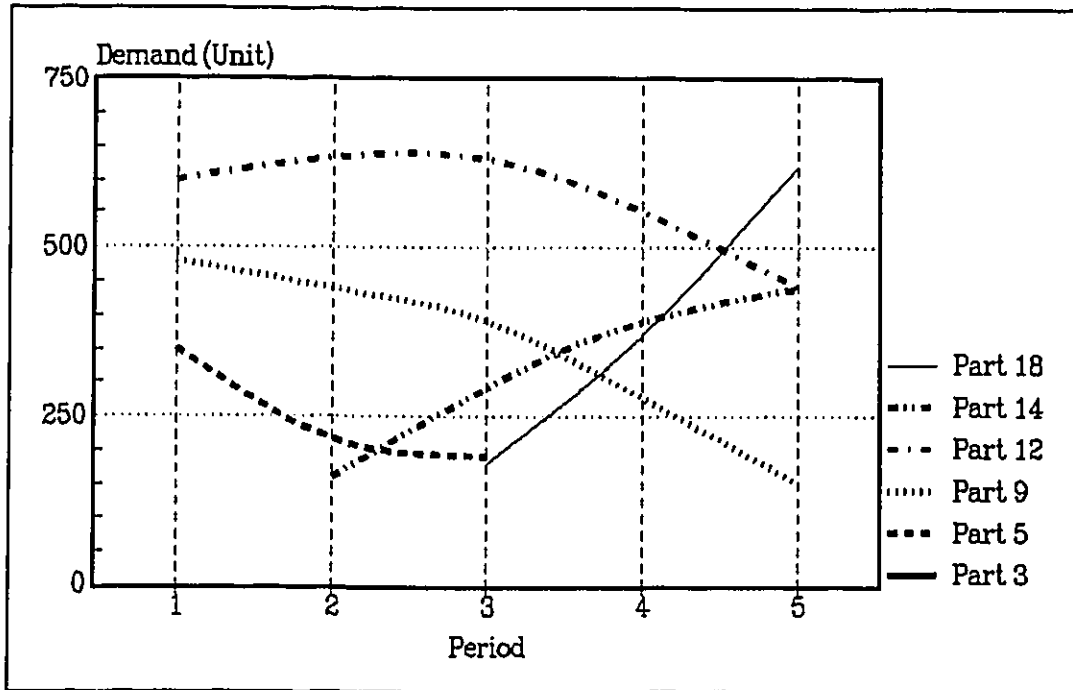


Figure 4.2 Example Product Demands in Five Periods

The results of MS-MM Method are same as 3-period application, since processing requirements of the parts are same. The results of first stage of Two-Stage Multi-Period PF/MCC Model are presented in Table 4.24. The cellular configurations of each strategy are presented in Table 4.25, Table 4.26, and Table 4.27. The results and comparison of applications are presented at Table 4.28, graphical presentation of arising costs for each strategy is in Figure 4.3. Results of strategies concerning cell utilization and within cell utilization are listed at Table 4.29, and graphical presentation of impacts of strategies on cell and system utilizations are presented in Figure 4.4.

**Table 4.24 Results of 1<sup>st</sup> stage of Two-Stage Multi-Period PF/MCC Model**

Cell Number		Members of Part Family		Members of Machine Group	
		Part Number	Total	Machine Number	Total
1	per1	1-4-7-11	4	1-4-5-8-9-11	6
	per2	4-7-11-15	4		
	per3	7-11-15-18-20	5		
	per4	11-15-18-20	4		
	per5	11-15-18-20	4		
2	per1	2-6-9-12	4	1-2-3-6-7-8-10-11-12	9
	per2	6-9-12-16-17	5		
	per3	6-9-12-16-17-19	6		
	per4	9-12-16-17-19	5		
	per5	9-12-16-17-19	5		
3	per1	3w-8-13	4	2-3-4-5-6-7-9-10	8
	per2	3-5-8-10-13-14	6		
	per3	5-8-10-13-14	5		
	per4	8-10-13-14	4		
	per5	8-13-14	3		

**Table 4.25 Results of STRATEGY I**

Cell Number		Members of Part Family		Members of Machine Group	
		Produced Parts	Subcont. Parts	Machine Number	Total
1	per1	1-4-7-11	-	4-8-9-11	4
	per2	4-7-11-15	-		
	per3	7-11-15-18-20	-		
	per4	11-15-18-20	20		
	per5	11-15	18-20		
2	per1	2-6-9-12	-	1-2-3-6-7-12	6
	per2	6-9-12-16-17	-		
	per3	6-9-12-16-17-19	-		
	per4	9-12-16-19	17		
	per5	9-12-16	19		
3	per1	3-5-8-13	-	2-5-7-9-10	5
	per2	3-5-8-13-14	10		
	per3	5-8-10-13-14	-		
	per4	8-10-13-14	-		
	per5	8-13-14	-		

**Table 4.26 Results of STRATEGY II**

Cell Number		Members of Part Family		Members of Machine Group	
		Produced Parts	Subcontr. Parts	Machine Number	Total
1	per1	1-4-7-11	-	4-8-9-11	4
	per2	4-7-11-15	-	4-8-9-11	4
	per3	7-11-15-18-20	-	4-8-9-11	4
	per4	11-15-18-20	-	1-4-5-8-11	5
	per5	11-18-20	15	1-4-5-8-9-11	6
2	per1	2-6-9-12	-	1-2-3-6-7-12	6
	per2	6-9-12-16-17	-	1-2-3-6-7-12	6
	per3	6-9-12-16-17-19	-	1-2-3-6-7-12	6
	per4	9-12-16-17	19	3-6-7-8-12	5
	per5	9-12-16-17	19	2-3-6-7-8-10-12	7
3	per1	3-5-8-13	-	2-5-7-9-10	5
	per2	3-5-8-10-13-14	-	2-3-5-7-9-10	6
	per3	5-8-10-13-14	-	2-5-7-9-10	5
	per4	8-10-14	13	2-4-5-9-10	5
	per5	8	13-14	-	0

**Table 4.27 Results of STRATEGY III**

Cell Number		Members of Part Family		Members of Machine Group	
		Produced Parts	Subcont. Parts	Machine Number	Total
1	per1	1-4-7-11	-		
	per2	4-7-11-15	-		
	per3	7-11-15-18-20	-		
	per4	11-15-20	18		
	per5	15-20	11-18	4-5-8-9-11	5
2	per1	2-6-9-12	-		
	per2	6-9-12-16-17	-		
	per3	6-9-12-16-17-19	-		
	per4	9-12-16-17-19	-		
	per5	9-12-17-19	16	1-2-3-6-7-12	6
3	per1	3-5-8-13	-		
	per2	3-5-8-10-13-14	-		
	per3	5-8-10-13-14	-		
	per4	8-10-13-14	-		
	per5	8-13-14	-	2-3-5-6-7-9-10	7

**Table 4.28 Strategy Comparisons for 5-Period Applications**

Cost Type	Periods	Strategy I	Strategy II	Strategy III	Best Strategy
Cell Configuration Cost	Period 1	300	300	300	
	2	0	100	0	
	3	0	100	0	
	4	0	300	0	
	5	0	200	0	
	<b>Total</b>		<b>300</b>	<b>1000</b>	<b>300</b>
Machine Procurement Cost	Period 1	150	150	180	
	2	0	10	0	
	3	0	0	0	
	4	0	30	0	
	5	0	0	0	
	<b>Total</b>		<b>150</b>	<b>190</b>	<b>180</b>
Machine Salvage Cost	Period 1	0	0	0	
	2	0	0	0	
	3	0	10	0	
	4	0	30	0	
	5	-	20	-	
	<b>Total</b>		<b>0</b>	<b>60</b>	<b>0</b>
Capital Investment Cost	Period 1	225	225	270	
	2	225	240	270	
	3	225	225	270	
	4	225	225	270	
	5	225	195	270	
	<b>Total</b>		<b>1125</b>	<b>1110</b>	<b>1350</b>
Idle Time Cost	Period 1	169	169	282	
	2	204	214	289	
	3	120	120	231	
	4	145	175	220	
	5	200	154	247	
	<b>Total</b>		<b>835</b>	<b>832</b>	<b>1269</b>
Inter-cell Movement Cost	Period 1	175	175	52	
	2	158	147	77	
	3	278	278	126	
	4	266	215	143	
	5	240	76	153	
	<b>Total</b>		<b>1117</b>	<b>891</b>	<b>551</b>
Part Subcontracting Cost	Period 1	0	0	0	
	2	80	0	0	
	3	0	0	0	
	4	160	160	80	
	5	240	320	240	
	<b>Total</b>		<b>480</b>	<b>480</b>	<b>320</b>
<b>TOTAL COST</b>	<b>Total</b>	<b>4007</b>	<b>4563</b>	<b>3970</b>	<b>III</b>

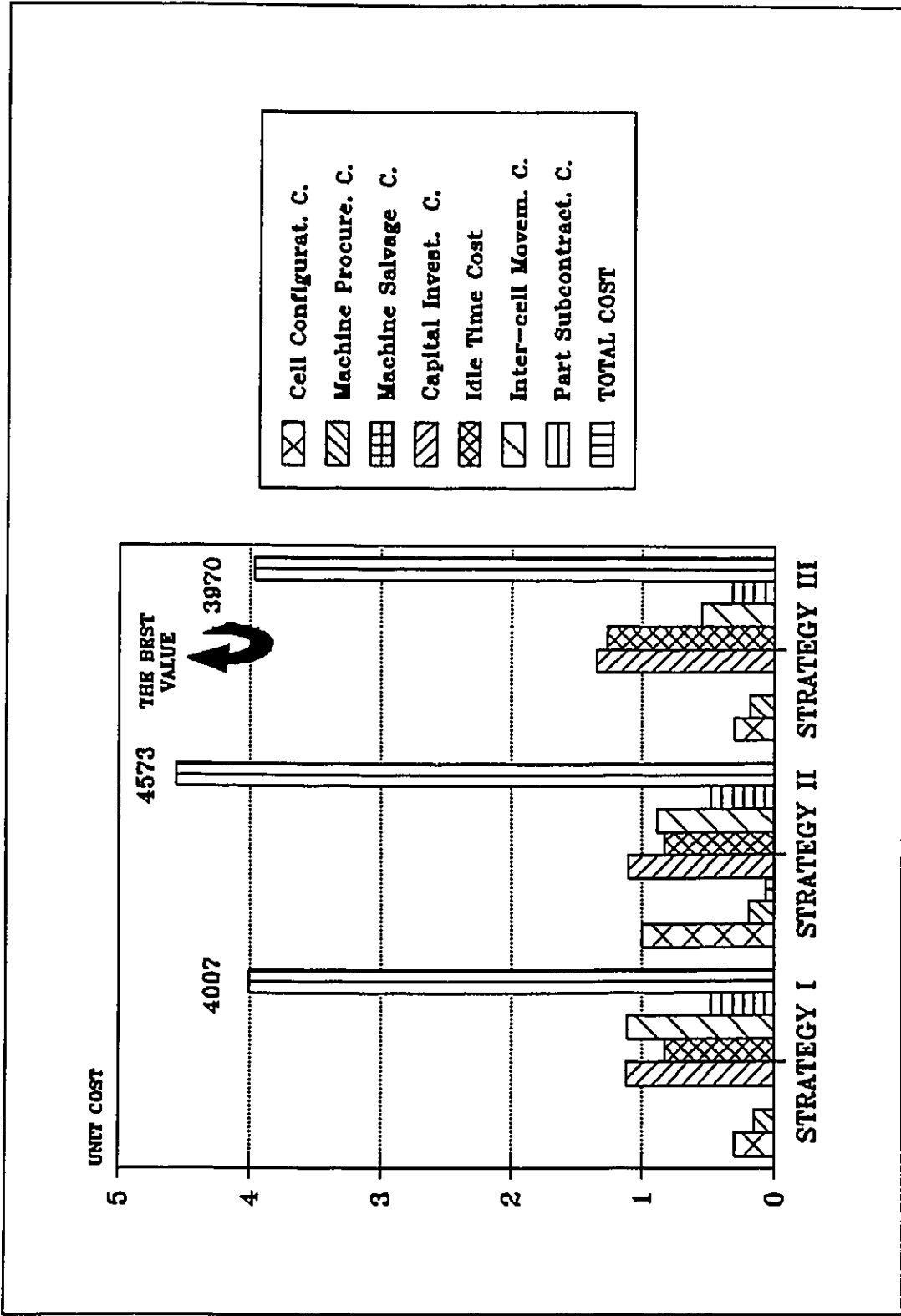


Figure 4.3 Evaluations of Strategies for 5-Period

When Total Costs of Strategies are compared, Strategy III gives the best result. Strategy III has more machine than other strategies. This causes high Idle Time Cost and low Inter-cell Movement Cost, and less Subcontracting Cost. Idle Time Cost for Strategy I and II are close to each other, since Capital Investment Cost and Subcontracting Costs are close to each other. However, Inter-cell Movement Cost of Strategy I is much more higher than Strategy II. Dynamic machine configuration of Strategy II causes that difference. However, this brings additional Machine Purchasing and Selling Cost.

Too many number of machines cause more Idle Time Cost, but less Inter-cell Movement Cost, and Subcontracting Cost. Strategy III has 3 Subcontracted Parts, while other strategies have 4 Subcontracted Parts. Subcontracted parts mostly vary for different strategies, with a few exceptions, as it can be observed at Table 4.29.

Strategy III has around half of the Inter-cell Movement Costs, which other strategies have. This provides 0.935 within cell utilization, which is only 0.854 for Strategy I, and 0.890 for Strategy II. However, Strategy III has the lowest cell utilization, which is 0.642. This value is 0.699 for Strategy I and 0.709 for Strategy II. Lower utilization would bring flexibility to the system, in the case of possible additional parts, that might be introduced into the system, in five periods.

**Table 4.29 Evaluation of Strategies for 5-Period Applications**

Type	Periods	Strategy I			Strategy II			Strategy III		
		Cell 1	Cell 2	Cell 3	Cell 1	Cell 2	Cell 3	Cell 1	Cell 2	Cell 3
Number of Machines	Per 1	4	6	5	4	6	5	5	6	7
	2	4	6	5	4	6	6	5	6	7
	3	4	6	5	4	6	5	5	6	7
	4	4	6	5	5	5	5	5	6	7
	5	4	6	5	6	7	0	5	6	7
	Cell Aver.	4	6	5	4.6	6	4.2	5	6	7
Sys. Aver.	5			4.93			6			
Number of Subcontracted Parts and Part Numbers	Per 1	0	0	0	0	0	0	0	0	0
	2	0	0	1	0	0	0	0	0	0
	3	0	0	0	0	0	0	0	0	0
	4	1	1	0	0	1	1	1	0	0
	5	2	1	0	1	1	2	2	1	0
	Cell Aver.	.6	.4	0.2	.2	.4	.6	.6	.2	0
Sys. Aver.	.4			.4			.267			
Processing Time within Cell (Unit Time)	Per 1	4071	5052	4877	4071	5052	4877	4487	5052	5683
	2	3224	5075	4502	3224	6075	5714	3472	5075	6162
	3	3888	5729	5314	3888	5729	5314	4468	5814	6256
	4	3786	5226	5048	5740	4274	3358	4674	6192	5936
	5	2518	5881	3528	5862	6751	-	4260	5765	5604
	Cell Aver.	3498	5393	4654	4557	5577	4816	4273	5580	5929
Sys. Aver.	4515			4984			5261			
Processing Time out of Cell (Unit Time)	Per 1	936	0	806	936	0	806	520	0	0
	2	538	227	808	538	227	700	290	227	252
	3	988	436	1354	908	436	1354	408	436	412
	4	1020	448	1196	456	1278	416	380	742	308
	5	680	1005	906	0	135	620	380	815	330
	Cell Aver.	833	424	1014	568	416	780	396	444	261
Sys. Aver.	757			588			367			
Within Cell Utilization	Per 1	.813	1.000	.858	.813	1.000	.858	.896	1.000	1.000
	2	.857	.957	.848	.857	.964	.891	.923	.957	.961
	3	.797	.929	.797	.811	.929	.797	.916	.930	.938
	4	.788	.921	.808	.926	.770	.890	.925	.893	.951
	5	.787	.854	.796	1.00	.980	-	.918	.876	.944
	Cell Aver.	.808	.932	.821	.881	.929	.859	.916	.931	.959
Sys. Aver.	.854			.890			.935			
Cell Utilization	Per 1	.678	.709	.706	.678	.709	.706	.598	.619	.541
	2	.577	.672	.648	.599	.757	.674	.590	.677	.614
	3	.758	.786	.811	.758	.786	.811	.738	.646	.613
	4	.705	.741	.787	.799	.625	.645	.725	.730	.565
	5	.562	.733	.606	.700	.673	-	.721	.730	.534
	Cell Aver.	.656	.728	.712	.707	.710	.709	.674	.680	.573
Sys. Aver.	.699			.709			.642			

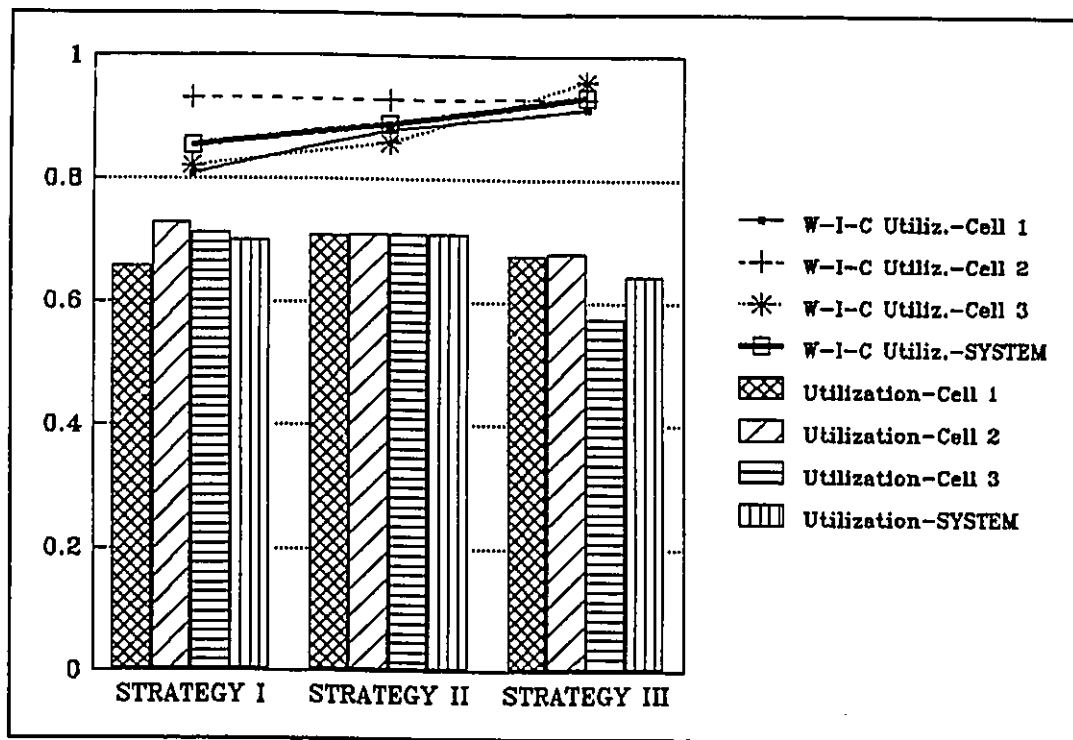


Figure 4.4 Impacts of Strategies on Cell and System Utilizations

Strategy I provides also good results, because part-mix shows smooth differentiation throughout periods. It gives the highest Inter-cell Movement Cost, and a value close to lowest one for Idle Time Cost. The utilization with Strategy I is reasonable good, because of less number of machines. However, less number of machines, without reallocation gives the highest Inter-cell Movement Cost, and lowest with-in cell utilization. Strategy II provides the lowest machine number, and also the lowest Idle Time Cost, and the highest system utilization. However, it does not provide highest with-in cell utilization, because of less number of machines. Strategy II could be the best strategy, if Cell Configuration Cost was less than 12 unit.

## 4.5 Test of Two-Stage Multi-Period PF/MCC Model

The purpose of this section is to observe the sensitivity of model to varying unit inter-cell movement cost values. The input data for part demands and machine requirements of parts, and processing times on those machines in the previous section is also used for the applications in this section. There are three applications of Strategy III, using Two-Stage Multi-Period PF/MCC Model. Planning horizon is 5 periods, and each period is 1500 unit time. In these three applications, all of the cost types are constant except Inter-cell Movement Cost. The model is tested with 80 unit Subcontracting Cost, 40 Unit Idle Time Cost, 100 unit Cell Set-up Cost, 10 Unit for Machine Purchasing Cost, and  $1/5$ ,  $1/10$ , and  $1/20$  Inter-cell Movement Cost. The results of applications are listed at Table 4.30, and graphical presentation of results are in Figure 4.5.

When Inter-cell Movement Cost per unit time increases, number of machines in the system which means Capital Investment Cost increases. Since, Subcontracting Cost is also reasonable high, the system must process most of the parts, and to be able to process these parts. If Inter-cell Movement Cost is high, the only solution is to purchase more machines. That is why number of machines in the system is increasing, from 16 to 22, while Inter-cell Movement Cost is increasing from  $1/20$  to  $1/5$  unit.

**Table 4.30 Results of Applications with IMC 1/5 - 1/10 - 1/20**

Cost Type	Periods	M 1/20	M 1/10	M 1/5
Cell Configuration Cost	Period 1	300	300	300
	2	0	0	0
	3	0	0	0
	4	0	0	0
	5	0	0	0
	Total	300	300	300
Machine Procurement Cost	Period 1	160	180	220
	2	0	0	0
	3	0	0	0
	4	0	0	0
	5	0	0	0
	Total	160	180	220
Capital Investment Cost	Period 1	240	270	330
	2	240	270	330
	3	240	270	330
	4	240	270	330
	5	240	270	330
	Total	1200	1350	1650
Idle Time Cost	Period 1	207	282	319
	2	214	289	401
	3	156	231	381
	4	177	220	341
	5	200	247	383
	Total	954	1269	1825
Intercell-Movement Cost	Period 1	74	52	0
	2	84	77	48
	3	122	126	75
	4	111	143	60
	5	130	153	0
	Total	521	551	183
Part Subcontracting Cost	Period 1	0	0	0
	2	0	0	0
	3	0	0	0
	4	160	80	0
	5	240	240	160
	Total	400	320	160
<b>TOTAL COST</b>	<b>Total</b>	<b>3535</b>	<b>3970</b>	<b>4338</b>

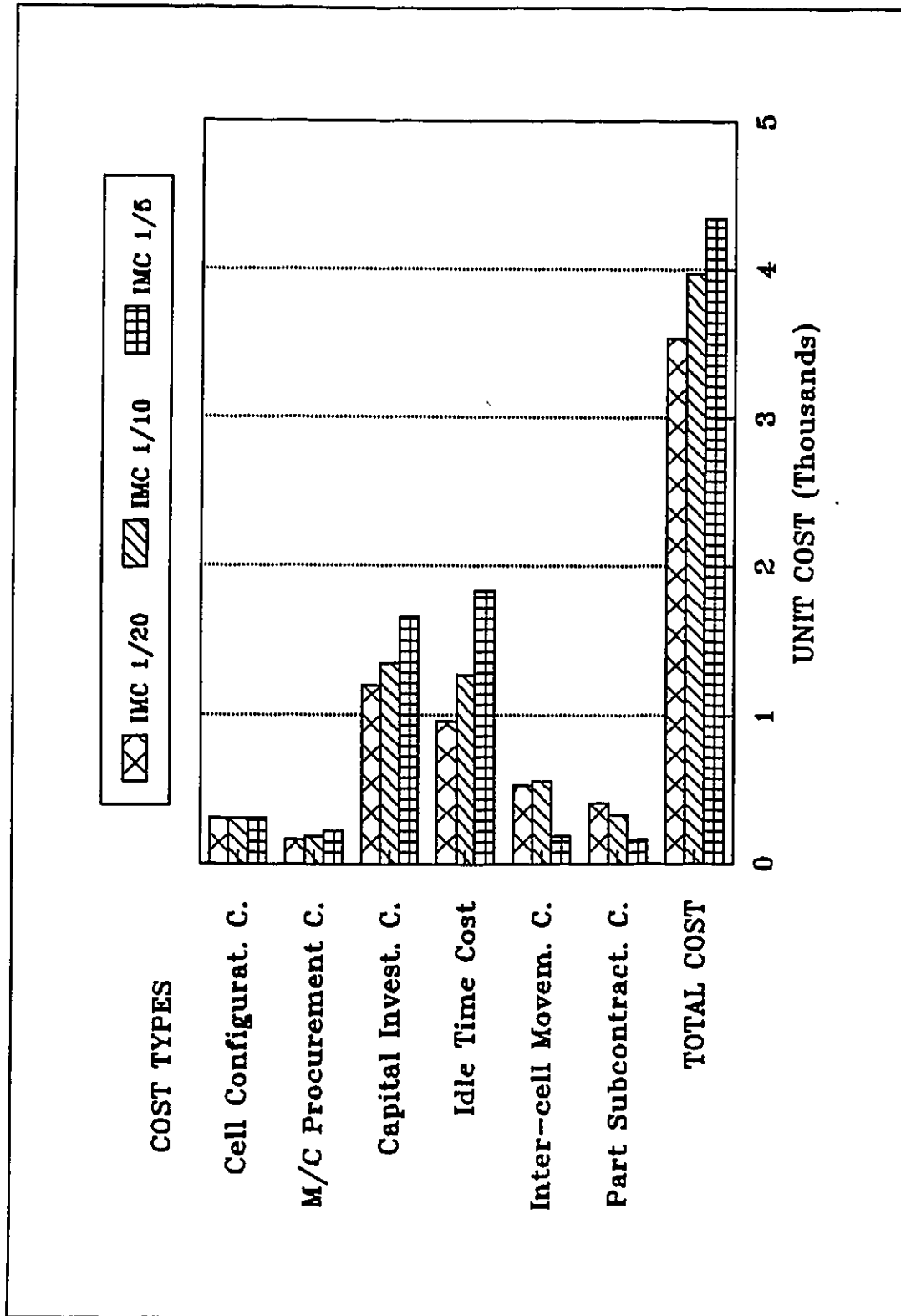


Figure 4.5 Impacts of different Inter-cell Movement Costs on system

Idle Time Cost which means idle time in the system is also increasing, while inter-cell movement cost per unit time is increasing. When, unit idle time cost becomes much less than unit Inter-cell Movement Cost, Capital Investment Cost is not high, and Subcontracting Cost is reasonable high, Idle Time Cost would increase with the number Capital Investment Cost.

Inter-cell movement is naturally decreasing, while unit Inter-cell Movement Cost is increasing. However, in terms of cost, this trend is not observed, because of varying unit inter-cell movement cost. Part Subcontracting Cost is decreasing, while unit inter-cell movement cost is increasing. The main reason of this negative correlation is low Capital Investment Cost, compared with Part Subcontracting Cost. High unit inter-cell movement cost makes the system to have more machine, and this causes less amount of Subcontracted Part. If Subcontracting Cost decreases from current level, and Capital Investment Cost increases from current level, than number of Subcontracting Parts would increase, instead of decreasing like the current situation in the examples.

Total Cost is increasing, while unit Inter-cell Movement Cost is increasing, since Machine Purchasing Cost, Capital Investment Cost, and Idle Time Costs are constantly increasing, while Part Subcontracting Cost and Inter-cell Movement Cost are decreasing in small amounts compared with other cost values.

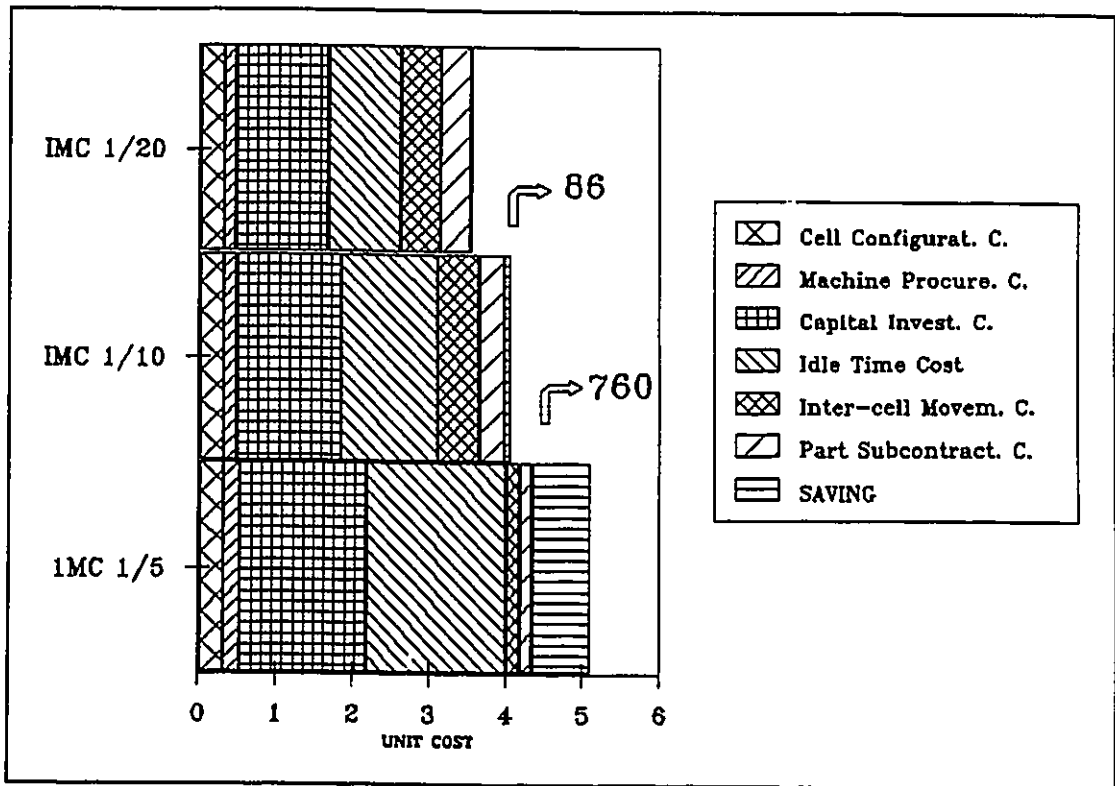
An increase in any kind of cost would increase the Total Cost. However, the Multi-Period model, tries to reduce the effects by changing the machine allocation. If the model did not change the system, when unit Inter-cell Movement Cost increases from 1/20 to 1/10, the Total Cost would be 4056 unit instead of 3970 unit, or when unit Inter-cell Movement Cost increases from 1/10 to 1/5, the Total Cost would be 4521 instead of current value 4338 unit, as indicated at Table 4.31.

A decrease in any kind of cost would decrease the Total Cost. However, the Multi-Period PF/MCC Model, tries to increase the effects in positive by changing the machine allocation. If the model does not change when unit Inter-cell Movement Cost decreases from 1/5 to 1/10, the Total Cost would be 4257 unit instead of 3970 unit, or when unit Inter-cell Movement Cost decrease from 1/10 to 1/20, the Total Cost would be 3694.5 instead of current value 3535 unit, as indicated at Table 4.31. Savings are also presented in Figure 4.6.

These results show that Multi-Period Model PF/MCC can provide different cellular configurations with different cost values, and its sensitive to the cost characteristics of the systems, while creating the machine cells.

**Table 4.31 Savings of Multi-Period PF/MCC Model**

Description	With the existing layout of ICM 1/20			With the existing layout of ICM 1/10			With the existing layout of ICM 1/5		
	ICM 1/20	ICM 1/10	ICM 1/5	ICM 1/20	ICM 1/10	ICM 1/5	ICM 1/20	ICM 1/10	ICM 1/5
With the ICM COST of existing layout	3535	4056	5098	3695	3970	4521	4293	4257	4338
With the original ICM Cost	3535	3970	4338	3535	3970	4338	3535	3970	4338
SAVINGS	-	86	760	160	-	183	758	287	-



**Figure 4.6 Savings of the Multi-Period Model**

## **CHAPTER 5**

### **CONCLUDING REMARKS & FUTURE RESEARCH**

#### **5.1 Concluding Remarks**

In this study, problems arising during implementation stage of Cellular Manufacturing Systems, such as the configuration and reconfiguration problems that are exist in many manufacturing environment were discussed. The literature review did not reveal any mathematical model that can be used to solve the reconfiguration problems associated with manufacturing cells under multi-period consideration.

A single-period and a multi-period mixed-integer mathematical models were developed for simultaneous part and machine grouping and seeking tradeoffs between Cell configuration, machine procurement and sale, capital investment, inter-cell movement and part subcontracting costs, in Chapter 3. A heuristic was developed to reduce the number of variables and constraints in mathematical models. The heuristic is based on the criteria of maximum similarity and minimum machine number in the system to form machine cells and part families.

The heuristic phase is integrated with a mathematical program, which generates the data inputs for LINDO, which, is considered as a two stage model. The first stage consists of the heuristics and the second stage consists of the mathematical program that optimizes various costs. The heuristics are modified slightly to take into consideration the multi-period problem. Furthermore, another mathematical program is developed and integrated with the modified heuristics to solve multi-period machine cell configuration. Again, results of the heuristics are prepared as an input to LINDO to perform the optimization stage.

Three Strategies are also presented to investigate the importance of Multi-Period Planning. Two of them are commonly used strategies one of which is to create a cellular configuration considering a period and push it as long as possible, and the other one of which is to create a cellular configuration considering one period and change the system, if optimum machine allocation changes. The proposed strategy is to create the cellular configuration considering more than one period, and not to change for those pre-determined periods.

The proposed strategy is compared with other strategies, and application examples, which include 3 and 5 periods are presented in Chapter 4. Possible results that strategies may give were presented. Strategy III always gave the best or one of the best answer, depending on input data. It always guarantees the lowest value in terms

of Total Cost. It was also observed that, Strategy III brings flexibility to the system, with lower system utilization and higher with-in cell utilization. The behaviour of the model tested with three different Inter-cell Movement Cost, and It is concluded that, the model is sensitive to changing cost values, and gives a solution same or better than the arising costs with the current configuration.

## **5.2 Future Research**

The developed heuristic and mathematical models may be inefficient for large scale problems, with the current software in market. The heuristic can be developed for machine assignment to cells, since MS-MM heuristic gives possible machines that can be exist in the cells. However, it does not mean that it gives the optimum solution. Instead of using a mathematical programming methods, developed heuristics would answer the problem in short time.

Tooling and inter-cell movement cost types could be included in the models directly with cost values or indirectly with similarity values. These additional costs would increase the number of variables. However, with heuristics, this would not be a problem.

Some equations can be developed to reflect various cost trade offs. It is possible to

integrate this step into another heuristics, so that a large size problem can be solved without the use of commercial optimization codes. Additional Studies can be also done to find out the optimal life of each machine cell separately considering various cost parameters. Moreover, these strategies can be modified to take into consideration multi-period problems.

## REFERENCES

- Askin, R.G., Cresswell, S.H., Goldberg, J.B. and Vakharia, A.J. (1991). A Hamiltonian Path Approach To Reordering the Part-Machine Matrix for Cellular Manufacturing, *International Journal of Production Research*, Vol 29, No 6, pp 1081-1100.
- Baran, J.J. (1991). Tips on Tackling GT-Based Cells, *Manufacturing Engineering*, February, pp 46-49.
- Bhat, M.V., Haupt, A. (1976). An Efficient Clustering Algorithm, *IEEE Transaction on Systems, Man, and Cybernetics*, Vol SMC-6, No 1, pp 61-64.
- Boe, W.J. and Cheng, C.H. (1991). A Close Neighbour Algorithm for Designing Cellular Manufacturing Design, *International Journal of Production Research* , Vol 29, No 10, pp 2097-2116.
- Burbidge, J.L. (1969). An Introduction of Group Technology, *Proceedings of Seminar in Group Technology*, Turbin.
- Burbidge, J.L. (1971). Production Flow Analysis, *The Production Engineer*, April, pp 139-152.
- Burbidge, J.L. (1975). *The Introduction of Group Technology*, John Wiley & Sons, New York.
- Burbidge, J.L. (1979). *Group Technology in the Engineering Industry*, Mechanical Engineering Publications, London, Chapter 6.
- Canada, J.R. and Sullivan, W.G. (1989). *Economic and Multiattribute Evaluation of Advanced Manufacturing Systems*, Prentice Hall, New Jersey, pp 66-81
- Carrie, A. (1975). Numerical Taxamony Applied to Group Technology and Plant Layout, *International Journal of Production Research*, Vol 13, No 6, pg 541-557.

- Chan, H.M. and Milner, D.A. (1982).** Direct CLustering Algorithm for Group Formation in Cellular Manufacture, *Journal of Manufacturing Systems*, Vol 1, No 1. pg 65-74.
- Rajagopalan, R. and Chandrasekan, M.P. (1986).** MODROC: An Extension of Rank Order Clustering for Group Technology, *International Journal of Production Research*, Vol 24, No 5, pg 1221-1223.
- Choobineh, F. (1988).** A Framework for the Design of Cellular Manufacturing Systems, *International Journal of Production Research*, Vol 26, No 7, pp 1161-1172.
- Chu, C.H. and Tsai, M. (1990).** A comparison of three array-based clustering techniques for Manufacturing Cell Formation, *International Journal of Production Research*, Vol 28, No 8, pp 1417-1433.
- Co, H.C. and Arar, A. (1988).** Configuring Cellular Manufacturing Systems, *International Journal of Production Research*, Vol 26, No 9, pp 1511-1522.
- De Beer, C. and De Witte, J. (1978).** Production Flow Synthesis, *CIRP Annals*, Vol 27, pp 389-392.
- De Witte, H. (1980).** The Use of Similarity Coefficients in Production Flow Analysis, *International Journal of Production Research*, Vol 18, pp 503-514.
- El-Assawy, I.F.K. and Torrance, J. (1972).** Component Flow Analysis: An Effective Approach to Production Systems Design, *The Production Engineer*, Vol 51, pp 165-170.
- Gomory, R.E. and Hu, T.C. (1971).** Multi-terminal network flows, *SIAM Journal of Applied Mathematics*, Vol 9, pp 551-571.
- Gu, P. (1991).** Process-based Machine Grouping for Cellular Manufacturing Systems, *Computers in Industry*, Vol 17, pg 9-17.
- Gunasingh, K.R. and Lashkari, R.S. (1989b).** Machine Grouping in Cellular Manufacturing Systems- An Integer Programming Approach, *International Journal of Production Research*, Vol 27, No 9, pp 1465-1473.
- Gunasingh, K.R. and Lashkari, R.S. (1991).** Simultaneous Grouping of Parts and Machines in Cellular Manufacturing Systems-An Integer Programming Approach, *Computers and Industrial Engineering*, Vol 20, No 1, pp 111-117.
- Gupta, T. and Hamid Seiffoddini (1990).** Clustering Algorithm for the Design of A Cellular Manufacturing System- An Analysis for Other Performance, *Computers and Industrial Engineering*, Vol 19, No 1-4, pp 432-436.

- Hilger, J. (1991).** Design of Manufacturing Cells: The Inter-cell Traffic, *ASME, PED-Vol 53, Design, Analysis, and Control of Manufacturing Cells*, pg 197-204.
- Jain,A.K., Kasilingam,R.G. and Bhole, S.D. (1991).** Joint Consideration of Cell Formation and Tool Provisioning Problems in Flexible Manufacturing Systems, *Computers and Industrial Engineering*, Vol 20, No 2, pp 271-277.
- King, J.R. (1980).** Machine Component Grouping Using POC Algorithm, *International Journal of Production Research*, April, pp 213-231.
- King, J.R., and Nakornachai, V. (1982).** Machine-Component Group Formation in Group Technology: Review and Extension, *International Journal of Production Research*, Vol 20, No 2, pp 117-133.
- Kusiak, A. (1985).** The Part-Families Problem in Flexible Manufacturing Systems, *Annals of Operational Research*, Vol 3, pp 279-300.
- Kusiak, A. (1987).** The Generalized Group Technology Concept, *International Journal of Production Research*, Vol 25, No 4, pp 561-569.
- Kusiak, A. and Cheng, C.H. (1990).** A Branch and Bound Algorithm for Solving the Group Technology Problem, *Annals of Operations Research*, Vol 26, pp 415-431.
- Kusiak, A. and Chow, W.S. (1987a).** An Efficient Cluster Identification Algorithm, *IEEE Transactions on Systems, Man, and Cybernetics*, Vol SMC-17, No 4, July/August, pp 696-699.
- Kusiak, A. and Chow, W.S. (1987b).** Efficient Solving of the Group Technology Problem, *Journal of Manufacturing Systems*, Vol 6, pp 117-124.
- Kusiak, A., Vannelli, A. and Kumar, K.R. (1986).** Clustering, Analysis: Models and Algorithms, *Control and Cybernetics*, Vol 15, No 2, pp 140-153.
- Lee, C.S. and Hwang, H. (1991).** A Hierarchical Divisive Clustering Method for Machine-Component Grouping Problems, *Engineering Optimization*, Vol 17, pp 65-78.
- Liang, M. and Taboun, S.M. (1992).** Part Selection and Part Assignment in Flexible Manufacturing Systems with Cellular Layout, *Proceedings of 14th Confernce on Computers and Industrial Engineering*, March, Orlando, FL.
- Logendran, R. (1990).** A Workload Based Model for Minimizing Total Intercell and Intracell Moves in Cellular Manufacturing, *International Journal of Production Research*, Vol 28, No 5, pp 914-925.

- Logendran, R. (1991).** Effect of the Identification of Key Machines in the Cell Formation Problem of Cellular Manufacturing Systems, *Computers and Industrial Engineering*, Vol 20, No 4, pp 439-449.
- Logendran, R. and West, T.M. (1990).** A Machine-Part Based Grouping Algorithm in Cellular Manufacturing, *Computers and Industrial Engineering*, Vol 19, No 1-4, pp 57-61.
- McAuley, J. (1972).** Machine Grouping for Efficient Production, *The Production Engineering*, Vol 51, No 53, February, pp 53-57.
- McCormick, C.J., Sweitzer, C.J. and White, T.R. (1972).** Problem decomposition and Data Reorganization by a Clustering Technique, *Operations Research*, Vol 20, pp 993-1009.
- Minis, I., Harhalakis, G. and Jajadia, S. (1990),** Manufacturing Cell Formation with Multiple, Functionally Identical Machines, *Manufacturing Review*, Vol 3, No 4, December, page 252-261.
- Mosier, C.T. (1989).** An Experiment Investigating the Application of Clustering Procedures and Similarity Coefficients to the GT Machine Cell Formation Problem, *International Journal of Production Research*, Vol 27, No 10, pp 1811-1835.
- Nagi, R., Harhalakis, G. and Proth, J.N. (1990).** Multiple Poutings and Capacity Considerations in Group Technology Applications, *International Journal of Production Research*, Vol 28, No 12, pg 2243-2257.
- Okogbaa, O.G., Chen, M.-T., Changchit, C. and Shell, R.L. (1992).** Manufacturing System Cell Formation and Evaluation Using a New Inter-cell Flow Reduction Heuristic., *International Journal of Production Research*, Vol 30, No 5, pg 1101-1118.
- Rajagopalan, R. and Batra, J.L. (1975).** Design of Cellular Production Systems: A Graph-Theoretic Approach, *International Journal of Production Research*, Vol 13, No 6, pp 567-579.
- Rajamani, D., Singh, N. and Aneja, Y.P. (1990).** Integrated Design of Cellular Manufacturing Systems in the Presence of Alternative Process Plans, *International Journal of Production Research*, Vol 28, No 8, pp 1541-1554.
- Sankaran, S. (1990).** Multiple Objective Decision Making Approach to Cell Formation: A Goal Programming Model, *Mathl. Comput. Modelling*, Vol 13, No 9, pp 71-82.

**Sankaran, S. and Kasilingam, R.G. (1990).** An Integrated Approach to Cell Formation and Part Routing in Group Technology Manufacturing Systems, *Engineering Optimization*, Vol 16, pp 235-245.

**Seifoddini, H.K. and Wolfe, P.M. (1986).** Application of the Similarity Coefficient Method in Group Technology, *AIIE Transactions*, Vol 18, pp 271-277.

**Shafer, S.M., Kern, G.M. and Wei, J.C. (1992).** A Mathematical Programming Approach for Dealing with Exceptional Elements in Cellular Manufacturing, *International Journal of Production Research*, Vol 30, No 5, pp 1161-1165.

**Srinivasan, G. and Narendran, T.T., Mahedevan, B. (1990).** An Assignment Model for the Part-Families Problem in Group Technology, *International Journal of Production Research*, Vol 28, No 3, pp 145-152.

**Stephanou, S.E. and Spiegl, F. (1992).** *The Manufacturing Challenge: From Concept to Production*, Van Nostrand Reinhold Press, New York.

**Sundaram, R.M. (1989).** Cellular Manufacturing Systems Design with Alternate Routing Considerations, *Journal of Mechanical Working Technology*, Vol 20, pp 425-432.

**Taboun, S.M., Sharma, A. (1991).** A Weighted Index for the Design of Cellular Manufacturing Systems, *Computers and Industrial Engineering*, Vol 21, No 1-4, pp 273-277.

**Tilsley, R. and Lewis, F.A. (1977).** Flexible Cell Manufacturing Systems- A Realistic Approach, *CIRP Annals*, Vol 24, pp 269-271.

**Vakharia, A.J. and Wemmerlow, U. (1985).** A New Design Method for Cellular Manufacturing Systems, Proceedings of the IXth ICPR, pp 2357-2363.

**Wemmerlow, U. (1984).** Comments on 'Direct Clustering Algorithm for Group Formation in Cellular Manufacture.' Letter to the Editor, *Journal of Manufacturing Systems*, Vol 3, No 1, pp vii-ix.

**Wemmerlow, U. and Hyer, N.L. (1986).** Procedures for the Part/Machine Group Identification Problem in Cellular Manufacture, *Journal of Operations Management*, Vol 6, No 2, February, pp 125-148.

**Zhang, C. and Wang, H. (1991).** Concurrent Formation of Part Families & Machine Cells Based on the Fuzzy Set Theory, *Journal of Manufacturing Systems*, Vol 11, No 1, pp 61-67.

## APPENDIX I

### MS-MM METHOD

```
/*THIS PROGRAM DOES MACHINE-COMPONENT GROUPING WITH MS-MM  
METHOD
```

```
EXAMPLE : BURBIDGE's PROBLEM*/
```

```
/* MAIN PROGRAM */
```

```
#include <stdio.h>
```

```
#include <conio.h>
```

```
#include <stdlib.h>
```

```
#include <math.h>
```

```
#define LEN 45
```

```
#define BEN 45
```

```
#define EN 45
```

```
long int MAXIM,IND;
```

```
int PAR,I12,I13,PL,AH,OH,MO1,d1,d2,MAX,MAK,I,I7,I8,I9;
```

```
int PRN2[BEN][LEN],PET[BEN][LEN];
```

```
int PN[LEN],SET[BEN][LEN],PRN[BEN][LEN],PRN1[BEN][LEN];
```

```
int CS,
```

```
h,CIK[LEN],I1,I2,I3,I4,I5,I6,Y,i, j, k,m,n,o,p, K,g;
```

```
float MIX,MAXIM1,TOP,HOP,A,B,C,SL,WHY2,WHY,WHY1,
```

```
SIM1[LEN],SIM[LEN][LEN];
```

```
/*PART MACHINE MATRIX FOR INPUT*/
```

```
static M[44][17] = {  
    {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},  
    {0,0,0,0,0,0,1,1,1,0,1,0,0,0,0,0,0},  
    {0,0,1,0,0,0,1,0,1,1,0,0,0,0,1,0,1},
```

{0,0,0,0,0,0,0,0,1,0,0,1,0,1,0,0,0},  
{0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0},  
{0,0,0,0,1,1,0,0,0,0,0,0,0,0,1,0},  
{0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0},  
{0,0,0,1,0,0,1,0,0,0,0,0,0,0,0,0,1},  
{0,0,0,0,0,1,1,0,1,0,0,0,0,0,0,0,0},  
{0,0,0,0,1,1,0,0,1,0,0,1,0,0,0,0,0},  
{0,0,1,0,0,0,0,0,0,1,0,0,0,0,0,0,1},  
{0,0,0,0,0,0,0,0,1,0,0,0,1,0,0,0,0},  
{0,0,0,0,0,0,1,0,1,0,1,0,0,0,0,0,0},  
{0,0,0,0,0,0,1,1,0,0,1,0,0,0,0,0,0},  
{0,0,0,0,1,1,1,0,0,0,0,0,0,0,0,1,0},  
{0,0,0,0,0,1,0,0,1,0,0,0,0,0,0,0,0},  
{0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0},  
{0,0,0,1,0,0,1,0,0,0,0,0,0,0,0,1,0,0},  
{0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,1},  
{0,0,0,0,1,1,1,0,1,0,0,0,0,0,0,0,1,0},  
{0,0,0,0,0,0,0,0,0,1,0,0,1,0,0,0,0,0},  
{0,0,0,0,1,1,0,0,1,0,0,0,0,0,0,0,1,0},  
{0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0},  
{0,0,0,0,1,1,1,0,1,0,0,0,0,0,0,0,0,0},  
{0,0,0,0,0,0,0,0,0,1,0,0,1,1,1,0,0,0},  
{0,0,0,0,0,0,0,0,1,0,0,1,0,0,0,0,0,0},  
{0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0},  
{0,0,0,0,0,0,0,0,0,1,0,0,1,1,0,0,0,0},  
{0,0,1,0,0,0,0,0,0,1,1,0,0,0,0,0,0,0},  
{0,0,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0},  
{0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0},  
{0,0,0,0,0,0,0,0,0,1,0,1,0,0,0,0,0,0},  
{0,0,1,0,0,0,1,0,0,1,0,0,0,0,0,0,0,1},  
{0,0,0,0,0,1,1,0,0,0,0,0,0,0,0,0,1,0},  
{0,0,0,1,0,0,1,0,0,0,0,0,0,0,0,0,0,0},  
{0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0},  
{0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0},  
{0,1,1,0,0,0,1,0,1,1,0,0,0,0,0,0,0,1},  
{0,0,1,0,0,0,0,0,0,1,1,0,0,0,0,0,0,1},  
{0,0,0,0,0,0,1,0,0,0,1,0,0,0,0,0,0,0},  
{0,0,1,0,0,0,1,0,0,1,0,0,0,0,0,0,0,0},  
{0,0,0,0,0,1,0,0,1,0,0,0,0,0,0,0,1,0},  
{0,1,1,0,0,0,1,0,0,1,0,0,0,0,0,0,0,1},  
{0,0,0,0,0,1,1,0,1,0,0,0,0,0,0,0,1,0}};

```

main()
{
clrscr();

/*INPUT*/

PL=3;  /* Minimum Allowed Cell Size for Search */
K=16;  /* Machine Type Number in the System */
PAR=43; /* Part Type Number in the System */
SL=0.4 ; /* Minimum Similarity Level for Search */
CS= 4 ; /* Cell Size for Search */

/*INITIALIZATION*/

MAXIM1=0.0;
MAK=1000;
WHY=0.0;
WHY1=0.0;
WHY2=0.0;

/*BEGINNING OF SEARCH*/

while (SL <= 0.70)
{

/* CREATING SIMILARITY MATRIX */

for(i=1;i<=PAR;i++)
for(j=1;j<=PAR;j++) SIM[i][j]=0;

for(i=1;i<=PAR;i++)
{
for(j=1;j<=PAR;j++)
{
A=0.0;B=0.0;C=0.0;
for(k=1;k<=K;k++)
{
if ( M[i][k] == 1 && M[j][k] == 1) A=A+1;
if ( M[i][k] == 0 && M[j][k] == 1) B=B+1;
if ( M[j][k] == 0 && M[i][k] == 1) C=C+1;
}
SIM[i][j] = (2*A)/(2*A+B+C);
}
}
}
}

```

```

    }
}

/*HEURISTIC PART A*/

for(i=1;i<=PAR;i++) CIK[i]=1;
for(i=1;i<=PAR;i++) for(j=1;j<=PAR;j++) SET[i][j]=0;
k=0;
for(i=1;i<=PAR;i++)
if(CIK[i]!=0)
{
    k=k+1;
    g=1;
    SET[k][g]=i;
    CIK[i]=0;
    h=1;
    while (SET[k][h]!=0)
    {
        I1=SET[k][g];
        for(j=1;j<=PAR;j++)
        if (SIM[I1][j]>=SL && CIK[j]!=0)
        {
            g=g+1;
            SET[k][g]=j;
            CIK[j]=0;
        }
        h=h+1;
    }
}

I2=0;
for(j=1;j<=PAR;j++)
if(SET[j][1]!=0)
I2=I2+1;

for(j=1;j<=PAR;j++)
if(SET[j][1]!=0)
{
    I3=0;
    for(i=1;i<=PAR;i++)
    if(SET[j][i]!=0)
    I3=I3+1;
    PN[j]=I3;
}

```

```
}
```

```
/*COMPUTATION OF PART FAMILIES GREATER THAN MINIMUM CELL  
SIZE*/
```

```
I12=0;  
for(j=1;j<=PAR;j++)  
if(SET[j][1] != 0 && PN[j] >= PL )  
I12=I12+1;
```

```
/*IF CELL SIZE IS SIMILAR TO WHAT IS PRE-SET*/
```

```
if (I12 == CS )
```

```
{
```

```
/*HEURISTIC PART B*/
```

```
for(j=1;j<=PAR;j++) SIM1[j]=0;  
for(j=1;j<=PAR;j++)  
if(SET[j][1] != 0 && PN[j] < PL)  
{  
    for(i=PN[j];i>=1;i--)  
    {  
        I4=SET[j][i];  
        MIX=0.0;  
  
        for(k=1;k<=PAR;k++)  
        if(SET[k][1] != 0 && PN[k] >= PL)  
        {  
            I6=PN[k];  
            for(l=1;l<=K;l++)  
            PET[k][l]=0 ;  
            for(n=1;n<=PN[k];n++)  
            {  
                MO1=SET[k][n];  
                for(l=1;l<=K;l++)  
                if(M[MO1][l] == 1 )  
                PET[k][l]=1;  
            }  
  
            A=0.0;B=0.0;C=0.0;  
            for(o=1;o<=K;o++)  
            {
```

```

        if ( M[I4][o] == 1 && PET[k][o] == 1) A=A+1;
        if ( M[I4][o] == 0 && PET[k][o] == 1) B=B+1;
        if ( PET[k][o] == 0 && M[I4][o] == 1) C=C+1;
    }
    SIM1[k] = (2*A)/(2*A+B+C);
    if ( SIM1[k] > MAX )
    {
        MIX = SIM1[k];
        I8 = k;
        I9 = I6 + 1;
    }
}
SET[j][i] = 0;
SET[I8][I9] = I4;
PN[I8] = I9;
}
}
/*CALCULATION OF SIMILARITY IN SYSTEM */

I12 = 0;
for(j = 1; j <= PAR; j++)
if(SET[j][1] != 0 )
I12 = I12 + 1;
HOP = 0.0;
for(j = 1; j <= PAR; j++)
if(SET[j][1] != 0 )
{
    TOP = 0;
    for(k = 1; k <= (PN[j]-1); k++)
    for(l = k + 1; l <= PN[j]; l++)
    {
        d1 = SET[j][k];
        d2 = SET[j][l];
        TOP = TOP + SIM[d1][d2];
    }
    HOP = HOP + (2.0*TOP)/((PN[j] + 0.0) * (PN[j]-1.0));
}
HOP = HOP/(I12 + 0.0);

/*COMPARISON OF CURRENT SIMILARITY WITH THE MAXIMUM
SIMILARITY IN SYSTEM*/

```

```

if (HOP >= MAXIM1)
{
    for(i = 1; i <= PAR; i++)
    for(j = 1; j <= PAR; j++) PRN1[i][j] = 0;
    WHY1 = SL;
    for(j = 1; j <= PAR; j++)
    if (SET[j][1] != 0)
    {
        for(k = 1; k <= PAR; k++)
        if (SET[j][k] != 0)
            PRN1[j][k] = SET[j][k];
    }
    MAXIM1 = HOP;

/*FINDING OUT MACHINE NUMBER IN THE SYSTEM */

for(j = 1; j <= PAR; j++)
if (PRN1[j][1] != 0)
{
    I3 = 0;
    for(i = 1; i <= PAR; i++)
    if (PRN1[j][i] != 0)
        I3 = I3 + 1;
    PN[j] = I3;
}
for(j = 1; j <= PAR; j++)
for(l = 1; l <= K; l++)
    PET[j][l] = 0;
OH = 0;
for(j = 1; j <= PAR; j++)
if (SET[j][1] != 0)
{
    for(k = 1; k <= PN[j]; k++)
    {
        MO1 = SET[j][k];
        for(l = 1; l <= K; l++)
        if (M[MO1][l] == 1)
            PET[j][l] = 1;
    }
    AH = 0;
    for(i = 1; i <= K; i++)
    if (PET[j][i] == 1)
        AH = AH + 1;
}

```

```

        OH = OH + AH;
    }
    /*COMPARISON OF CURRENT M/C NUMBER IN THE SYSTEM WITH
    MINIMUM M/C NUMBER*/

    if ( OH < = MAK)
    {
        for(i = 1;i < = PAR;i + +)
        for(j = 1;j < = PAR;j + +) PRN2[i][j] = 0;
        WHY2 = SL;
        for(j = 1;j < = PAR;j + +)
        if(SET[j][1] != 0 )
        {
            for(k = 1;k < = PAR;k + +)
            if(SET[j][k] != 0 )
            PRN2[j][k] = SET[j][k];
        }
        MAK = OH;
    }
    }
    SL = SL + .01;
}
/*END OF SEARCH

PRINTOUT PREPERATION*/
for(j = 1;j < = PAR;j + +)
for(l = 1;l < = K;l + +)
PET[j][l] = 0 ;
OH = 0;
for(j = 1;j < = PAR;j + +)
if(PRN1[j][1] != 0 )
{
    for(k = 1;k < = PN[j];k + +)
    {
        MO1 = PRN1[j][k];
        for(l = 1;l < = K;l + +)
        if(M[MO1][l] == 1 )
        PET[j][l] = 1;
    }
}
for(j = 1;j < = PAR;j + +)
if(PRN1[j][1] != 0 )

```

```

{
    for(k = 1;k <= PAR;k + +)
    {
        if(PRN1[j][k] != 0 )
            printf ("%d ",PRN1 [j][k]);
    }
    printf ("\n");
}
for(j = 1;j <= PAR;j + +)
if(PRN1[j][1] != 0 )
{
    for(k = 1;k <= K;k + +)
    {
        if(PET[j][k] != 0 )
            printf ("%d ",k);
    }
    printf ("\n");
}
printf ("%f\n",WHY1);
printf ("%f\n",MAXIM1);
for(j = 1;j <= PAR;j + +)
if(PRN2[j][1] != 0 )
{
    for(k = 1;k <= PAR;k + +)
    {
        if(PRN2[j][k] != 0 )
            printf ("%d ",PRN2[j][k]);
    }
    printf ("\n");
}
printf ("%f\n",WHY2);
printf ("%d\n",MAK);
}
/*END OF PROGRAM*/

```

## APPENDIX II

### FIRST STAGE OF TWO-STAGE

### SINGLE-PERIOD PF/MCC MODEL

```
/*THIS PROGRAM PRODUCES LINDO INPUT FILES  
BY USING MS-MM METHOD FOR SINGLE-PERIOD PF/MCC MODEL*/
```

```
/* MAIN PROGRAM */  
#include <stdio.h>  
#include <conio.h>  
#include <stdlib.h>  
#include <math.h>  
/*****  
*****/  
#define LEN      22  /* # of parts in the system + 1 */  
#define BEN      7   /* # of periods in the system + 1 */  
#define SEN      14  /* # of machines in the system + 1 */  
#define KEN      5   /* # of expected cells in the system + 1 */  
/*****  
*****/  
  
int   YEK,MO1,I12,I13,PL,d1,d2,MAX,I,I7,I8,I9;  
int   PN[LEN],SET[LEN][LEN],PRN1[LEN][LEN],h,CIK[LEN];  
int   I1,I2,I3,I4,I5,I6,i,j,k,K,g,m,n,o,p;  
float P[LEN][SEN],MAXIM1, TOP,HOP,A,B,C,SL,WHY1,SIM[LEN][LEN],SI  
M1[LEN];  
int   PART,DUM1,S[BEN][KEN][LEN], SE[LEN][LEN],  
PN1[LEN],COST,ST,CN;  
int   DURA,PER,PES[LEN], PET[BEN][KEN][SEN],PET1[LEN][SEN],  
PAT[KEN][SEN],PEK[BEN][KEN],I;
```

```
float A,B,C,SL,R[BEN][LEN][SEN],MIX;
```

```
FILE *TET;
```

```
/*INPUT DATA IS IN PROGRAM-NO INTERFACE WITH SCREEN
```

```
MACHINE-COMPONENT MATRIX*/
```

```
int static M[21][13] = {  
{0,0,0,0,0,0,0,0,0,0,0,0,0},  
{0,1,0,0,1,1,0,0,1,0,0,1,0},  
{0,1,0,1,0,0,1,0,0,0,0,0,1},  
{0,0,1,1,0,1,0,0,0,1,1,0,0},  
{0,1,0,0,1,0,0,0,1,1,0,1,0},  
{0,0,1,1,0,0,0,1,0,1,1,0,0},  
{0,1,1,1,0,0,0,1,0,0,0,0,1},  
{0,0,0,0,1,1,0,0,1,1,0,1,0},  
{0,0,1,0,0,1,0,1,0,1,1,0,0},  
{0,1,0,1,0,0,1,1,0,0,0,0,1},  
{0,0,1,1,0,1,0,0,0,1,1,0,0},  
{0,1,0,0,1,0,0,0,1,1,0,1,0},  
{0,0,1,1,0,0,1,1,0,0,0,0,1},  
{0,0,1,0,0,0,1,1,0,1,1,0,0},  
{0,0,1,0,1,1,0,0,0,1,1,0,0},  
{0,0,0,0,1,1,0,0,1,0,0,1,0},  
{0,0,1,1,0,0,0,1,0,0,1,0,1},  
{0,0,1,1,0,0,1,0,1,0,0,0,1},  
{0,1,0,0,1,1,0,0,1,0,0,1,0},  
{0,0,0,1,0,0,1,1,0,0,0,1,1},  
{0,1,0,0,1,1,0,0,1,0,0,1,0}};
```

```
/*PROCESSING TIMES PER UNIT PART*/
```

```
float static P[21][13] = {  
{0,0,0,0,0,1.25,.75,0,0,.75,.75,0,0},  
{.75,.4,.75,0,.5,.6,.75,1.25,.7,0,0,1.2,0},  
{0,.5,0,.4,0,0,.9,0,0,0,0,0,.6},  
{0,0,.75,.6,0,.5,0,0,0,.4,.7,0,0},  
{0,.75,0,0,1.2,0,0,0,.8,.5,0,.4,0},  
{0,0,.75,.6,0,0,0,.7,0,.9,.5,0,0},  
{0,.4,.6,.75,0,0,0,.8,0,0,0,0,.75},  
{0,0,0,0,.5,.4,0,0,.7,.8,0,.6,0},
```

```

{0,0,.6,0,0,.5,0,.4,0,1.0,.6,0,0},
{0,.9,0,.75,0,0,.6,.4,0,0,0,0,.5},
{0,0,.75,.7,0,.6,0,0,0,1.0,.4,0,0},
{0,.5,0,0,.8,0,0,0,1.2,.6,0,1.25,0},
{0,0,.6,.4,0,0,.5,.9,0,0,0,0,.7},
{0,0,.75,0,0,0,.8,.4,0,.8,.5,0,0},
{0,0,.6,0,.75,1.0,0,0,0,.4,.6,0,0},
{0,0,0,0,.7,.9,0,0,.5,0,0,1.2,0},
{0,0,.4,.75,0,0,0,1.0,0,0,.6,0,.75},
{0,0,.5,.8,0,0,.4,0,.7,0,0,0,.6},
{0,.6,0,0,.8,.5,0,0,.9,0,0,.4,0},
{0,0,0,.75,0,0,1.2,1.25,0,0,0,.5,.6},
{0,.4,0,0,.7,.6,0,0,.6,0,0,.5,0}};

```

**/\*PART DEMANDS\*/**

```

int static Q[21][6] = {
{0,0,0,0,0,0},
{0,400,0,0,0,0},
{0,260,0,0,0,0},
{0,380,200,0,0,0},
{0,280,120,0,0,0},
{0,350,200,190,0,0},
{0,320,240,160,0,0},
{0,440,440,400,0,0},
{0,600,540,460,340,200},
{0,480,440,400,280,150},
{0,0,320,360,400,0},
{0,300,400,400,360,280},
{0,600,640,640,560,440},
{0,460,560,720,760,720},
{0,0,160,300,400,440},
{0,0,80,200,360,600},
{0,0,160,310,480,800},
{0,0,170,250,420,750},
{0,0,0,180,360,620},
{0,0,0,150,320,580},
{0,0,0,250,500,950}};

```

**/\*EXISTING MACHINES IN THE SYSTEM\*/**

```

int static EM[13] = {0,0,0,0,0,0,0,0,0,0,0,0,0};

```

```

main()
{
clrscr();

/*INITIALIZATION*/

DURA = 1500; /* DURATION OF A PERIOD*/
PART = 20; /* TOTAL PART NUMBER IN THE SYSTEM*/
PER = 2; /* PERIOD FOR SEARCH*/
YEK = PER;
COST = 15; /* COST FOR CAPITAL INVESTMENT */
ST = 10; /* COST FOR PURCHASING*/
PL = 3;
K = 12; /* MACHINE NUMBER IN THE SYSTEM*/
SL = 0.4; /* MINIMUM SIMILARITY LEVEL FOR SEARCH */
MAXIM1 = 0.0;
WHY1 = 0.0;

/*FILE CREATION*/

TET = fopen ("B:\SM1-60.C", "w");

/*MS - MM METHOD*/

while ( SL <= .7)
{
for(i = 1; i <= PART; i++)
for(j = 1; j <= PART; j++) SIM[i][j] = 0;

for(i = 1; i <= PART; i++)
{
for(j = 1; j <= PART; j++)
{
A = 0.0; B = 0.0; C = 0.0;
for(k = 1; k <= K; k++)
{
if ( M[i][k] == 1 && M[j][k] == 1) A = A + 1;
if ( M[i][k] == 0 && M[j][k] == 1) B = B + 1;
if ( M[j][k] == 0 && M[i][k] == 1) C = C + 1;
}
SIM[i][j] = (2 * A) / (2 * A + B + C);
}
}
}

```

```

}
/*****/
for(i = 1; i <= PART; i++) CIK[i] = 1;
for(i = 1; i <= PART; i++) for(j = 1; j <= PART; j++) SET[i][j] = 0;
/*****/
k = 0;
for(i = 1; i <= PART; i++)
if(CIK[i] != 0)
{
    k = k + 1;
    g = 1;
    SET[k][g] = i;
    CIK[i] = 0;
    h = 1;
    while (SET[k][h] != 0)
    {
        I1 = SET[k][g];
        for(j = 1; j <= PART; j++)
        if (SIM[I1][j] >= SL && CIK[j] != 0)
        {
            g = g + 1;
            SET[k][g] = j;
            CIK[j] = 0;
        }
        h = h + 1;
    }
}
/*****/
I2 = 0;
for(j = 1; j <= PART; j++)
if(SET[j][1] != 0)
I2 = I2 + 1;
/*****/
for(j = 1; j <= PART; j++) PN[j] = 0;
for(j = 1; j <= PART; j++)
if(SET[j][1] != 0)
{
    I3 = 0;
    for(i = 1; i <= PART; i++)
    if(SET[j][i] != 0)
    I3 = I3 + 1;
    PN[j] = I3;
}
}

```

```

/*****/

for(j= 1;j< =PART;j+ +) SIM1[j]=0;

for(j= 1;j< =PART;j+ +)
if(SET[j][1] != 0 && PN[j] < PL)
{
    for(i= PN[j];i> = 1;i--)
    {
        I4 = SET[j][i];
        MIX = 0.0;

        for(k= 1;k< =PART;k+ +)
        if(SET[k][1] != 0 && PN[k] > = PL)
        {
            I6 = PN[k];
            for(l= 1;l< =K;l+ +)
            PET1[k][l] = 0 ;
            for(n= 1;n< =PN[k];n+ +)
            {
                MO1 = SET[k][n];
                for(l= 1;l< =K;l+ +)
                if(M[MO1][l] == 1 )
                PET1[k][l] = 1;
            }

            A=0.0;B=0.0;C=0.0;
            for(o= 1;o< =K;o+ +)
            {
                if ( M[I4][o] == 1 && PET1[k][o] == 1) A=A+ 1;
                if ( M[I4][o] == 0 && PET1[k][o] == 1) B=B+ 1;
                if ( PET1[k][o] == 0 && M[I4][o] == 1) C=C+ 1;
            }
            SIM1[k] = (2*A)/(2*A+ B+ C);
            if ( SIM1[k] > MIX )
            {
                MIX = SIM1[k];
                I8 = k;
                I9 = I6 + 1;
            }
        }
    }
}

```

```

        SET[j][i] = 0;
        SET[I8][I9] = I4;
        PN[I8] = I9;
    }

}
/*****/
I12 = 0;
for(j = 1; j <= PART; j++)
if(SET[j][1] != 0)
I12 = I12 + 1;
/*****/
    HOP = 0.0;
    for(j = 1; j <= PART; j++)
    if(SET[j][1] != 0)
    {
        TOP = 0;
        for(k = 1; k <= (PN[j]-1); k++)
        for(l = k + 1; l <= PN[j]; l++)
        {
            d1 = SET[j][k];
            d2 = SET[j][l];
            TOP = TOP + SIM[d1][d2];
        }
        HOP = HOP + (2.0 * TOP) / ((PN[j] + 0.0) * (PN[j] - 1.0));
    }
    HOP = HOP / (I12 + 0.0);
/*****/
if (HOP > MAXIM1)
{
    for(i = 1; i <= PART; i++)
    for(j = 1; j <= PART; j++) PRN1[i][j] = 0;
    WHY1 = SL;
    for(j = 1; j <= PART; j++)
    if(SET[j][1] != 0)
    {
        for(k = 1; k <= PART; k++)
        if(SET[j][k] != 0)
        PRN1[j][k] = SET[j][k];
    }
}
MAXIM1 = HOP;
CN = I12;

```

```

for(l = 1;l <= PART;l + +) PN1[l] = PN[l];
}
SL = SL + .01;
}
/*****/
for(j = 1;j <= PART;j + +)
if(PRN1[j][1] != 0 )
{
    for(k = 1;k <= PART;k + +)
    {
        if(PRN1[j][k] != 0 )
            printf ("%d ",PRN1[j][k]);
    }
printf ("\n");
}
printf ("%f\n",WHY1);
printf ("%f\n",MAXIM1);
/*****/
    for(i = 1;i <= PART;i + +) for(j = 1;j <= PART;j + +) SE[i][j] = 0;

    for(i = 1;i <= PART;i + +) PES[i] = 0;
    l = 0;
    for(i = 1;i <= PART;i + +)
    if (PRN1[i][1] != 0)
    {
        l = l + 1;
        for(j = 1;j <= PART;j + +)
        {
            SE[l][j] = PRN1 [i][j];
            PES[l] = PN1[i];
        }
    }
/*****/
for(i = 1;i <= PER;i + +) for(j = 1;j <= CN;j + +) PEK[i][j] = 0;
for(l = 1;l <= PER;l + +)
{
    for(i = 1;i <= CN;i + +)
    {
        m = 0;
        for(j = 1;j <= PES[i];j + +)
        {
            DUM1 = SE[i][j];
            if ( Q[DUM1][l] != 0 )

```

```

        {
            m = m + 1;
            S[I][i][m] = DUM1;
        }
    }
    PEK[I][i] = m;
}
}
/*****
for(m = 1; m <= PER; m++)
for(j = 1; j <= CN; j++)
for(l = 1; l <= K; l++)
PET[m][I][j] = 0;
for(m = 1; m <= PER; m++)
{
    for(j = 1; j <= CN; j++)
    for(k = 1; k <= PEK[m][j]; k++)
    {
        MO1 = S[m][j][k];
        for(l = 1; l <= K; l++)
        if(M[MO1][I] == 1)
            PET[m][j][l] = 1;
    }
}
}
/*LINDO INPUT FILE CREATION
OBJECTIVE FUCTION*/

fprintf(TET, "min Z\n");
fprintf(TET, "st\n");
fprintf(TET, "Z1 + Z2 + Z3 + Z4 + Z5 + Z6 - Z = 0\n");
fprintf(TET, "100CN - Z1 = 0\n");
for(i = YEK; i <= PER; i++)
for(j = 1; j <= K; j++)
PAT[k][j] = 0;
for(i = YEK; i <= PER; i++)
for(j = 1; j <= K; j++)
for(k = 1; k <= CN; k++)
PAT[k][j] = PAT[k][j] + PET[i][k][j];
for(j = 1; j <= K; j++)
{
    for(k = 1; k <= CN; k++)
    if ( PAT[k][j] >= 1)

```

```

        fprintf(TET,"%dY%d%d + ",COST,j,k);
        fprintf(TET,"-Z2 = 0 \n");
    }
    /*****/
    for(i = 1;i <= K;i++)
        fprintf(TET,"%dPUR%d + %dSEL%d + ",ST,i,ST,i);
        fprintf(TET,"-Z3 = 0\n");
    /*****/
    for(j = YEK;j <= PER;j++)
    {
        for(i = 1;i <= K;i++)
            fprintf(TET,"ID%d%d + ",i,j);
            fprintf(TET,"-Z4 %d = 0\n",j);
        }
        for(j = YEK;j <= PER;j++) fprintf(TET,"Z4 %d + ",j);
        fprintf(TET,"-4OZ4 = 0\n");
    /*****/
    for(i = YEK;i <= PER;i++)
    {
        for(j = 1;j <= K;j++)
        {
            for(k = 1;k <= CN;k++)
                if ( PET[i][k][j] == 1)
                    fprintf(TET,"M%d%d%d + ",j,k,i);
                    fprintf(TET,"\n");
            }
            fprintf(TET,"-Z5 %d = 0\n",i);
        }
        for(j = YEK;j <= PER;j++) fprintf(TET,"Z5 %d + ",j);
        fprintf(TET,"-2OZ5 = 0\n");
    /*****/
    for(m = YEK;m <= PER;m++)
    {
        for(j = 1;j <= PART;j++)
            if ( Q[j][m] != 0 )
            {
                fprintf(TET,"S%d%d + ",j,m);
            }
            fprintf(TET,"-Z6 %d = 0\n",m);
        }
        for(j = YEK;j <= PER;j++) fprintf(TET,"6OZ6 %d + ",j);
        fprintf(TET,"-Z6 = 0\n");

```

```

/*CONSTRAINT SETS*/
for(i=YEK;i<=PER;i++)
{
    for(l=1;l<=CN;l++)
    for(k=1;k<=PEK[i][l];k++)
    {
        MO1=S[i][l][k];
        fprintf(TET,"X%d%d%d+",MO1,l,i);
        fprintf(TET,"S%d%d=1\n",MO1,i);
    }
}
/*****/
fprintf(TET,"CN=%d\n",CN);
/*****/
for(l=YEK;l<=PER;l++)
for(j=1;j<=PART;j++)
for(i=1;i<=K;i++)
R[l][j][i]=Q[j][l]*M[j][i]*P[j][i];
for(i=YEK;i<=PER;i++)
for(j=1;j<=K;j++)
{
    for(k=1;k<=CN;k++)
    if ( PET[i][k][j] == 1)
    {
        for(l=1;l<=PEK[i][k];l++)
        {
            MO1=S[i][k][l];
            if ( M[MO1][j] == 1 && R[i][MO1][j] != 0.0 )
                fprintf (TET,"%.0fX%d%d%d+",R[i][MO1][j],MO1,k,i);
        }
    }

    fprintf(TET,"I%d%d%d-%dY%d%d-M%d%d%d=0\n",j,k,i,DURA,j,k,j,k,i);
}
/*****/
for(i=YEK;i<=PER;i++)
{
    for(j=1;j<=K;j++)
    {
        for(k=1;k<=CN;k++)
        if ( PET[i][k][j] == 1)
            fprintf(TET,"-M%d%d%d+I%d%d%d",j,k,i,j,k,i);
    }
}

```

```

        fprintf(TET, "-ID%d%d = 0\n", j, i);
    }
}
for(j = 1; j <= K; j++)
{
    for(k = 1; k <= CN; k++)
    if ( PAT[k][j] >= 1)
    fprintf(TET, "Y%d%d + ", j, k);
    fprintf(TET, "-EM%d-PUR%d + SEL%d = 0\n", EM[j], j, j);
}
/*****END of CONSTRAINTS*****/
fprintf (TET, "end\n");
/*****VARIABLES*****/
for(i = YEK; i <= PER; i++)
{
    for(l = 1; l <= CN; l++)
    for(k = 1; k <= PEK[i][l]; k++)
    {
        MO1 = S[i][l][k];
        fprintf(TET, "int X%d%d%d\n", MO1, l, i);
    }
}
/*****/

for(j = 1; j <= K; j++)
{
    for(k = 1; k <= CN; k++)
    if ( PAT[k][j] >= 1)
    fprintf(TET, "gin Y%d%d\n", j, k);
}
fprintf (TET, "leave");
}
/* END OF PROGRAM */

```

## APPENDIX III

### FIRST STAGE OF TWO-STAGE

### MULTI-PERIOD PF/MCC MODEL

**/\*THIS PROGRAM PRODUCES LINDO INPUT FILES  
BY USING MS-MM METHOD FOR MULTI-PERIOD PF/MCC MODEL \*/**

```
/* MAIN PROGRAM */
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
#include <math.h>

/*****/
#define LEN 22 /* # of parts in the system + 1 */
#define BEN 9 /* # of periods in the system + 1 */
#define SEN 14 /* # of machines in the system + 1 */
#define KEN 8 /* # of expected cells in the system + 1 */
/*****/
int MO1,I12,I13,PL,d1,d2,MAX,I,I7,I8,I9;
int PN[LEN],SET[LEN][LEN],PRN1[LEN][LEN],h,CIK[LEN];
int I1,I2,I3,I4,I5,I6,i,j,k,K,g,m,n,o,p;
float MIX,MAXIM1,TOP,HOP,A,B,C,SL,WHY1,SIM[LEN][LEN],SIM
1[LEN];
int PART,DUM1,S[BEN][KEN][LEN],SE[LEN][LEN],
PN1[LEN],COST,ST,Q[LEN][BEN],CN;
int DURA,PER,PES[LEN],PET[BEN][KEN][SEN],PET1[LEN][SEN],
PAT[KEN][SEN],PEK[BEN][KEN],I;
float A,B,C,SL,R[BEN][LEN][SEN];

FILE *TET;
```

**/\*INPUT DATA IS IN PROGRAM-NO INTERFACE WITH SCREEN  
MACHINE-COMPONENT MATRIX\*/**

```
int static M[21][13]={
{0,0,0,0,0,0,0,0,0,0,0,0,0},
{0,1,0,0,1,1,0,0,1,0,0,1,0},
{0,1,0,1,0,0,1,0,0,0,0,0,1},
{0,0,1,1,0,1,0,0,0,1,1,0,0},
{0,1,0,0,1,0,0,0,1,1,0,1,0},
{0,0,1,1,0,0,0,1,0,1,1,0,0},
{0,1,1,1,0,0,0,1,0,0,0,0,1},
{0,0,0,0,1,1,0,0,1,1,0,1,0},
{0,0,1,0,0,1,0,1,0,1,1,0,0},
{0,1,0,1,0,0,1,1,0,0,0,0,1},
{0,0,1,1,0,1,0,0,0,1,1,0,0},
{0,1,0,0,1,0,0,0,1,1,0,1,0},
{0,0,1,1,0,0,1,1,0,0,0,0,1},
{0,0,1,0,0,0,1,1,0,1,1,0,0},
{0,0,1,0,1,1,0,0,0,1,1,0,0},
{0,0,0,0,1,1,0,0,1,0,0,1,0},
{0,0,1,1,0,0,0,1,0,0,1,0,1},
{0,0,1,1,0,0,1,0,1,0,0,0,1},
{0,1,0,0,1,1,0,0,1,0,0,1,0},
{0,0,0,1,0,0,1,1,0,0,0,1,1},
{0,1,0,0,1,1,0,0,1,0,0,1,0}};
```

**/\*PROCESSING TIMES PER UNIT PART\*/**

```
float static P[21][13]={
{0,0,0,0,0,1.25,.75,0,0,.75,.75,0,0},
{.75,.4,.75,0,.5,.6,.75,1.25,.7,0,0,1.2,0},
{0.5,0,.4,0,0,.9,0,0,0,0,.6},
{0,0,.75,.6,0,.5,0,0,0,.4,.7,0,0},
{0,.75,0,0,1.2,0,0,0,.8,.5,0,.4,0},
{0,0,.75,.6,0,0,0,.7,0,.9,.5,0,0},
{0,.4,.6,.75,0,0,0,.8,0,0,0,.75},
{0,0,0,0,.5,.4,0,0,.7,.8,0,.6,0},
{0,0,.6,0,0,.5,0,.4,0,1.0,.6,0,0},
{0,.9,0,.75,0,0,.6,.4,0,0,0,0,.5},
{0,0,.75,.7,0,.6,0,0,0,1.0,.4,0,0},
{0,.5,0,0,.8,0,0,0,1.2,.6,0,1.25,0},
{0,0,.6,.4,0,0,.5,.9,0,0,0,0,.7},
```

```

{0,0,.75,0,0,0,.8,.4,0,.8,.5,0,0},
{0,0,.6,0,.75,1.0,0,0,0,.4,.6,0,0},
{0,0,0,0,.7,.9,0,0,.5,0,0,1.2,0},
{0,0,.4,.75,0,0,0,1.0,0,0,.6,0,.75},
{0,0,.5,.8,0,0,.4,0,.7,0,0,0,.6},
{0,.6,0,0,.8,.5,0,0,.9,0,0,.4,0},
{0,0,0,.75,0,0,1.2,1.25,0,0,0,.5,.6},
{0,.4,0,0,.7,.6,0,0,.6,0,0,.5,0}};

```

**/\*PART DEMANDS\*/**

```

int static Q[21][6] = {
{0,0,0,0,0,0},
{0,400,0,0,0,0},
{0,260,0,0,0,0},
{0,380,200,0,0,0},
{0,280,120,0,0,0},
{0,350,200,190,0,0},
{0,320,240,160,0,0},
{0,440,440,400,0,0},
{0,600,540,460,340,200},
{0,480,440,400,280,150},
{0,0,320,360,400,0},
{0,300,400,400,360,280},
{0,600,640,640,560,440},
{0,460,560,720,760,720},
{0,0,160,300,400,440},
{0,0,80,200,360,600},
{0,0,160,310,480,800},
{0,0,170,250,420,750},
{0,0,0,180,360,620},
{0,0,0,150,320,580},
{0,0,0,250,500,950}};

```

**/\*EXISTING MACHINES IN THE SYSTEM\*/**

```

int static EM[13] = {0,0,0,0,0,0,0,0,0,0,0,0,0};
/*****/
main()
{
clrscr();
/*****/

```

```

DURA = 1500; /*DURATION OF EACH PERIOD*/
PART = 20; /*NUMBER OF PARTS IN SYSTEM*/
PER = 5; /*NUMBER OF PERIOD IN THE SYSTEM*/
COST = 75; /*CAPITAL INVESTMENT COST FOR FIVE YEAR*/
ST = 10; /*MACHINE PURCHASING COST*/
PL = 4;
K = 12; /*NUMBER OF MACHINES IN THE SYSTEM*/
SL = 0.4;
MAXIM1 = 0.0;
WHY1 = 0.0;

/*CREATING A FILE*/
TET = fopen ("MTR-80.C", "w");

/*BEGINNING OF MS-MM METHOD*/
while ( SL <= .7)
{
for(i = 1; i <= PART; i++)
for(j = 1; j <= PART; j++) SIM[i][j] = 0;
for(i = 1; i <= PART; i++)
{
for(j = 1; j <= PART; j++)
{
A = 0.0; B = 0.0; C = 0.0;
for(k = 1; k <= K; k++)
{
if ( M[i][k] == 1 && M[j][k] == 1) A = A + 1;
if ( M[i][k] == 0 && M[j][k] == 1) B = B + 1;
if ( M[j][k] == 0 && M[i][k] == 1) C = C + 1;
}
SIM[i][j] = (2 * A) / (2 * A + B + C);
}
}
}
/*****/
for(i = 1; i <= PART; i++) CIK[i] = 1;
for(i = 1; i <= PART; i++) for(j = 1; j <= PART; j++) SET[i][j] = 0;
/*****/
k = 0;
for(i = 1; i <= PART; i++)
if(CIK[i] != 0)
{
k = k + 1;
}

```

```

g = 1;
SET[k][g] = i;
CIK[i] = 0;
h = 1;
while (SET[k][h] != 0 )
{
    I1 = SET[k][g];
    for(j = 1; j <= PART; j++)
    if (SIM[I1][j] >= SL && CIK[j] != 0)
    {
        g = g + 1;
        SET[k][g] = j;
        CIK[j] = 0;
    }
    h = h + 1;
}
}
/*****/
I2 = 0;
for(j = 1; j <= PART; j++)
if(SET[j][1] != 0 )
I2 = I2 + 1;
/*****/
for(j = 1; j <= PART; j++) PN[j] = 0;
for(j = 1; j <= PART; j++)
if(SET[j][1] != 0 )
{
    I3 = 0;
    for(i = 1; i <= PART; i++)
    if(SET[j][i] != 0 )
    I3 = I3 + 1;
    PN[j] = I3;
}
/*****/
for(j = 1; j <= PART; j++) SIM1[j] = 0;
for(j = 1; j <= PART; j++)
if(SET[j][1] != 0 && PN[j] < PL)
{
    for(i = PN[j]; i >= 1; i--)
    {
        I4 = SET[j][i];
        MIX = 0.0;
        for(k = 1; k <= PART; k++)

```

```

if(SET[k][1] != 0 && PN[k] >= PL)
{
    I6 = PN[k];
    for(l = 1; l <= K; l++)
    PET1[k][l] = 0;
    for(n = 1; n <= PN[k]; n++)
    {
        MO1 = SET[k][n];
        for(l = 1; l <= K; l++)
        if(M[MO1][l] == 1)
        PET1[k][l] = 1;
    }
    A = 0.0; B = 0.0; C = 0.0;
    for(o = 1; o <= K; o++)
    {
        if ( M[I4][o] == 1 && PET1[k][o] == 1) A = A + 1;
        if ( M[I4][o] == 0 && PET1[k][o] == 1) B = B + 1;
        if ( PET1[k][o] == 0 && M[I4][o] == 1) C = C + 1;
    }
    SIM1[k] = (2 * A) / (2 * A + B + C);
    if ( SIM1[k] > MIX )
    {
        MIX = SIM1[k];
        I8 = k;
        I9 = I6 + 1;
    }
}
SET[j][i] = 0;
SET[I8][I9] = I4;
PN[I8] = I9;
}
}
/*****/
I12 = 0;
for(j = 1; j <= PART; j++)
if(SET[j][1] != 0)
I12 = I12 + 1;
/*****/
HOP = 0.0;
for(j = 1; j <= PART; j++)
if(SET[j][1] != 0)
{

```

```

        TOP=0;
        for(k=1;k<=(PN[j]-1);k++)
        for(l=k+1;l<=PN[j];l++)
        {
            d1=SET[j][k];
            d2=SET[j][l];
            TOP=TOP+SIM[d1][d2];
        }
        HOP=HOP+(2.0*TOP)/((PN[j]+0.0)*(PN[j]-1.0));
    }
    HOP=HOP/(I12+0.0);
    /*****
if (HOP > MAXIM1)
{
    for(i=1;i<=PART;i++)
    for(j=1;j<=PART;j++) PRN1[i][j]=0;
    WHY1=SL;
    for(j=1;j<=PART;j++)
    if(SET[j][1]!=0)
    {
        for(k=1;k<=PART;k++)
        if(SET[j][k]!=0)
        PRN1[j][k]=SET[j][k];
    }
    MAXIM1=HOP;
    CN=I12;
    for(l=1;l<=PART;l++) PN1[l]=PN[l];
}
SL=SL+.01;
}
/*BRIEF OUTPUT OF MS-MM METHOD ON SCREEN*/

for(j=1;j<=PART;j++)
if(PRN1[j][1]!=0)
{
    for(k=1;k<=PART;k++)
    {
        if(PRN1[j][k]!=0)
        printf ("%d ",PRN1[j][k]);
    }
    printf ("\n");
}
printf ("%f\n",WHY1);

```

```
printf ("%f\n",MAXIM1);
```

```
/*FINDING OUT PART TO BE PROCESSED IN EACH PERIOD*/
```

```
for(i = 1;i <= PART;i++) for(j = 1;j <= PART;j++) SE[i][j] = 0;
```

```
for(i = 1;i <= PART;i++) PES[i] = 0;
```

```
l = 0;
```

```
for(i = 1;i <= PART;i++)
```

```
if (PRN1[i][1] != 0)
```

```
{
```

```
    l = l + 1;
```

```
    for(j = 1;j <= PART;j++)
```

```
    {
```

```
        SE[l][j] = PRN1[i][j];
```

```
        PES[l] = PRN1[i];
```

```
    }
```

```
}
```

```
/*.....*/
```

```
for(i = 1;i <= PER;i++) for(j = 1;j <= CN;j++) PEK[i][j] = 0;
```

```
for(l = 1;l <= PER;l++)
```

```
{
```

```
    for(i = 1;i <= CN;i++)
```

```
    {
```

```
        m = 0;
```

```
        for(j = 1;j <= PES[i];j++)
```

```
        {
```

```
            DUM1 = SE[i][j];
```

```
            if ( Q[DUM1][i] != 0 )
```

```
            {
```

```
                m = m + 1;
```

```
                S[l][i][m] = DUM1;
```

```
            }
```

```
        }
```

```
        PEK[l][i] = m;
```

```
    }
```

```
}
```

```
/*.....*/
```

```
for(m = 1;m <= PER;m++)
```

```
for(j = 1;j <= CN;j++)
```

```
for(l = 1;l <= K;l++)
```

```

PET[m][l][j] = 0 ;
for(m = 1;m <= PER;m + +)
{
    for(j = 1;j <= CN;j + +)
        for(k = 1;k <= PEK[m][j];k + +)
            {
                MO1 = S[m][j][k];
                for(l = 1;l <= K;l + +)
                    if(M[MO1][l] == 1 )
                        PET[m][j][l] = 1;
            }
}
/*LINDO INPUT FILE CREATION

OBJECTIVE FUNCTION*/

fprintf(TET,"min Z\n");
fprintf(TET,"st\n");
fprintf(TET,"Z1 + Z2 + Z3 + Z4 + Z5 + Z6-Z = 0\n");
fprintf(TET,"100CN-Z1 = 0\n");
for(i = YEK;i <= PER;i + +)
for(j = 1;j <= K;j + +)
PAT[k][j] = 0;
for(i = YEK;i <= PER;i + +)
for(j = 1;j <= K;j + +)
for(k = 1;k <= CN;k + +)
PAT[k][j] = PAT[k][j] + PET[i][k][j];
for(j = 1;j <= K;j + +)
{
    for(k = 1;k <= CN;k + +)
        if ( PAT[k][j] >= 1)
            fprintf(TET,"%dY%d%d + ",COST,j,k);
            fprintf(TET,"-Z2 = 0 \n");
}

/*****/

for(i = 1;i <= K;i + +)
fprintf(TET,"%dPUR%d + %dSEL%d + ",ST,i,ST,i);
fprintf(TET,"-Z3 = 0\n");

/*****/

```

```

for(j= 1;j< =PER;j+ +)
{
for(i= 1;i< =K;i+ +)
fprintf(TET,"ID%d%d+ ",i,j);
fprintf(TET,"-Z4%d=0\n",j);
}
for(j= 1;j< =PER;j+ +) fprintf(TET,"Z4%d+ ",j);
fprintf(TET,"-4OZ4=0\n");
/*****/
for(i= 1;i< =PER;i+ +)
{
for(j= 1;j< =K;j+ +)
{
for(k= 1;k< =CN;k+ +)
if ( PET[i][k][j] == 1)
fprintf(TET,"M%d%d%d+ ",j,k,i);
fprintf(TET,"\n");
}
}
fprintf(TET,"-Z5%d=0\n",i);
}
for(j= 1;j< =PER;j+ +) fprintf(TET,"Z5%d+ ",j);
fprintf(TET,"-2OZ5=0\n");

/*****/
for(m= 1;m< =PER;m+ +)
{
for(j= 1;j< =PART;j+ +)
if ( Q[j][m] != 0 )
{
fprintf(TET,"S%d%d+ ",j,m);
}
}
fprintf(TET,"-Z6%d=0\n",m);
}

for(j= 1;j< =PER;j+ +) fprintf(TET,"8OZ6%d+ ",j);
fprintf(TET,"-Z6=0\n");

/*CONSTRAINT SETS*/

for(i= 1;i< =PER;i+ +)
{
for(l= 1;l< =CN;l+ +)

```

```

for(k = 1;k <= PEK[i][l];k + +)
{
    MO1 = S[i][l][k];
    fprintf(TET,"X%d%d%d + ",MO1,l,i);
    fprintf(TET,"S%d%d = 1\n",MO1,i);
}
}
/*****/
fprintf(TET,"CN = %d\n",CN);
/*****/
for(l = 1;l <= PER;l + +)
for(j = 1;j <= PART;j + +)
for(i = 1;i <= K;i + +)
R[l][j][i] = Q[j][l] * M[j][i] * P[j][i];
for(i = 1;i <= PER;i + +)
for(j = 1;j <= K;j + +)
{
    for(k = 1;k <= CN;k + +)
    if ( PET[i][k][j] == 1)
    {
        for(l = 1;l <= PEK[i][k];l + +)
        {
            MO1 = S[i][k][l];
            if ( M[MO1][j] == 1 && R[i][MO1][j] != 0.0 )
                fprintf (TET,"%.0fX%d%d%d + ",R[i][MO1][j],MO1,k,i);
        }
    }

fprintf(TET,"I%d%d%d-%dY%d%d-M%d%d%d = 0\n",j,k,i,DURA,j,k,j,k,i);
}
}
/*****/
for(i = 1;i <= PER;i + +)
{
    for(j = 1;j <= K;j + +)
    {
        for(k = 1;k <= CN;k + +)
        if ( PET[i][k][j] == 1)
            fprintf(TET,"-M%d%d%d + I%d%d%d",j,k,i,j,k,i);
        fprintf(TET,"-ID%d%d = 0\n",j,i);
    }
}
for(j = 1;j <= K;j + +)

```

```

{
    for(k = 1;k <= CN;k + +)
        if ( PAT[k][j] >= 1)
            fprintf(TET,"Y%d%d + ",j,k);
            fprintf(TET,-EM%d-PUR%d + SEL%d = 0\n",EM[j],j,j);
}
/*****
fprintf (TET,"end\n");

/*****VARIABLES*****/
for(i = 1;i <= PER;i + +)
{
    for(l = 1;l <= CN;l + +)
        for(k = 1;k <= PEK[i][l];k + +)
        {
            MO1 = S[i][l][k];
            fprintf(TET,"int X%d%d%d\n",MO1,l,i);
        }
}
/*****
for(j = 1;j <= K;j + +)
{
    for(k = 1;k <= CN;k + +)
        if ( PAT[k][j] >= 1)
            fprintf(TET,"gin Y%d%d\n",j,k);
}
/*****
fprintf (TET,"leave");
}

/* END OF PROGRAM */

```

## APPENDIX IV

### EXAMPLE LINDO INPUT FILES FOR SINGLE-PERIOD PF/MCC MODEL AND TWO-STAGE SINGLE-PERIOD PF/MCC MODEL

THIS LINDO INPUT FILE IS CREATED BY USING SINGLE-PERIOD MODEL  
FOR A CASE OF 7 PARTS, 6 MACHINES

```
min Z
st
Z1 + Z2 + Z3 + Z4 + Z5 + Z6 - Z = 0
Z1 = 200
10Y11 + 10Y12 + 10Y13 + 10Y14 + 10Y15 + 10Y16 + 10Y17 +
10Y21 + 10Y22 + 10Y23 + 10Y24 + 10Y25 + 10Y26 + 10Y27 +
10Y31 + 10Y32 + 10Y33 + 10Y34 + 10Y35 + 10Y36 + 10Y37 +
10Y41 + 10Y42 + 10Y43 + 10Y44 + 10Y45 + 10Y46 + 10Y47 +
10Y51 + 10Y52 + 10Y53 + 10Y54 + 10Y55 + 10Y56 + 10Y57 +
10Y61 + 10Y62 + 10Y63 + 10Y64 + 10Y65 + 10Y66 + 10Y67 - Z2 = 0
10PUR1 + 10SEL1 + 10PUR2 + 10SEL2 + 10PUR3 + 10SEL3 +
10PUR4 + 10SEL4 + 10PUR5 + 10SEL5 + 10PUR6 + 10SEL6 - Z3 = 0
ID1 + ID2 + ID3 + ID4 + ID5 + ID6 - 40Z4 = 0
M11 + M12 + M13 + M14 + M15 + M16 + M17 +
M21 + M22 + M23 + M24 + M25 + M26 + M27 +
M31 + M32 + M33 + M34 + M35 + M36 + M37 +
M41 + M42 + M43 + M44 + M45 + M46 + M47 +
M51 + M52 + M53 + M54 + M55 + M56 + M57 +
M61 + M62 + M63 + M64 + M65 + M66 + M67 - 10Z5 = 0
100S1 + 100S2 + 100S3 + 100S4 + 100S5 + 100S6 + 100S7 - Z6 = 0
X11 + X22 + X33 + X44 + X55 + X66 + X77 = 2
X11 + X12 + X13 + X14 + X15 + X16 + X17 + S1 = 1
X21 + X22 + X23 + X24 + X25 + X26 + X27 + S2 = 1
```

$X31 + X32 + X33 + X34 + X35 + X36 + X37 + S3 = 1$   
 $X41 + X42 + X43 + X44 + X45 + X46 + X47 + S4 = 1$   
 $X51 + X52 + X53 + X54 + X55 + X56 + X57 + S5 = 1$   
 $X61 + X62 + X63 + X64 + X65 + X66 + X67 + S6 = 1$   
 $X71 + X72 + X73 + X74 + X75 + X76 + X77 + S7 = 1$   
 $X11 + X21 + X31 + X41 + X51 + X61 + X71 - 4X11 < = 0$   
 $X12 + X22 + X32 + X42 + X52 + X62 + X72 - 4X22 < = 0$   
 $X13 + X23 + X33 + X43 + X53 + X63 + X73 - 4X33 < = 0$   
 $X14 + X24 + X34 + X44 + X54 + X64 + X74 - 4X44 < = 0$   
 $X15 + X25 + X35 + X45 + X55 + X65 + X75 - 4X55 < = 0$   
 $X16 + X26 + X36 + X46 + X56 + X66 + X76 - 4X66 < = 0$   
 $X17 + X27 + X37 + X47 + X57 + X67 + X77 - 4X77 < = 0$   
 $250X11 + 60X31 + 120X71 + I11 - 900Y11 - M11 = 0$   
 $250X12 + 60X32 + 120X72 + I12 - 900Y12 - M12 = 0$   
 $250X13 + 60X33 + 120X73 + I13 - 900Y13 - M13 = 0$   
 $250X14 + 60X34 + 120X74 + I14 - 900Y14 - M14 = 0$   
 $250X15 + 60X35 + 120X75 + I15 - 900Y15 - M15 = 0$   
 $250X16 + 60X36 + 120X76 + I16 - 900Y16 - M16 = 0$   
 $250X17 + 60X37 + 120X77 + I17 - 900Y17 - M17 = 0$   
 $300X11 + 100X31 + 308X51 + I21 - 900Y21 - M21 = 0$   
 $300X12 + 100X32 + 308X52 + I22 - 900Y22 - M22 = 0$   
 $300X13 + 100X33 + 308X53 + I23 - 900Y23 - M23 = 0$   
 $300X14 + 100X34 + 308X54 + I24 - 900Y24 - M24 = 0$   
 $300X15 + 100X35 + 308X55 + I25 - 900Y25 - M25 = 0$   
 $300X16 + 100X36 + 308X56 + I26 - 900Y26 - M26 = 0$   
 $300X17 + 100X37 + 308X57 + I27 - 900Y27 - M27 = 0$   
 $150X11 + 300X41 + 336X51 + 156X71 + I31 - 900Y31 - M31 = 0$   
 $150X12 + 300X42 + 336X52 + 156X72 + I32 - 900Y32 - M32 = 0$   
 $150X13 + 300X43 + 336X53 + 156X73 + I33 - 900Y33 - M33 = 0$   
 $150X14 + 300X44 + 336X54 + 156X74 + I34 - 900Y34 - M34 = 0$   
 $150X15 + 300X45 + 336X55 + 156X75 + I35 - 900Y35 - M35 = 0$   
 $150X16 + 300X46 + 336X56 + 156X76 + I36 - 900Y36 - M36 = 0$   
 $150X17 + 300X47 + 336X57 + 156X77 + I37 - 900Y37 - M37 = 0$   
 $140X21 + 130X31 + 280X51 + 262X61 + 72X71 + I41 - 900Y41 - M41 = 0$   
 $140X22 + 130X32 + 280X52 + 262X62 + 72X72 + I42 - 900Y42 - M42 = 0$   
 $140X23 + 130X33 + 280X53 + 262X63 + 72X73 + I43 - 900Y43 - M43 = 0$   
 $140X24 + 130X34 + 280X54 + 262X64 + 72X74 + I44 - 900Y44 - M44 = 0$   
 $140X25 + 130X35 + 280X55 + 262X65 + 72X75 + I45 - 900Y45 - M45 = 0$   
 $140X26 + 130X36 + 280X56 + 262X66 + 72X76 + I46 - 900Y46 - M46 = 0$   
 $140X27 + 130X37 + 280X57 + 262X67 + 72X77 + I47 - 900Y47 - M47 = 0$   
 $158X21 + 312X41 + I51 - 900Y51 - M51 = 0$   
 $158X22 + 312X42 + I52 - 900Y52 - M52 = 0$

```

158X23 + 312X43 + I53-900Y53-M53 = 0
158X24 + 312X44 + I54-900Y54-M54 = 0
158X25 + 312X45 + I55-900Y55-M55 = 0
158X26 + 312X46 + I56-900Y56-M56 = 0
158X27 + 312X47 + I57-900Y57-M57 = 0
210X21 + 175X41 + 188X61 + I61-900Y61-M61 = 0
210X22 + 175X42 + 188X62 + I62-900Y62-M62 = 0
210X23 + 175X43 + 188X63 + I63-900Y63-M63 = 0
210X24 + 175X44 + 188X64 + I64-900Y64-M64 = 0
210X25 + 175X45 + 188X65 + I65-900Y65-M65 = 0
210X26 + 175X46 + 188X66 + I66-900Y66-M66 = 0
210X27 + 175X47 + 188X67 + I67-900Y67-M67 = 0
I11-M11 + I12-M12 + I13-M13 + I14-M14 + I15-M15 + I16-M16 + I17-M17-
ID1 = 0
I21-M21 + I22-M22 + I23-M23 + I24-M24 + I25-M25 + I26-M26 + I27-M27-
ID2 = 0
I31-M31 + I32-M32 + I33-M33 + I34-M34 + I35-M35 + I36-M36 + I37-M37-
ID3 = 0
I41-M41 + I42-M42 + I43-M43 + I44-M44 + I45-M45 + I46-M46 + I47-M47-
ID4 = 0
I51-M51 + I52-M52 + I53-M53 + I54-M54 + I55-M55 + I56-M56 + I57-M57-
ID5 = 0
I61-M61 + I62-M62 + I63-M63 + I64-M64 + I65-M65 + I66-M66 + I67-M67-
ID6 = 0
Y11 + Y12 + Y13 + Y14 + Y15 + Y16 + Y17-PUR1 + SEL1 = 0
Y21 + Y22 + Y23 + Y24 + Y25 + Y26 + Y27-PUR2 + SEL2 = 0
Y31 + Y32 + Y33 + Y34 + Y35 + Y36 + Y27-PUR3 + SEL3 = 0
Y41 + Y42 + Y43 + Y44 + Y45 + Y46 + Y27-PUR4 + SEL4 = 0
Y51 + Y52 + Y53 + Y54 + Y55 + Y56 + Y27-PUR5 + SEL5 = 0
Y61 + Y62 + Y63 + Y64 + Y65 + Y66 + Y27-PUR6 + SEL6 = 0
end
int X11      int X12      int X13      int X14      int X15      int X16
int X17      int X21      int X22      int X23      int X24      int X25
int X26      int X27      int X31      int X32      int X33      int X34
int X35      int X36      int X37      int X41      int X42      int X43
int X44      int X45      int X46      int X47      int X51      int X52
int X53      int X54      int X55      int X56      int X57      int X61
int X62      int X63      int X64      int X65      int X66      int X67
int X71      int X72      int X73      int X74      int X75      int X76
int X77      gin Y11      gin Y12      gin Y13      gin Y14      gin Y15
gin Y16      gin Y17      gin Y21      gin Y22      gin Y23      gin Y24
gin Y25      gin Y26      gin Y27      gin Y31      gin Y32      gin Y33

```

gin Y34	gin Y35	gin Y36	gin Y37	gin Y41	gin Y42
gin Y43	gin Y44	gin Y45	gin Y46	gin Y47	gin Y51
gin Y52	gin Y53	gin Y54	gin Y55	gin Y56	
gin Y57	gin Y61	gin Y62	gin Y63	gin Y64	
gin Y65	gin Y66	gin Y67			
leave					

**THIS LINDO INPUT FILE IS CREATED BY USING 2-STAGE SINGLE-PERIOD MODEL FOR A CASE OF 7 PARTS, 6 MACHINES**

```

min Z
st
Z1 + Z2 + Z3 + Z4 + Z5 + Z6 - Z = 0
100CN - Z1 = 0
15Y11
15Y21
15Y31 + 15Y32
15Y41 + 15Y42
15Y52 +
15Y62 - Z2 = 0
10PUR1 + 10SEL1 + 10PUR2 + 10SEL2 + 10PUR3 + 10SEL3 +
10PUR4 + 10SEL4 + 10PUR5 + 10SEL5 + 10PUR6 + 10SEL6 - Z3 = 0
ID11 + ID21 + ID31 + ID41 + ID51 + ID61 - Z41 = 0
Z41 - 40Z4 = 0
M111 +
M211 +
M311 + M321 +
M411 + M421 +
M521 +
M621 - Z51 = 0
Z51 - 10Z5 = 0
S11 + S21 + S31 + S41 + S51 + S61 + S71 - Z61 = 0
100Z61 - Z6 = 0
X111 + S11 = 1
X311 + S31 = 1
X511 + S51 = 1
X711 + S71 = 1
X221 + S21 = 1

```

```

X421 + S41 = 1
X621 + S61 = 1
CN = 2
250X111 + 60X311 + 120X711 + I111-900Y11-M111 = 0
300X111 + 100X311 + 308X511 + I211-900Y21-M211 = 0
150X111 + 336X511 + 156X711 + I311-900Y31-M311 = 0
300X421 + I321-900Y32-M321 = 0
130X311 + 280X511 + 72X711 + I411-900Y41-M411 = 0
140X221 + 262X621 + I421-900Y42-M421 = 0
158X221 + 312X421 + I521-900Y52-M521 = 0
210X221 + 175X421 + 188X621 + I621-900Y62-M621 = 0
-M111 + I111-ID11 = 0
-M211 + I211-ID21 = 0
-M311 + I311-M321 + I321-ID31 = 0
-M411 + I411-M421 + I421-ID41 = 0
-M521 + I521-ID51 = 0
-M621 + I621-ID61 = 0
Y11-PUR1 + SEL1 = 0
Y21-PUR2 + SEL2 = 0
Y31 + Y32-PUR3 + SEL3 = 0
Y41 + Y42-PUR4 + SEL4 = 0
Y51-PUR5 + SEL5 = 0
Y61-PUR6 + SEL6 = 0
end
int X111    int X311    int X511    int X711    int X221    int X421
int X621    gin Y11     gin Y21     gin Y31     gin Y32     gin Y41
gin Y42     gin Y52     gin Y62
leave

```

## APPENDIX V

### EXAMPLE LINDO INPUT FILES FOR MULTI-PERIOD PF/MCC

### MODEL AND TWO-STAGE MULTI-PERIOD PF/MCC MODEL

THIS LINDO INPUT FILE IS CREATED BY USING MULTI-PERIOD PF/MCC MODEL FOR A CASE OF 10 PARTS, 6 MACHINES AND 3 PERIODS

```
min Z
st
Z1 + Z2 + Z3 + Z4 + Z5 + Z6 - Z = 0
100CN - Z1 = 0
15Y11 + 15Y12 + 15Y13 + 15Y14 + 15Y15 + 15Y16 + 15Y17 + 15Y18 +
15Y22 + 15Y22 + 15Y23 + 15Y24 + 15Y25 + 15Y26 + 15Y27 + 15Y28 +
15Y33 + 15Y32 + 15Y33 + 15Y34 + 15Y35 + 15Y36 + 15Y27 + 15Y28 +
15Y44 + 15Y42 + 15Y43 + 15Y44 + 15Y45 + 15Y46 + 15Y27 + 15Y28 +
15Y55 + 15Y52 + 15Y53 + 15Y54 + 15Y55 + 15Y56 + 15Y27 + 15Y28 +
15Y66 + 15Y62 + 15Y63 + 15Y64 + 15Y65 + 15Y66 + 15Y27 + 15Y28 - Z2 = 0
10PUR1 + 10SEL1 + 10PUR2 + 10SEL2 + 10PUR3 + 10SEL3 +
10PUR4 + 10SEL4 + 10PUR5 + 10SEL5 + 10PUR6 + 10SEL6 - Z3 = 0
ID11 + ID21 + ID31 + ID41 + ID51 + ID61 - Z41 = 0
ID12 + ID22 + ID32 + ID42 + ID52 + ID62 - Z42 = 0
ID13 + ID23 + ID33 + ID43 + ID53 + ID63 - Z43 = 0
Z41 + Z42 + Z43 - 20Z4 = 0
M111 + M121 + M131 + M141 +
M211 + M221 + M231 + M241 +
M311 + M321 + M331 + M341 +
M411 + M421 + M431 + M441 +
M511 + M521 + M531 + M541 +
M611 + M621 + M631 + M641 - Z51 = 0
M132 + M142 + M152 + M162 +
```

$M232 + M242 + M252 + M262 +$   
 $M332 + M342 + M352 + M362 +$   
 $M432 + M442 + M452 + M462 +$   
 $M532 + M542 + M552 + M562 +$   
 $M632 + M642 + M652 + M662 - Z52 = 0$   
 $M153 + M163 + M173 + M183 +$   
 $M253 + M263 + M273 + M283 +$   
 $M353 + M363 + M373 + M383 +$   
 $M453 + M463 + M473 + M483 +$   
 $M553 + M563 + M573 + M583 +$   
 $M653 + M663 + M673 + M683 - Z53 = 0$   
 $Z51 + Z52 + Z53 - 10Z5 = 0$   
 $S11 + S21 + S31 + S41 - Z61 = 0$   
 $S32 + S42 + S52 + S62 - Z62 = 0$   
 $S53 + S63 + S73 + S83 - Z63 = 0$   
 $100Z61 - 100Z62 + 100Z63 - Z6 = 0$   
 $X111 + X121 + X131 + X141 + S11 = 1$   
 $X211 + X221 + X231 + X241 + S21 = 1$   
 $X311 + X321 + X331 + X341 + S31 = 1$   
 $X411 + X421 + X431 + X441 + S41 = 1$   
 $X332 + X342 + X352 + X362 + S32 = 1$   
 $X432 + X442 + X452 + X462 + S42 = 1$   
 $X532 + X542 + X552 + X562 + S52 = 1$   
 $X632 + X642 + X652 + X662 + S62 = 1$   
 $X553 + X563 + X573 + X583 + S53 = 1$   
 $X653 + X663 + X673 + X683 + S63 = 1$   
 $X753 + X763 + X773 + X783 + S73 = 1$   
 $X853 + X863 + X873 + X883 + S83 = 1$   
 $CN = 2$   
 $X111 + X211 + X311 + X411 - 2X111 < = 0$   
 $X121 + X221 + X321 + X421 - 2X221 < = 0$   
 $X131 + X231 + X331 + X431 - 2X331 < = 0$   
 $X141 + X241 + X341 + X441 - 2X441 < = 0$   
 $X111 + X221 + X331 + X441 = 2$   
 $X332 + X432 + X532 + X632 - 2X332 < = 0$   
 $X342 + X442 + X542 + X642 - 2X442 < = 0$   
 $X352 + X452 + X552 + X652 - 2X552 < = 0$   
 $X362 + X462 + X562 + X662 - 2X662 < = 0$   
 $X332 + X442 + X552 + X662 = 2$   
 $X553 + X653 + X753 + X853 - 2X553 < = 0$   
 $X563 + X663 + X763 + X863 - 2X663 < = 0$   
 $X573 + X673 + X773 + X873 - 2X773 < = 0$

$X583 + X683 + X783 + X883 - 2X883 < = 0$   
 $X553 + X663 + X773 + X883 = 2$   
 $188X111 + 0X211 + 120X311 + 0X411 + 1111 - 500Y11 - M111 = 0$   
 $188X121 + 0X221 + 120X321 + 0X421 + 1121 - 500Y12 - M121 = 0$   
 $188X131 + 0X231 + 120X331 + 0X431 + 1131 - 500Y13 - M131 = 0$   
 $188X141 + 0X241 + 120X341 + 0X441 + 1141 - 500Y14 - M141 = 0$   
 $225X111 + 0X211 + 200X311 + 0X411 + 1211 - 500Y21 - M211 = 0$   
 $225X121 + 0X221 + 200X321 + 0X421 + 1221 - 500Y22 - M221 = 0$   
 $225X131 + 0X231 + 200X331 + 0X431 + 1231 - 500Y23 - M231 = 0$   
 $225X141 + 0X241 + 200X341 + 0X441 + 1241 - 500Y24 - M241 = 0$   
 $112X111 + 0X211 + 0X311 + 216X411 + 1311 - 500Y31 - M311 = 0$   
 $112X121 + 0X221 + 0X321 + 216X421 + 1321 - 500Y32 - M321 = 0$   
 $112X131 + 0X231 + 0X331 + 216X431 + 1331 - 500Y33 - M331 = 0$   
 $112X141 + 0X241 + 0X341 + 216X441 + 1341 - 500Y34 - M341 = 0$   
 $0X111 + 96X211 + 260X311 + 0X411 + 1411 - 500Y41 - M411 = 0$   
 $0X121 + 96X221 + 260X321 + 0X421 + 1421 - 500Y42 - M421 = 0$   
 $0X131 + 96X231 + 260X331 + 0X431 + 1431 - 500Y43 - M431 = 0$   
 $0X141 + 96X241 + 260X341 + 0X441 + 1441 - 500Y44 - M441 = 0$   
 $0X111 + 108X211 + 0X311 + 225X411 + 1511 - 500Y51 - M511 = 0$   
 $0X121 + 108X221 + 0X321 + 225X421 + 1521 - 500Y52 - M521 = 0$   
 $0X131 + 108X231 + 0X331 + 225X431 + 1531 - 500Y53 - M531 = 0$   
 $0X141 + 108X241 + 0X341 + 225X441 + 1541 - 500Y54 - M541 = 0$   
 $0X111 + 144X211 + 0X311 + 126X411 + 1611 - 500Y61 - M611 = 0$   
 $0X121 + 144X221 + 0X321 + 126X421 + 1621 - 500Y62 - M621 = 0$   
 $0X131 + 144X231 + 0X331 + 126X431 + 1631 - 500Y63 - M631 = 0$   
 $0X141 + 144X241 + 0X341 + 126X441 + 1641 - 500Y64 - M641 = 0$   
 $96X332 + 0X432 + 0X532 + 0X632 + 1132 - 500Y13 - M132 = 0$   
 $96X342 + 0X442 + 0X542 + 0X642 + 1142 - 500Y14 - M142 = 0$   
 $96X352 + 0X452 + 0X552 + 0X652 + 1152 - 500Y15 - M152 = 0$   
 $96X362 + 0X462 + 0X562 + 0X662 + 1162 - 500Y16 - M162 = 0$   
 $160X332 + 0X432 + 330X532 + 0X632 + 1232 - 500Y23 - M232 = 0$   
 $160X342 + 0X442 + 330X542 + 0X642 + 1242 - 500Y24 - M242 = 0$   
 $160X352 + 0X452 + 330X552 + 0X652 + 1252 - 500Y25 - M252 = 0$   
 $160X362 + 0X462 + 330X562 + 0X662 + 1262 - 500Y26 - M262 = 0$   
 $0X332 + 156X432 + 360X532 + 0X632 + 1332 - 500Y33 - M332 = 0$   
 $0X342 + 156X442 + 360X542 + 0X642 + 1342 - 500Y34 - M342 = 0$   
 $0X352 + 156X452 + 360X552 + 0X652 + 1352 - 500Y35 - M352 = 0$   
 $0X362 + 156X462 + 360X562 + 0X662 + 1362 - 500Y36 - M362 = 0$   
 $208X332 + 0X432 + 300X532 + 350X632 + 1432 - 500Y43 - M432 = 0$   
 $208X342 + 0X442 + 300X542 + 350X642 + 1442 - 500Y44 - M442 = 0$   
 $208X352 + 0X452 + 300X552 + 350X652 + 1452 - 500Y45 - M452 = 0$   
 $208X362 + 0X462 + 300X562 + 350X662 + 1462 - 500Y46 - M462 = 0$

OX332 + 162X432 + OX532 + OX632 + I532-500Y53-M532 = 0  
 OX342 + 162X442 + OX542 + OX642 + I542-500Y54-M542 = 0  
 OX352 + 162X452 + OX552 + OX652 + I552-500Y55-M552 = 0  
 OX362 + 162X462 + OX562 + OX662 + I562-500Y56-M562 = 0  
 OX332 + 91X432 + OX532 + 250X632 + I632-500Y63-M632 = 0  
 OX342 + 91X442 + OX542 + 250X642 + I642-500Y64-M642 = 0  
 OX352 + 91X452 + OX552 + 250X652 + I652-500Y65-M652 = 0  
 OX362 + 91X462 + OX562 + 250X662 + I662-500Y66-M662 = 0  
 OX553 + OX653 + 300X753 + OX853 + I153-500Y15-M153 = 0  
 OX563 + OX663 + 300X763 + OX863 + I163-500Y16-M163 = 0  
 OX573 + 300X673 + OX773 + OX873 + I173-500Y17-M173 = 0  
 OX583 + 300X683 + OX783 + OX883 + I183-500Y18-M183 = 0  
 275X553 + OX653 + OX753 + OX853 + I253-500Y25-M253 = 0  
 275X563 + OX663 + OX763 + OX863 + I263-500Y26-M263 = 0  
 OX573 + OX673 + OX773 + OX873 + I273-500Y27-M273 = 0  
 OX583 + OX683 + OX783 + OX883 + I283-500Y28-M283 = 0  
 300X553 + OX653 + 390X753 + 250X853 + I353-500Y35-M353 = 0  
 300X563 + OX663 + 390aw6gw250X863 + I363-500Y36-M363 = 0  
 OX573 + 390X673 + 250X773 + 250X873 + I373-500Y37-M373 = 0  
 OX583 + 390X683 + 250X783 + 250X883 + I383-500Y38-M383 = 0  
 250X553 + 262X653 + 180X753 + OX853 + I453-500Y45-M453 = 0  
 250X563 + 262X663 + 180X763 + OX863 + I463-500Y46-M463 = 0  
 262X573 + 180X673 + OX773 + OX873 + I473-500Y47-M473 = 0  
 262X583 + 180X683 + OX783 + OX883 + I483-500Y48-M483 = 0  
 OX553 + OX653 + OX753 + 312X853 + I553-500Y55-M553 = 0  
 OX563 + OX663 + OX763 + 312X863 + I563-500Y56-M563 = 0  
 OX573 + OX673 + 312X773 + 312X873 + I573-500Y57-M573 = 0  
 OX583 + OX683 + 312X783 + 312X883 + I583-500Y58-M583 = 0  
 OX553 + 188X653 + OX753 + 275X853 + I653-500Y65-M653 = 0  
 OX563 + 188X663 + OX763 + 275X863 + I663-500Y66-M663 = 0  
 188X573 + OX673 + 275X773 + 275X873 + I673-500Y67-M673 = 0  
 188X583 + OX683 + 275X783 + 275X883 + I683-500Y68-M683 = 0  
 + I111-M111 + I121-M121 + I131-M131 + I141-M141-ID11 = 0  
 + I211-M211 + I221-M221 + I231-M231 + I241-M241-ID21 = 0  
 + I311-M311 + I321-M321 + I331-M331 + I341-M341-ID31 = 0  
 + I411-M411 + I421-M421 + I431-M431 + I441-M441-ID41 = 0  
 + I511-M511 + I521-M521 + I531-M531 + I541-M541-ID51 = 0  
 + I611-M611 + I621-M621 + I631-M631 + I641-M641-ID61 = 0  
 + I132-M132 + I142-M142 + I152-M152 + I162-M162-ID12 = 0  
 + I232-M232 + I242-M242 + I252-M252 + I262-M262-ID22 = 0  
 + I332-M332 + I342-M342 + I352-M352 + I362-M362-ID32 = 0  
 + I432-M432 + I442-M442 + I452-M452 + I462-M462-ID42 = 0

```

+I532-M532 +I542-M542 +I552-M552 +I562-M562-ID52 =0
+I632-M632 +I642-M642 +I652-M652 +I662-M662-ID62 =0
+I153-M153 +I163-M163 +I173-M173 +I183-M183-ID13 =0
+I253-M253 +I263-M263 +I273-M273 +I283-M283-ID23 =0
+I353-M353 +I363-M363 +I373-M373 +I383-M383-ID33 =0
+I453-M453 +I463-M463 +I473-M473 +I483-M483-ID43 =0
+I553-M553 +I563-M563 +I573-M573 +I583-M583-ID53 =0
+I653-M653 +I663-M663 +I673-M673 +I683-M683-ID63 =0
Y11 +Y12 +Y13 +Y14 +Y15 +Y16 +Y17 +Y18-PUR1 + SEL1 =0
Y21 +Y22 +Y23 +Y24 +Y25 +Y26 +Y27 +Y28-PUR2 + SEL2 =0
Y31 +Y32 +Y33 +Y34 +Y35 +Y36 +Y27 +Y28-PUR3 + SEL3 =0
Y41 +Y42 +Y43 +Y44 +Y45 +Y46 +Y27 +Y28-PUR4 + SEL4 =0
Y51 +Y52 +Y53 +Y54 +Y55 +Y56 +Y27 +Y28-PUR5 + SEL5 =0
Y61 +Y62 +Y63 +Y64 +Y65 +Y66 +Y27 +Y28-PUR6 + SEL6 =0
end

```

```

int X111    int X121    int X131    int X141    int X211    int X221
int X231    int X241    int X311    int X321    int X331    int X341
int X411    int X421    int X431    int X441    int X332    int X342
int X352    int X362    int X432    int X442    int X452    int X462
int X532    int X542    int X552    int X562    int X632    int X642
int X652    int X662    int X553    int X563    int X573    int X583
int X653    int X663    int X673    int X683    int X753    int X763
int X773    int X783    int X853    int X863    int X873    int X883
gin Y11     gin Y12     gin Y13     gin Y14     gin Y15     gin Y16
gin Y17     gin Y18     gin Y21     gin Y22     gin Y23     gin Y24
gin Y25     gin Y26     gin Y27     gin Y28     gin Y31     gin Y32
gin Y33     gin Y34     gin Y35     gin Y36     gin Y37     gin Y38
gin Y41     gin Y42     gin Y43     gin Y44     gin Y45     gin Y46
gin Y47     gin Y48     gin Y51     gin Y52     gin Y53     gin Y54
gin Y55     gin Y56     gin Y57     gin Y58     gin Y61     gin Y62
gin Y63     gin Y64     gin Y65     gin Y66     gin Y67     gin Y68
leave

```

**THIS LINDO INPUT FILE IS CREATED BY USING 2-STAGE MULTI-PERIOD PF/MCC MODEL FOR A CASE OF 10 PARTS, 6 MACHINES AND 3 PERIODS (SIMILAR CASE)**

```

min Z
st

```

$Z1 + Z2 + Z3 + Z4 + Z5 + Z6 - Z = 0$   
 $100CN - Z1 = 0$   
 $15Y11 +$   
 $15Y21 +$   
 $15Y31 + Y32 +$   
 $15Y41 + Y42 +$   
 $15Y52 +$   
 $15Y62 - Z2 = 0$   
 $10PUR1 + 10SEL1 + 10PUR2 + 10SEL2 + 10PUR3 + 10SEL3 +$   
 $10PUR4 + 10SEL4 + 10PUR5 + 10SEL5 + 10PUR6 + 10SEL6 - Z3 = 0$   
 $ID11 + ID21 + ID31 + ID41 + ID51 + ID61 - Z41 = 0$   
 $ID12 + ID22 + ID32 + ID42 + ID52 + ID62 - Z42 = 0$   
 $ID13 + ID23 + ID33 + ID43 + ID53 + ID63 - Z43 = 0$   
 $Z41 + Z42 + Z43 - 20Z4 = 0$   
 $M111 +$   
 $M211 +$   
 $M311 + M321 +$   
 $M411 + M421 +$   
 $M521 +$   
 $M621 - Z51 = 0$   
 $M112 +$   
 $M212 +$   
 $M312 + M322 +$   
 $M412 + M422 +$   
 $M522 +$   
 $M622 - Z52 = 0$   
 $M113 +$   
 $M213 +$   
 $M313 + M323 +$   
 $M413 + M423 +$   
 $M523 +$   
 $M623 - Z53 = 0$   
 $Z51 + Z52 + Z53 - 10Z5 = 0$   
 $S11 + S21 + S31 + S41 - Z61 = 0$   
 $S32 + S42 + S52 + S62 - Z62 = 0$   
 $S53 + S63 + S73 + S83 - Z63 = 0$   
 $100Z61 + 100Z62 + 100Z63 - Z6 = 0$   
 $X111 + S11 = 1$   
 $X311 + S31 = 1$   
 $X221 + S21 = 1$   
 $X421 + S41 = 1$   
 $X312 + S32 = 1$

$X512 + S52 = 1$   
 $X422 + S42 = 1$   
 $X622 + S62 = 1$   
 $X513 + S53 = 1$   
 $X713 + S73 = 1$   
 $X623 + S63 = 1$   
 $X823 + S83 = 1$   
 $CN = 2$   
 $188X111 + 120X311 + I111 - 500Y11 - M111 = 0$   
 $225X111 + 200X311 + I211 - 500Y21 - M211 = 0$   
 $112X111 + I311 - 500Y31 - M311 = 0$   
 $216X421 + I321 - 500Y32 - M321 = 0$   
 $260X311 + I411 - 500Y41 - M411 = 0$   
 $96X221 + I421 - 500Y42 - M421 = 0$   
 $108X221 + 225X421 + I521 - 500Y52 - M521 = 0$   
 $144X221 + 126X421 + I621 - 500Y62 - M621 = 0$   
 $96X312 + I112 - 500Y11 - M112 = 0$   
 $160X312 + 330X512 + I212 - 500Y21 - M212 = 0$   
 $360X512 + I312 - 500Y31 - M312 = 0$   
 $156X422 + I322 - 500Y32 - M322 = 0$   
 $208X312 + 300X512 + I412 - 500Y41 - M412 = 0$   
 $350X622 + I422 - 500Y42 - M422 = 0$   
 $162X422 + I522 - 500Y52 - M522 = 0$   
 $91X422 + 250X622 + I622 - 500Y62 - M622 = 0$   
 $300X713 + I113 - 500Y11 - M113 = 0$   
 $275X513 + I213 - 500Y21 - M213 = 0$   
 $300X513 + 390X713 + I313 - 500Y31 - M313 = 0$   
 $250X823 + I323 - 500Y32 - M323 = 0$   
 $250X513 + 180X713 + I413 - 500Y41 - M413 = 0$   
 $262X623 + I423 - 500Y42 - M423 = 0$   
 $312X823 + I523 - 500Y52 - M523 = 0$   
 $188X623 + 275X823 + I623 - 500Y62 - M623 = 0$   
 $-M111 + I111 - ID11 = 0$   
 $-M211 + I211 - ID21 = 0$   
 $-M311 + I311 - M321 + I321 - ID31 = 0$   
 $-M411 + I411 - M421 + I421 - ID41 = 0$   
 $-M521 + I521 - ID51 = 0$   
 $-M621 + I621 - ID61 = 0$   
 $-M112 + I112 - ID12 = 0$   
 $-M212 + I212 - ID22 = 0$   
 $-M312 + I312 - M322 + I322 - ID32 = 0$   
 $-M412 + I412 - M422 + I422 - ID42 = 0$

```

-M522 + I522-ID52 = 0
-M622 + I622-ID62 = 0
-M113 + I113-ID13 = 0
-M213 + I213-ID23 = 0
-M313 + I313-M323 + I323-ID33 = 0
-M413 + I413-M423 + I423-ID43 = 0
-M523 + I523-ID53 = 0
-M623 + I623-ID63 = 0
Y11-PUR1 + SEL1 = 0
Y21-PUR2 + SEL2 = 0
Y31 + Y32-PUR3 + SEL3 = 0
Y41 + Y42-PUR4 + SEL4 = 0
Y52-PUR5 + SEL5 = 0
Y62-PUR6 + SEL6 = 0
end
int X111    int X311    int X221    int X421    int X312    int X512
int X422    int X622    int X513    int X713    int X623    int X823
gin Y11     gin Y21     gin Y31     gin Y32     gin Y41     gin Y42
gin Y52     gin Y62
leave

```