View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Texas A&M Repository

DESIGN OF A CLUSTER ANALYSIS HEURISTIC FOR THE CONFIGURATION AND CAPACITY MANAGEMENT OF MANUFACTURING CELLS

A Dissertation

by

YOUNG HAK SHIM

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

May 2006

Major Subject: Industrial Engineering

DESIGN OF A CLUSTER ANALYSIS HEURISTIC FOR THE CONFIGURATION AND CAPACITY MANAGEMENT OF MANUFACTURING CELLS

A Dissertation

by

YOUNG HAK SHIM

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Chair of Committee,	Cesar O. Malave
Committee Members,	Amarnath P. Banerjee
	Lewis Ntaimo
	Sheng-Jen "Tony" Hsieh
Head of Department,	Brett A. Peters

May 2006

Major Subject: Industrial Engineering

ABSTRACT

Design of a Cluster Analysis Heuristic for the Configuration and

Capacity Management of Manufacturing Cells. (May 2006)

Young Hak Shim, B.S., Kyunghee University;

M.S., Texas A&M University

Chair of Advisory Committee: Dr. Cesar O. Malave

This dissertation presents the configuration and capacity management of manufacturing cells using cluster analysis. A heuristic based on cluster analysis is developed to solve cell formation in cellular manufacturing systems (CMS). The clustering heuristic is applied for cell formation considering processing requirement (CFOPR) as well as various manufacturing factors (CFVMF).

The proposed clustering heuristic is developed by employing a new solving structure incorporating hierarchical and non-hierarchical clustering methods. A new similarity measure is constructed by modifying the Jarccard similarity and a new assignment algorithm is proposed by employing the new pairwise exchange method.

In CFOPR, the clustering heuristic is modified by adding a feedback step and more exact allocation rules. Grouping efficacy is employed as a measure to evaluate solutions obtained from the heuristic. The clustering heuristic for CFOPR was evaluated on 23 test problems taken from the literature in order to compare with other approaches and produced the best solution in 18 out of 23 and the second best in the remaining

problems. These solutions were obtained in a considerably short time and even the largest test problem was solved in around one and a half seconds.

In CFVMF, the machine capacity was first ensured, and then manufacturing cells were configured to minimize intercellular movements. In order to ensure the machine capacity, the duplication of machines and the split of operations are allowed and operations are assigned into duplicated machines by the largest-first rule. The clustering heuristic for CFVMF proposes a new similarity measure incorporating processing requirement, material flow and machine workload and a new machine-part matrix representing material flow and processing time assigned to multiple identical machines. Also, setup time, which has not been clearly addressed in existing research, is discussed in the solving procedure.

The clustering heuristic for CFVMF employs two evaluation measures such as the number of intercellular movements and grouping efficacy. In two test problems taken from the literature, the heuristic for CFVMF produced the same results, but the trade-off problem between the two evaluation measures is proposed to consider the goodness of grouping.

DEDICATION

To my parents and my family for everything they gave me.

ACKNOWLEDGMENTS

I would like to express my appreciation to Dr. Cesar O. Malave, my advisor and mentor, who has encouraged me and guided and advised this research to the right destination during my graduate study. I would also like to express my gratitude to my advisory committee: Dr. Amarnath P. Banerjee, Dr. Sheng-Jen "Tony" Hsieh and Dr. Lewis Ntaimo who gave me great advice and a great influence.

I give special thanks to my parents, Euiyong Shim and Sukja Seo, who have been the spiritual foundation of my life and to my parents-in-law, Wonsup Yoon and Byungsook Lee, for their encouragement and support. I am especially grateful to my lovely wife, Mihyun Yoon, who supports me with her love, sacrifice and patience and my precious son, Wongi. Thanks also to my sisters and sister-in-law.

TABLE OF CONTENTS

Page

CHAPTER Page

CHAPTER Page

CHAPTER Page

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

TABLE Page

CHAPTER I

INTRODUCTION

1.1 Motivation

At present, the flexibility of the manufacturing environment and the speed of adaptation for customer requirements have become a vital aspect of manufacturing systems. These recent manufacturing environments have made the change from mass production of a single product to the manufacturing of various products in medium size batches. In order to satisfy this new situation, the flexibility of the production process and the reduction of processing time are required in manufacturing systems.

Flexibility manufacturing systems (FMS) and group technology (GT) cope with the current manufacturing environment. GT is recognized as a manufacturing philosophy to improve manufacturing flexibility and production efficiency by grouping parts and machines into families and cells based on similar or dissimilar characteristics. Cellular manufacturing is an application of GT that applies the mass production effect to various products and medium-sized production in batch manufacturing systems. As reasons for the establishment of a cellular manufacturing system (CMS), Wemmerlov and Hyer (1989) and Wemmerlov and Johnson (1997) mention that CMS offers many benefits:

- reduced throughput time
- reduced setup time

This dissertation follows the style and the format of the *International Journal of Production Research.*

- reduced work-in-process inventory
- improved part/product quality
- shortened response time to customer orders
- reduced material handling distances/times

The implementation of CMS consists of three phases: (1) cell formation, (2) cell design, and (3) cell management. Cell formation implements machine-part grouping, cell design decides machine layout and cell management deals with scheduling of groups. Among these CMS phases, one of the important problems faced in the implementation of CMS is cell formation. Selim *et al*. (1998) mentioned the three fundamental tasks of cell formation: (1) grouping parts into families, (2) grouping machines into cells, and (3) assigning of families and cells into groups. In other words, cell formation is a method of finding the groups formed by parts and the machines needed for processing the parts. The primary objective of cell formation is to identify part families and machine cells; that is, the parts with similar characteristics are processed within the same family and machines are grouped into cells so as to minimize intercellular flows of parts traveling between machine cells (Heragu 1994).

Due to the change in manufacturing environment, cell formation in CMS has received a great deal of attention from researchers, and many approaches have been developed to solve the cell formation problem. While the cell formation problem has received much attention, most of it is related to the binary version-based cell formation considering only production requirement (CFOPR) having the entry of 1 or 0. A ‘1’

entry indicates a requirement for a part to be processed on a machine and a '0' entry indicates that a part is not required to be processed. However, the cell formation problem is known to be NP-complete (King and Nakornchai 1982), which means that there are no known algorithms for optimally solving the cell formation problems in polynomial time. Thus, most approaches employ methods based on a heuristic in order to obtain near optimal solutions in reasonable time. However, the trade-off between the accuracy and size of solutions and the computational complexity has been an interesting topic to many researchers in cell formation in CMS.

In order to solve CFOPR, many researchers proposed a variety of approaches. Recently, meta-heuristics such as genetic algorithm (Cheng *et al*. 1998, Onwubolu and Mutingi 2001, and Goncalves and Resende 2004), simulated annealing algorithm (Boctor 1991 and Adil *et al.* 1997), and evolutionary programming (Dimopoulos and Mort 2001 and 2004) are utilized to solve large-scale problems. They reported that these methods produced better results in comparison with other approaches. However, these methods usually have the disadvantage of high computational complexity, because the statistical nature in a heuristic makes algorithms conduct many runs. In order to overcome this disadvantage and achieve good quality solutions, the development of a new solving methodology is required. In this dissertation, the new clustering heuristic, which provides good solutions in a short time by adopting the concept of rough-cut in finding an initial solution, is proposed to solve the cell formation problems.

CFOPR can be thought of as a basic but important problem in the configuration of manufacturing systems, because this problem deals with the relationship of processing

requirement between machines and parts. In CFOPR, only the identification of machine cells and part families is the main concern to researchers. However, it may not be in practice. Wu and Salvendy (1993) mentioned that solving methods based on binary information are not suitable to represent a variety of practical manufacturing factors that should be considered in real manufacturing systems. Hence, these manufacturing factors, which include processing time, production volume, operation sequence, split job by lot size, multiple identical machines, setup time and machine available time, should be involved in the design step of CMS. However, because of the high complexity of cell formation considering a variety of manufacturing factors (CFVMF), it is very difficult to solve the cell formation problems considering these manufacturing factors. A few researchers have proposed methods considering a part of the various manufacturing factors in the configuration of CMS, but not all of them.

However, in the limit of my knowledge, researchers have not considered setup time as an independent factor until now. Researchers suppose that setup time is a manufacturing factor included in processing time, or all the setup times are equal. But these assumptions are not practical in industry. As processing times required to process parts are different, setup time required to prepare for processing the parts also varies in practice. Since the main purpose facilitating CMS is to reduce material handling time/cost and setup time/cost (Wemmerlov and Hyer 1989), setup time must be one of the more important factors in the configuration of CMS. As a result, if setup time can be reduced, a reduction in throughput time and the response time to customer orders can be attained. Thus, the setup time factor should be involved when CFVMF is dealt with.

1.2 Problem Description

This research describes the procedures to configure CMS and to manage machine capacity issues based on cluster analysis. The primary objective is to solve the cell formation problems concerned with processing requirement between machines and parts as well as a variety of manufacturing factors in CMS and construct a heuristic based on cluster analysis in order to deal with the cell formation problems.

Cluster analysis is a statistical method recognizing any natural structure from an input data set. In other words, cluster analysis is a data segmentation method assigning a set of objects into subsets, so-called clusters, so that objects within the same clusters have higher resemblance than objects between different clusters (Hastie *et al*. 2001). Usually, cluster analysis does not have any assumption related to the number of clusters or the cluster structure (Johnson and Wichern 1998). Cluster analysis can be applied to cell formation with little modification.

Cell formation is an approach to group parts and machines by identifying similarities or dissimilarities between two parts or two machines. In other words, after calculating similarity (or dissimilarity) coefficients for all pairs of parts or machines, the closest pairs of parts or machines are grouped into the same cell, that is, pairs of parts or machines with the largest similarity coefficient are allocated to the same part families or machine cells, respectively. As mentioned, this research deals with two kinds of the cell formation problems. One is for binary information, and another is for various manufacturing factors. Burbidge (1991) mentioned that only the relationship of processing requirement between machines and parts in the part routings is useful to

identify machine cells and part families. The purpose of cell formation based on a binary data matrix is that all the entries of '1' collect in the diagonal blocks and all the entries of '0' collect in the off-diagonal blocks. However, this case is very unusual in practice. The operation, called an exceptional element, generating an intercellular movement requires extra time/cost. Then, CFOPR is concerned with the minimization of exceptional elements traveling between clusters, that is, the minimization of intercellular movements.

In this dissertation, various manufacturing factors are considered as follows: unit processing time, production volume, operation sequence, machine available time, lot size and setup time. The objective of CFVMF is to minimize intercellular movements traveling between machine-part clusters satisfying the capacity requirement. In order to achieve the objective, it is necessary to satisfy the following issues. First, the capacity of the machine to process operations required by all the parts should be ensured. Second, a new similarity coefficient incorporating various manufacturing factors should be constructed. Third, setup time should be considered in the procedure of algorithm. Fourth, operations should be assigned to multiple identical machines by proper rules. Finally, the split of operations and the duplication of machines are considered to meet the machine capacity.

1.3 Expected Procedures and Contributions

In the procedure for solving the cell formation problems mentioned in this dissertation, the first step that we have to do is to construct a new efficient clustering heuristic based on cluster analysis. Then, the proposed clustering heuristic is modified to

solve CFOPR. A new similarity measure adequate to the binary version is introduced. In particular, a feedback procedure, which is an opposite concept to part allocation, is added to improve the quality of solutions obtained from the proposed clustering heuristic. In order to deal with CFVMF in the configuration of CMS, five issues already mentioned are solved step by step. After ensuring machine capacity, the proposed clustering algorithm is also used to obtain initial seeds using a new similarity measure incorporating similarity coefficients based on processing requirement, material flow and total processing time. Then the allocation of parts into machine cells is conducted. In particular, a setup time issue, which is barely considered in literature, makes the cell formation problem more realistic. The significant contributions of this research are summarized as follows:

- The new heuristic based on cluster analysis is proposed to minimize intercellular movements.
- The proposed clustering heuristic is applied to two kinds of fundamental problems for cell formation in CMS.
- New similarity measures considering processing requirement as well as manufacturing factors are introduced.
- Setup time, which isn't considered thoroughly in the existing research, is involved in the procedure of a heuristic.
- A new machine-part matrix representing multiple identical machines and material flow is constructed.

- Actual material flow of parts on duplicated machines is calculated.

1.4 Outline of the Dissertation

To briefly recap, CHAPTER II presents a literature review related to cluster analysis as well as two kinds of the cell formation problems. The overview of the given topics is provided, and the existing approaches are classified and compared. In CHAPTER III, a heuristic based on cluster analysis is developed and presented. The clustering heuristic including a new similarity measure and a new assignment algorithm is proposed. Each step in the proposed assignment algorithm is described in detail. Cell formation considering only the relationship of processing requirement between parts and machines is described, and the solving procedure based on the clustering heuristic is proposed in CHAPTER IV. The backgrounds of cell formation in CMS are also given. CHAPTER V provides the description of machine capacity management and introduces a new similarity measure incorporating a variety of manufacturing factors. The solving procedure, which includes the proposed clustering heuristic and consists of four sub-algorithms for solving CFVMF, is presented and illustrated. Finally, the summary of this research and future works are discussed in CHAPTER VI.

CHAPTER II

LITERATURE REVIEW

In this chapter, a brief literature review is presented on the following topics: (1) cluster analysis, (2) cell formation considering only processing requirement and (3) cell formation considering various manufacturing factors. In each section, existing solving approaches are classified by a classification category taken from the literature. Finally, a comparison of the existing studies is also provided.

2.1 Cluster Analysis

The objective of cluster analysis is to partition the complex data set into subsets, called clusters. Hence, cluster analysis can be considered as a kind of a sorting problem that allocates *m* objects into *n* clusters. The number of candidate ways of allocating in this manner is given by $s_m(n)$, known as a Stirling number of the second kind as follows:

$$s_m(n) = \frac{\sum_{k=0}^{n} (-1)^{n-k} \binom{n}{k} k^m}{n!}$$

As the number of objects and clusters increases, the number of possible clustering patterns increases exponentially. Due to the high computational complexity of the clustering problem, most clustering algorithms based on cluster analysis are heuristic in nature. Recently, many researchers have attempted to classify various algorithms based on cluster analysis (Hansen and Jaumard 1997, Halkidi *et al*. 2001, Xu and Wunsch II 2005 and Lingras and Huang 2005). Based on these works, clustering

algorithms can widely be classified as follows: (1) hierarchical clustering, (2) partitional clustering and (3) combinatorial search clustering. Even though cluster analysis is conducted by various methods, it follows the main procedure of cluster analysis as follows (Nair and Narendran 1999):

i) Defining a proximity measure: A proximity measure representing the degree of resemblance between two objects is defined.

ii) Choosing a basis for clustering: Clustering criterion is chosen.

iii) Identification of seeds: Points forming clusters are identified.

The clustering heuristic proposed in this research is based on cluster analysis which finds distinct subgroups from a collection of data sets. Cluster analysis is usually based on a proximity matrix that represents similarities or dissimilarities between pairs of objects. The proximity matrix is used as input data of the clustering algorithm. Cluster analysis is similar to classification. However, classification is different from cluster analysis in that it does not require prior constraints. Kusiak and Chow (1988) mentioned two basic methods, such as cluster analysis and classification, for solving the GT problem. Classification is based on predefined design features to group parts, but cluster analysis finds the 'natural' hierarchical structure that the given data set contains. In this research, cluster analysis is employed to solve the cell formation problem in CMS.

2.1.1 Hierarchical Clustering

A hierarchical clustering method utilizes a hierarchical structure built by the resemblance relationship for all pairs of objects. Through a sequence of partitions, the

two closest clusters are merged to form one large cluster according to the proximity matrix calculated at each step. Hierarchical clustering can be classified into agglomerative and divisive methods (Selim *et al*. 1998). Agglomerative clustering methods make the number of clusters decrease from a singleton to a cluster including all objects. However, divisive clustering methods make the number of clusters increase. Linkage clustering methods are most popular among many hierarchical clustering methods. McAuley (1972) introduced the first hierarchical method called the single linkage clustering (SLC) algorithm, which clusters objects with the nearest distance.

2.1.2 Partitional Clustering

A partitional clustering method attempts to directly obtain sub-divided clusters without a hierarchical structure. Usually, the number of clusters is predefined and the structure for finding a solution adopts an iterative procedure. The *K-Means* algorithm (MacQueen 1967) is the best known partitional clustering method. The *K-Means* clustering method first decides k clusters arbitrarily and then assigns objects to the cluster with the nearest mean.

2.1.3 Combinatorial Search Clustering

Due to the high computational complexity of clustering problems, various search techniques are used to solve combinatorial optimization problems. Local search algorithms as well as meta-heuristics such as simulated annealing, tabu search, and genetic algorithm are employed to find a good solution, but may not ensure global

optimum. A combinatorial search clustering method may take huge computational time to obtain very good solutions for large-scale problems.

2.1.4 Application

Kusiak and Chow (1988) mentioned that cluster analysis has been applied in various areas and listed the application areas. Among many applicable areas of cluster analysis, this research focuses on the configuration of manufacturing cells in CMS. The cluster analysis in CMS is concerned with the assignment of machines and parts to machine cells and part families, respectively.

Cluster analysis can be applied in many areas. For example, in biology, biologists collect a set of gene expressions from DNA microarray experiments, and then the collected samples are clustered using a clustering algorithm. In detail, Hastie *et al*. (2001) provided an example with 6830 genes and 64 samples that represent 64 cancer tumors from different patients. The biologists attempted to obtain any organization between genes and samples from the given data set. The procedure to employ cluster analysis in this example is summarized as follows:

i) The given data is a 6830 × 64 matrix representing a measurement for genes (row) and samples (column).

ii) Since 64 samples should be clustered based on 6830 gene expressions, an initial input data is a 64 × 64 matrix obtained by a similarity measure.

iii) Cluster analysis is implemented to find clusters corresponding to the desired query.

Another example is image compression for cluster analysis. Image compression involves breaking the image into small blocks in order to save on the amount of storage required. An image is broken down into very small pieces, and then by the given proximity criterion, each small piece is clustered into its closest cluster centroid. As a result, an approximated image that can save on storage space can be obtained, but with a loss in quality.

2.2 Cell Formation Considering Only Processing Requirement

Approaches for solving CFOPR can be classified as array-based clustering, hierarchical clustering, non-hierarchical clustering, mathematical formulation and heuristics. The first three methods are regarded as procedures based on cluster analysis (Selim *et al*. 1998). Burbidge (1963 and 1971) first introduced the production flow analysis (PFA) based on group analysis. Group analysis divides all the parts into families using similar production operations and then machines into groups which can handle all the operations of parts in the corresponding family by rearranging columns and rows intuitively.

2.2.1 Array-Based Clustering

Array-based clustering methods, which are suitable for small problems, deal with the rearrangement of columns and rows in a machine-part incidence matrix (MPIM) in order to form non-zeroes into diagonal blocks. These methods were used in earlier studies on this topic. McCormick *et al*. (1972) developed the bond-energy algorithm (BEA), which is the first array-based clustering method that maximizes the sum of

column and row bond energies which are created when the values of adjoining element pairs are equal to one. Rank order clustering (ROC) algorithm (King 1980) is a method to group machines and parts by organizing columns and rows in the order of decreasing binary weights. ROC2, which is proposed by King and Nakornchai (1982), is an enhanced ROC which locates rows or columns with entry '1' to the head of the matrix. In addition, the cell formation problems were reviewed and classified. Chandrasekharan and Rajagopalan (1986a) proposed a modified ROC, MODROC, which inserts the stage of hierarchical clustering for cells resulting from ROC.

2.2.2 Hierarchical Clustering

Hierarchical clustering methods group parts and machines into families and cells by similar or dissimilar characteristics. These methods usually consist of three basic steps: first, the generation of similarity or dissimilarity coefficients and second, the construction of the dendogram indicating machine cells or part families at different resemblance levels. Finally, clusters of machines or parts are decided by the dendogram. In hierarchical clustering methods, a similarity coefficient plays an important role for grouping parts and machines because the parts and machines with similar characteristics are processed within the same family and cell.

McAuley (1972) introduced the first hierarchical method called SLC algorithm, which is based on the Jaccard similarity coefficient, to form groups with the highest similarity in the manufacturing systems. Carrie (1973) employed the numerical taxonomy to classify the objects numerically expressed. The average linkage clustering (ALC) algorithm was adopted by Seifoddini and Wolfe (1986) for solving the cell

formation problems. The average similarity coefficient is defined as the average of Jaccard's coefficients for all the machines or parts within two clusters. Seifoddini (1988 and 1989) suggested the use of the ALC algorithm in order to overcome the chaining problem that means to group dissimilar parts into the same cell. They reported that ALC has a better performance than SLC in the aspect of intercellular moves. Wei and Kern (1989) proposed a new similarity coefficient called the commonality score that is defined by common machines that the two parts visit. Khan, Islam and Sarker (2000) compared the existing similarity coefficients and developed a heuristic using the new similarity coefficient. Yasuda and Yin (2001) proposed the dissimilarity coefficient, called an average voids value, indicating the average number of new voids within a new machine group.

2.2.3 Non-hierarchical Clustering

Non-hierarchical clustering methods usually use similarity or dissimilarity information to obtain initial seeds and adopt an iterative procedure. In other words, non-hierarchical clustering methods begin with initial seeds and then repeat the procedure of seeding and clustering to obtain better machine cells or part families. Thus, the solution obtained from this method is affected by initial seeds.

Chandrasekharan and Rajagopalan (1986b) introduced the first non-hierarchical procedure called the ideal seed algorithm. This procedure includes generating ideal seeds using distance measure and clustering machines and parts from ideal seeds. Grouping efficiency was introduced as a measure of the goodness of block diagolization. ZODIAC (Chandrasekharan and Rajagopalan 1987) is the improved version of the ideal seed

algorithm that includes a new generating method of ideal seeds. In this algorithm, ideal seeds are generated by the manipulation of artificial and natural seeds. Srinivasan and Narendran (1991) developed GRAFICS, which is an iterative method based on an assignment clustering model. In that study the importance of initial seeds was mentioned. Srinivasan (1994) improved GRAFICS by considering a minimum spanning tree algorithm for the identification of initial seeds for machines.

2.2.4 Mathematical Formulation

Mathematical formulation is utilized to solve cell formation problems dealing with various objectives and manufacturing factors. Kusiak (1987) introduced the *p*-median modeling with the limitation of the number of machine cells and part families and the generalized IP formulation based on process plans for each part. Shtub (1989) modeled the cell formation problem as the generalized assignment problem (GAP) considering the minimization of allocation costs as the objective function. Srinivasan, Narendran and Mahadevan (1990) developed an assignment clustering model that first solves an assignment problem with maximization objective and then assigns parts into machine cells by the maximum density rule. Boctor (1991) formulated the zero-one linear programming. Adil *et al*. (1997) proposed an assignment allocation algorithm to solve a nonlinear mathematical programming with the objective of the minimization of the weighted sum of voids and exceptional elements. Wang and Roze (1997) modified the *p*-median model by introducing an upper bound for the number of machines or parts per cluster. Chen and Heragu (1999) proposed two stepwise decomposition approaches for the large-scale cell formation problems. Won and Lee (2004) proposed the extended

p-median model to easily deal with large-sized problems by reducing the number of decision variables.

2.2.5 Heuristic

Because it is difficult to solve the large-scale cell formation problems in a reasonable amount of time, heuristics have recently been regarded as new tools to assign machines and parts into clusters with similar characteristics and associated part families. These methods include meta-heuristic, combined methods of meta-heuristic and other clustering method and other statistical methods. Boctor (1991) used a simulated annealing approach to deal with large-scale problems. Ng (1996) employed a minimum spanning tree algorithm to form machine and part clusters using grouping efficiency as the objective function. Adil *et al*. (1997) utilized a simulated annealing algorithm to solve the proposed assignment allocation algorithm. Cheng *et al*. (1998) employed a genetic algorithm, GA-TSP, to solve the cell formation problem formulated as a traveling salesman problem. Dimopoulos and Mort (2001 and 2004) proposed a clustering method (GP-SLCA) based on genetic programming known as an evolutionary method and SLC. In GP-SLCA, various similarity coefficients are generated by modifying the Jaccard coefficient through genetic programming, and then these coefficients are input into SLC in order to obtain solutions. Onwubolu and Mutingi (2001) proposed a genetic algorithm considering the minimization of intercellular movements and cell load-variation. A data-mining technique based on the association rules was applied by Chen (2003). Goncalves and Resende (2004) considered an approximation of the grouping efficacy in order to assign machines/parts into machine

cells/part families at iteration. A genetic algorithm is employed to find the initial machine cells, and the local search is applied to obtain final clusters in the second part. Results from these methods are comparatively better than other methods. However, these methods usually have a disadvantage of high computational complexity, since the statistical nature in a heuristic makes algorithms implement numerous runs.

2.2.6 Other Related Works

Since Burbidge (1963 and 1971) first introduced PFA, many researchers reviewed the cell formation problems with a variety of viewpoints. King and Nakornchai (1982) classified cell formation methods and showed that a cell formation problem is NP-complete. Chandrasekharan and Rajagopalan (1989) tested 7 data sets to evaluate the suitability of the binary data for GT applications. The standard deviation of Jaccard similarities is considered as the factor to control groupability. Wemmerlov and Hyer (1989) and Wemmerlov and Johnson (1997) surveyed users related to CMS in industry and reported benefits such as reduction in throughput time, WIP inventory, material handling and setup time from survey. Kumar and Chandrasekharan (1990) introduced grouping efficacy as a new measure to represent the degree of the goodness of grouping. Miltenburg and Zhang (1991) compared the existing 9 clustering methods and noted that non-hierarchical clustering methods are better than array-based clustering and hierarchical clustering methods. Heragu (1994) reviewed papers on GT and CMS and stated some design factors. Sarker (1996) presented an overview on similarity and dissimilarity coefficients and compared and evaluated various similarity coefficients. Moier *et al*. (1997) reviewed similarity metrics in terms of structural form. Selim *et al*.

(1998) proposed a methodology-based classification on prior research and provided a critical evaluation. Sarker and Mondal (1999) presented a review for a variety of performance measures in cell formation. Sarker (2001) introduced a new grouping measure to evaluate the goodness of grouping in cell formation.

2.3 Cell Formation Considering Various Manufacturing Factors

2.3.1 Operation Sequence and Production Volume

The existing MPIM is constructed by considering only processing requirements between machines and parts. Therefore, this type of matrix is not adequate to represent an actual intercellular flow. Harhalakis *et al*. (1990) argued that the binary MPIM does not represent operation sequence and intercellular movement. They developed a component-machine incidence matrix represented by operation sequence instead of binary information, but this type of matrix cannot deal with production volume. Wu (1998) developed a machine-machine relation matrix to overcome the drawback of the binary MPIM. Xambre and Vilarinho (2003) provided the interoperation flow matrix to consider the flow volume between operations. However, both the machine-machine relation matrix and the interoperation flow matrix do not represent actual intercellular flow of parts traveling between machine clusters.

Venugopal and Narendran (1992) mentioned that cell formation methodologies using binary information are based on the assumption of the same production volume for parts. However, this assumption may make the configuration of manufacturing cells impractical. Vakharia and Wemmerlov (1990) defined operation sequence as an ordering of the machines on which the part is sequentially processed. As a result of this research a

similarity coefficient considering operation sequence was developed. Logendran (1991) considered operation sequence in evaluating the intercell and intracell movements. Wu and Salvendy (1993) proposed the cell formation method modeled by an undirected graph considering only operation sequence. Nair and Narendran (1998) proposed the non-hierarchical clustering approach with a new similarity measure involving operation sequence, but they did not consider production volume. Sarker and Xu (1998) surveyed cell formation methods and similarity measures based on the operation sequence.

Won and Lee (2001) emphasized that cell formation in CMS should incorporate operation sequence and production volume in order to calculate intercellular part flow. They proposed a new production data-based part machine incidence matrix that can represent actual intercellular flow considering operation sequence and production volume, but they did not consider the machine capacity issue. Gupta and Seifoddini (1990) developed the similarity coefficient incorporating production volume, operation sequence and unit operation time. Wu (1998) and Wu and Salvendy (1999) proposed the cell formation method based on the network model taking operation sequence and production volume into consideration.

2.3.2 Machine Capacity and Duplication

Heragu (1994) and Heragu and Gupta (1994) emphasized that machine capacity is the most important factor, hence the number of each machine type, which can satisfy the capacity of machines required to process operations of all the parts, should be considered first. Wu and Salvendy (1999) mentioned that the duplication of any machine type is allowed to ensure the corresponding machine capacity in real manufacturing

systems. Xambre and Vilarinho (2003) mentioned that operation splitting is practical in real industry in the following cases: (1) Identical machines are used near the given capacity, and (2) large batches of similar parts are processed in the same cell. King (1980) and Vakharia and Wemmerlov (1990) considered the duplication of machines to remove exceptional elements. Heragu and Gupta (1994) proposed a heuristic considering the capacity and the duplication of machines. They first determined the number of machine types and then modified machine-part clusters by duplicating required machine types to ensure the capacity constraints. Wu and Salvendy (1999) proposed a merging and breaking heuristic based on a network model considering multiple identical machines to ensure the capacity requirement. They assigned multiple identical machines into different clusters. Nair and Narendran (1999) developed a bicriteria model for minimizing intercellular moves and within-cell load variation using the similarity coefficient based on ordinal data and ratio-level data, but did not consider the duplication of machine types. Xambre and Vilarinho (2003) proposed the cell formation method of a new viewpoint for meeting the capacity requirement through the allocation of operations to machines and mentioned that operation splitting in manufacturing systems where multiple identical machines are used can occur to satisfy the capacity requirement.

2.4 Comparison with Existing Studies

2.4.1 Complexity

As mentioned, the complexity of the cell formation problem is known to be NP-complete (King and Nakornchai 1982). Thus, the solving methodology is based on a

heuristic in nature. Many approaches based on a heuristic have been proposed in literature. Recently, meta-heuristics have been introduced as new tools for solving the cell formation problem and produced better solutions in comparison with other methods. However, meta-heuristics have high computational complexity (Chen 2003). Thus, these methods may spend a prohibitive amount of time to obtain solutions for large-sized problems. Adil *et al*. (1997) employed simulated annealing to deal with the large-scale cell formation problems. In addition, it was mentioned that the simulated annealing algorithm requires considerable computing time. Although other methods, except for meta-heuristic, have a lower computational complexity for solving the cell formation problem, the quality of solutions obtained from methods proposed in the literature is not good. Due to these reasons, a new solving method is required to obtain a good solution in a short amount of time.

Many researchers have used clustering methods for solving the cell formation problems because of their simple procedure. Hierarchical and non-hierarchical clustering methods first construct a similarity coefficient matrix, and then the cell formation problems are solved by modifying an initial seed obtained from a similarity matrix. Hierarchical clustering methods employ linkage algorithms that merge initial clusters based on group similarity. Non-hierarchical clustering methods begin with initial seeds and then repeat the procedure of seeding and clustering to obtain better machine cells or part families. This research proposes a new heuristic based on cluster analysis and combining hierarchical and non-hierarchical structures which have simple procedures. From the characteristics of two clustering methods, the proposed heuristic is developed

by adapting the generation of an initial seed from a non-hierarchical method and the incorporation of initial clusters from a hierarchical method.

Since a meta-heuristic has statistical characteristics, the methods based on these heuristics should make many runs on each test problem. Boctor (1991) mentioned that the quality of solutions can be improved by conducting many runs for each problem. Dimopoulos and Mort (2001) proposed the cell formation method based on genetic programming which belongs to the type of evolutionary algorithms. The study reported that 20 runs of the proposed heuristic were made on each problem. Goncalves and Resende (2004) conducted 10 runs of the proposed algorithm based on a genetic algorithm for each problem. Besides, the solution produced by each run may be different. Thus, we need to employ a statistical skill to analyze results obtained by many runs for each problem. It means extra effort and time for solving the cell formation problems. However, the proposed algorithm can obtain the final solution, which is not changed by another run, at only one time run.

Usually, a meta-heuristic is implemented by creating neighborhood solutions. In order to obtain good solutions, numerous neighborhood solutions should be created, modified and evaluated. Thus, as the number of neighborhood solutions generated by a heuristic increases, the probability of finding a better solution can be improved. However, the proposed heuristic is implemented by using only one basis generated from the assignment algorithm based on the pairwise exchange method, which means that the proposed clustering heuristic can obtain the solution in a shorter time than a meta-heuristic. Table 1 summarizes the comparison of the proposed clustering heuristic with

methodologies based on a meta-heuristic. From the characteristics mentioned above, the proposed heuristic can be considered as an efficient method for solving the cell formation problems.

2.4.2 Comparison with the Selected Approaches

This research focuses on the application of cluster analysis for cell formation in CMS. Thus, the comparison with methodologies related to cell formation is needed to mention the efficiency of the proposed heuristic. This research classifies the cell formation methods into five categories: array-based clustering, hierarchical clustering, non-hierarchical clustering, mathematical formulation and algorithm based on a meta-heuristic. Notable solving approaches from these categories are analyzed to compare with the proposed heuristic. Methodologies selected from the literature on cell formation are as follows:

- Array-based clustering: Rank order clustering (ROC) algorithm (King 1980)
- Hierarchical clustering: Single linkage clustering (SLC) algorithm (McAuley 1972)

	Clustering heuristic	Meta-heuristic
Computing time	Good	Bad
Search method	One initial solution	Neighborhood solutions
Solving structure	Iterative procedure	Statistical analysis

Table 1. The comparison of the clustering heuristic with meta-heuristic

- Non-hierarchical clustering: ZODIAC (Chandrasekharan and Rajagopalan 1987)
- Mathematical formulation: The *p*-median model (Kusiak 1987)
- Algorithm based on meta-heuristic: GP-SLCA (Dimopoulos and Mort 2001)

Aspects of the solving procedure, advantages and disadvantages of the selected methodologies are compared in tables 2 and 3. Several comments are provided from the analysis of the selected methods. The contents mentioned above are organized in a table in order to help readers easily understand. Table 2 represents the brief solving procedure and table 3 summarizes advantages and disadvantages on the selected approaches.

Approaches	Solving procedure
Rank order clustering	To group machines and parts by organizing columns and rows in the order of decreasing binary weights
Single linkage clustering	To obtain initial clusters based on a similarity matrix using Jaccard's coefficient and to merge initial clusters. A dendogram is used to find machine or part groups with the highest similarity
ZODIAC	To repeat seeding and clustering from ideal seeds generated by manipulation of artificial and natural seed
The *p*-median model	To use the generalized IP formulation based on process plans for each part with the fixed number of clusters
GP-SLCA	Based on genetic programming and SLC, various similarity coefficients are generated by replacing Jaccard's coefficient with genetic programming, and then these values are input to obtain solution into SLC
Proposed clustering heuristic	Using the new similarity measure, initial clusters are generated, and then these clusters are merged to get a better solution

Table 2. Solving procedure for the selected approaches

Approaches	Advantage	Disadvantage
Rank order clustering	Saves computational time from a simple procedure	Generates bad solutions for even middle-sized problems
Single linkage clustering	Based on a simple procedure and obtains clusters of various sizes by using dendogram	Have the chaining problem
ZODIAC	Generates a better solution from initial seeds based on a quantified evaluation measure called grouping efficiency	Generates bad initial seeds and not proper clustering criterion for ill-structured problem
The *p*-median model	Non-heuristic approach to attain the optimal solution	Not adequate for ill-structured problems and requires long computational time for even middle-sized problems
GP-SLCA	Generates good solutions and tests various similarity coefficients	Requires long computational time and many runs to solve even small-sized problems

Table 3. Advantages and disadvantages for the selected approaches

Based on tables 2 and 3, the following suggestions are recommended from the comparison of the selected methodologies.

- Most approaches, except for rank order clustering, are based on similarity measures to solve cell formation.
- Most approaches, except for GP-SLCA, generate bad solutions for ill-structured problems.
- Most approaches, except for the *p*-median model, employ the solving procedure based on a heuristic.

- Most approaches, except for ZODIAC, do not deal with initial seeds. But adapting a good initial seed as a basis of algorithm can deliver a good solution.
- Approaches based on clustering methods have less computational time than the *p*-median model and GP-SLCA.

Table 4 provides a summary on the comparison of the aspect of the computing time and the goodness of solutions obtained from approaches included in five categories based on the results in table 3.

2.5 Summary

This chapter presents a survey of existing literature on cluster analysis, two types of the cell formation problems in CMS and the comparison of the existing approaches. Heuristics based on cluster analysis are classified as follows: (1) hierarchical clustering, (2) partitional clustering and (3) combinatorial search clustering. Cluster analysis can be applied to many application areas. One among those application areas is cell formation in CMS.

	Array-based clustering	Hierarchical clustering	Non-hierarchical clustering	Mathematical formulation	Meta-heuristics
Solution quality	Bad	Bad	Normal	Normal	Good
Computing time	Good	Good	Good	Bad	Bad

Table 4. Results from five categories of cell formation

In the literature, cell formation is widely classified into two types of problems: CFOPR and CFVMF. This chapter classifies CFOPR as array-based clustering, hierarchical clustering, non-hierarchical clustering, mathematical formulation and heuristics. However, because CFOPR is not adequate to the design of real manufacturing systems, this research introduces CFVMF so that the duplication of machine types and the split of operations are employed to ensure machine capacity.

The need of a new cell formation approach is proposed from a review of the existing cell formation methodologies. Also, this chapter provides several comments recommended from the comparison of selected methodologies.

CHAPTER III

CLUSTERING HEURISTIC

3.1 Introduction

Cluster analysis deals with a sorting problem to partition *m* objects into *n* clusters considering the resemblance relationship for all pairs of objects. Due to the high computational complexity of clustering problems, most algorithms based on cluster analysis are heuristic in nature. In this chapter, a new heuristic based on cluster analysis, which can provide good solutions in a short amount of time by adopting the concept of rough-cut in finding an initial solution, is proposed as a basis to deal with two versions of cell formation. Since the purpose of this research is concerned with the configuration of manufacturing cells, the objective of the proposed clustering heuristic is to obtain clusters maximizing the total sum of values related to elements within clusters. It indicates the minimization of the total sum of values related to exceptional elements, which means intercellular movements, outside clusters.

The proposed clustering heuristic presents a new solving structure that combines and modifies procedures taken from hierarchical and non-hierarchical clustering methods. The agglomerative hierarchical clustering technique controls the entire algorithm in order to obtain the best solution. An ALC algorithm taken from hierarchical clustering methods is employed as a method to merge initial clusters in order to form better clusters. By employing a hierarchical clustering technique, the proposed heuristic can have the flexibility in the decision of the number of clusters. Using an initial seed and an iterative method in non-hierarchical clustering methods, the proposed heuristic is

initiated and better solutions, which are produced from the seed, are obtained. In order to obtain an initial seed, a new similarity coefficient, which is modified from Jaccard's similarity coefficient, and a new assignment algorithm, which is inferred from the concept of rough-cut, are proposed to form initial object clusters.

The proposed clustering heuristic consists of four sub-algorithms as follows: (1) similarity coefficient matrix, (2) assignment algorithm, (3) incorporation and (4) allocation. These sub-algorithms can be summarized as a similarity stage for quantifying resemblances between two objects based on all attributes, an assignment stage for forming initial object clusters, an incorporation stage for obtaining better clusters from initial object clusters, and an allocation stage for allocating attributes to corresponding object clusters.

3.2 New Similarity Measure

A proximity measure is calculated by the relationship between two objects for all attributes in the object-attribute incidence matrix (OAIM) that represents objects in rows and attributes in columns. The relationship of objects and attributes can be represented by binary information, zero or one. A 'one' indicates that an object has the corresponding attribute. A 'zero' indicates that an object does not. The proposed clustering algorithm employs a similarity measure in order to represent the degree of the resemblance between objects and attributes in OAIM. Since the problem focuses on the clustering of objects, the square matrix of similarity coefficients for objects is used as input data to the newly proposed assignment algorithm.

A new similarity coefficient consists of two steps. The first step is to modify the definition of the Jaccard coefficient. The new similarity coefficient is defined as the ratio of the number of attributes having the same indicator on both objects to the number of attributes having indicators equal to 1 on one or both objects. Next, the double centering method is employed to strengthen the discriminating ability of similarity coefficients. Most existing similarity coefficients in the literature have considered only resemblance between two objects. However, by employing the double centering method, the new similarity coefficient is expanded into taking the correlation with other objects into consideration.

3.2.1 Modification Step

- Index and notation

i and j	Objects
k	Attributes
m	The number of objects
p	The number of attributes
s_{ij}	Similarity coefficient between objects i and j

- Binary variable

x_i^k 1, If object i has attribute k
0, otherwise

u_{ij}^k 1, $x_i^k = x_j^k = 1$
0, otherwise

v_{ij}^{k} 1, $x_i^k = 1$ or $x_j^k = 1$
0, otherwise

w_{ij}^{k} 1, $x_i^k = x_j^k = 0$
0, otherwise

McAuley (1972) first employed the Jaccard similarity coefficient, defined as the ratio of the number of attributes included in both objects to the number of attributes included in one or both objects. The Jaccard similarity coefficient was calculated as follows:

$$s_{ij} = \frac{\sum_{k=1}^{p} u_{ij}^{k}}{\sum_{k=1}^{p} v_{ij}^{k}}$$

u_{ij}^{k} can be defined as the matching information having the same indicator of 'one'. The Jaccard coefficient considers only matching information with the same indicator of 'one' as a numerator of the ratio. However, we can recognize w_{ij}^{k} as matching information having the same indicator of 'zero'. Thus, the definition of the Jaccard similarity coefficient can be modified as the ratio of the number of attributes having the same indicator for both objects to the number of attributes having indicators relating '1' in either object *i* or *j* in OAIM. Thus, the '0-0' matching information, w_{ij}^{k}, representing non-existing attributes on both objects, is considered by the definition of the expanded similarity coefficient. Finally, the new similarity coefficient includes matches

having the same indicators such as u_{ij}^{k} and w_{ij}^{k} in the numerator instead of the '1-1' matching information. The new similarity coefficient is defined as follows:

$$s_{ij} = \frac{\sum_{k=1}^{p} u_{ij}^{k} + \sum_{k=1}^{p} w_{ij}^{k}}{\sum_{k=1}^{p} v_{ij}^{k}}$$

By the nature of resemblance between two objects, diagonal coefficients representing a similarity of the same object have the largest value in the similarity coefficient matrix. It leads an assignment of objects into themselves, because the objective of the proposed assignment algorithm is maximization. Thus, the similarity coefficients in the diagonal location have to be replaced with the value of zero in order to prevent a self-loop. Similarity coefficients are dealt as the form of a square symmetric matrix, called a similarity coefficient matrix that is denoted by SM_{mxm}.

3.2.2 Expansion Step

Most similarity measures represent resemblance between a pair of objects. In other words, only the resemblance between two objects for all the attributes affects developing similarity coefficients without considering the correlation with other objects. However, because cluster analysis deals with multivariate data indicating the relationship of many objects and attributes, not only should the relationship with each pair of objects be considered, but also one with other objects in the similarity coefficient matrix. In the proposed heuristic, the double centering transformation is employed to consider the correlation of all the objects. This consideration allows similarity

coefficients to have an improved discriminating ability in the aspect of the comparison with similarities between all the objects in the similarity coefficient matrix. The double centering method is to transform the original data by subtracting an average for row and column and adding a total average in the similarity coefficient matrix. Double centering transformation can be defined as follows:

- Definition

  $s_{i.}$ Average for row i in the similarity coefficient matrix

  $s_{.j}$ Average for column j in the similarity coefficient matrix

  $s_{..}$ Average for all the coefficients in the similarity coefficient matrix

  ms_{ij} Similarity coefficient transformed by double centering

Thus, the new similarity coefficient is calculated as follows:

$$ms_{ij} = s_{ij} - s_{i.} - s_{.j} + s_{..}$$

In the equation above, the transformed similarity coefficient is affected by other objects in the row and column of an object as well as all objects. The new similarity coefficient matrix has a symmetric square form where the sum of each row and column is equal to zero, denoted by MS_{mxm}. Coefficients in the resulting matrix include plus or minus values computed from the similarity coefficient matrix obtained in the previous step. The similarity coefficient proposed in this heuristic is based on the Jaccard similarity and the correlation of all objects and used to form initial object clusters as input data of the new assignment algorithm.

3.2.3 Advantages

This research proposes a new similarity measure based on the Jaccard similarity and the double centering transformation. Two considerations mentioned in sections 3.2.1 and 3.2.2 make the proposed similarity measure have the improved discrimination ability to cluster objects. The proposed similarity measure has advantages in comparison with the Jaccard similarity as follows:

- The matching information having the same indicators of 'one' as well as 'zero' are considered.
- The relationship between two objects as well as the correlation of all the objects is considered by employing the double centering transformation.

3.3 New Assignment Algorithm

A classical assignment problem is to match *n* persons and *n* objects in the form of a one-to-one basis to maximize the total benefit. The assignment problem, which is a special case of the transportation problem, is employed in many practical cases. The proposed algorithm is inferred from the concept of rough-cut. Thus, the algorithm attempts to find a near optimal solution in a short amount of time. The objective of the proposed algorithm is to find initial clusters that maximize the total sum of similarity coefficients over all possible assignments of given objects. The proposed assignment algorithm uses the modified similarity coefficient matrix, MS_{mxm}, as input data and is based on a pairwise exchange method and an iterative solving procedure in order to obtain initial object clusters.

New assignment algorithm consists of three parts; the first part is the selection of two objects to be exchanged, the second is the assignment of two selected objects into the same cluster, and the third is the subtraction of margin for the selected object. In the selection part, an algorithm chooses a pair of objects with the maximum increment for pairs of all objects. In the assignment part, the two objects which are already chosen are exchanged with each other and assigned into the same cluster. Then, in the subtraction part, from all the column coefficients of the object having a larger margin out of the two objects having the maximum increment, the corresponding margin is subtracted.

In each iteration, the objective is always a non-decreasing function because only a pair of objects having a non-negative increment is considered for exchange as a candidate object pair. Thus, the speed of the proposed algorithm is faster than other methods having ups and downs in the objective function. The proposed algorithm employs two termination conditions which prevent the new algorithm from resulting in never-ending iteration. By utilizing the concept of rough-cut in the proposed assignment algorithm, an optimal solution for large problems may not be obtained, but the near optimal in a considerably shorter amount of time. The optimal or near optimal solution is considered as an initial seed that is converted as the best solution through the entire clustering heuristic. In order to describe the new assignment algorithm, some notations are required as follows:

- Notation

  s and *t* Index of an object in similarity coefficient matrix

  k Index of a cluster

m	The number of objects
p	The number of attributes
mv	Maximum increment value
ci_s	Set of indexes on row and column of an object assigned as a cluster indicator in row s
a_s	Similarity coefficient of an object assigned as a cluster indicator in row s
u_s^t	Similarity coefficient of an unassigned object located on the same column as a_t in row t out of elements in row s
d_s^t	Difference of similarity coefficients for a object s, $d_s^t = u_s^t - a_s$
C_k	Set of objects assigned into a cluster k

3.3.1 Initialization

The proposed assignment algorithm begins from the assignment of cluster indicators and initial clusters. A cluster indicator represents objects assigned into a cluster and is denoted by ci_s, where s indicates a row in the similarity coefficient matrix. Row and column indices, i and j, for a cluster indicator represent objects within a cluster. For example, if indexes corresponding to a cluster indicator are row 1 and column 2 in the similarity coefficient matrix, objects 1 and 2 are assigned into the same cluster. In this algorithm, clusters are initialized by having only one object. The procedure of the initialization is described as follows:

Step 1.1 Cluster indicators for all rows in the similarity coefficient matrix are defined. In this algorithm, each diagonal element is assigned as a cluster indicator.

$$ci_1 = \{1, 1\}, ci_2 = \{2, 2\}, \ldots, ci_m = \{m, m\}$$

Step 1.2 Cluster indicators are assigned into clusters. In the initialization, one cluster has to include only one indicator.

$$C_1 = \{ci_1 = 1\}, C_2 = \{ci_2 = 2\}, \ldots, C_m = \{ ci_m = m\}$$

Step 1.3 A similarity coefficient of element assigned as ci_s in row s is defined. As mentioned, ms_{ij} is a new similarity coefficient after double centering transformation between objects i and j.

$$a_1 = ms_{11}, a_2 = ms_{22}, \ldots, a_m = ms_{mm}$$

3.3.2 Selection

The selection part out of three main parts, which employs a pairwise exchange method, is a core part of this assignment algorithm. The pairwise exchange is a method used to compare similarity coefficients of all object pairs in order to select an object pair with the maximum increment. Before implementing the pairwise exchange method, assigned and unassigned objects for all rows should be defined. Figure 1 illustrates an example of the assigned and unassigned objects. As mentioned, the assigned objects, which can also be called cluster indicators, are already decided in the previous step. Unassigned objects can be defined as objects other than the assigned object in each row.

In order to calculate the increment of the objective value generated by exchanging two assigned objects, two differences, which are calculated by two assigned objects to be exchanged, should be considered. Each difference is defined by subtracting the coefficient of each assigned object from the coefficient of unassigned objects located on the column corresponding to another object in an object pair out of elements in the

same row as each assigned object. If the sum of two differences calculated from a pair of objects is not negative, the object pair is considered as a candidate for exchange. Then an object pair with the largest increment among all candidates is selected as the best object pair improving the objective value. The similarity coefficient matrix after the pairwise exchange of object A and C is represented in figure 2. As a result, the selection part plays a role in the selection of two objects to exchange cluster indicators and is described as follows:

Step 2.1 Similarity coefficients of unassigned elements are defined.

$$u_s^t = ms_{st}$$

Here, ms_{st} is a similarity coefficient other than a_s in a row s and t is an index indicating the column corresponding to a_t in row t out of elements in a row s.

Step 2.2 The differences of similarity coefficients between assigned and unassigned objects are calculated for all object pairs.

$$d_s^t = u_s^t - a_s \text{ and } d_t^s = u_t^s - a_t, \text{ for all } s \text{ and } t.$$

	A	B	C
A	u_A^C	a_A	u_A^B
B	u_B^C	u_B^A	a_B
C	a_C	u_C^A	u_C^B

Figure 1. Example of the assigned and unassigned objects in the similarity coefficient matrix

Step 2.3 Object pairs, *s* and *t*, with the nonnegative sum of two differences are considered as candidates.

$$d_s^t + d_t^s = u_s^t - a_s + u_t^s - a_t \geq 0$$

Step 2.4 An object pair with the maximum increment is selected as the best.

$$mv = arg\ max\ \{d_s^t + d_t^s\}$$

3.3.3 Assignment

In the previous step, the best object pair was selected, and the corresponding assigned and unassigned objects were known. In this step, the unassigned objects located on the columns of two objects known as the best object pair in the selection part become new assigned objects, and the object newly assigned in each row is defined as a new cluster indicator. As mentioned, a cluster indicator represents objects to be assigned into a cluster. Thus, new clusters are formed by assigned objects newly known as cluster indicators. As results, new clusters are decided by the indexes indicating the best object pair having the maximum increment.

	A	B	C
A	a_A	u_A^C	
B			a_B
C	u_C^A	a_C	

Figure 2. Example of the assigned and unassigned objects after the pairwise exchange of object A and C

Figure 2 illustrates the assigned and unassigned objects newly defined after pairwise exchange of the best object pair, A and C. By the result in figure 2, cluster indicators are renewed as ci_1 = {A, A}, ci_2 = {B, C} and ci_3 = {C, B}. Since cluster indicators are decided, new clusters can be obtained by cluster indicators as C_1 = {A} and C_2 = {B, C}. This procedure can be described as follows:

Step 3.1 Let row and column indexes of new assigned objects be *s* and *t*. A new cluster indicator is renewed by a new assigned object.

$$ci_{new} = \{s, t\}$$

Step 3.2 A new cluster is obtained by a new cluster indicator.

$$C_{new} = \{s, t\}$$

3.3.4 Subtraction

The proposed assignment algorithm generates new clusters from initial clusters through an iterative procedure in order to obtain a better solution. But if the number of iterative steps is excessive, the algorithm consumes much more computational time and may not obtain a solution within a reasonable amount of time. So the trade-off between the quality of solution and the computational time is required. In this algorithm, any subtraction is used to balance the time to iterate procedures and the quality of solution. As mentioned, the proposed assignment algorithm is based on the concept of rough-cut in order to obtain a solution used as an initial input in the entire clustering heuristic. Hence, this algorithm focuses on obtaining a solution within a short amount of time, but with a good solution quality.

The subtraction of a proper value can prevent the exchange of the object pair already assigned as the best object pair from successively regenerating or never ending, which leads to reduce the number of iterative steps. Figure 3 represents a similarity coefficient matrix for a two-object problem with two cluster indicators, and table 5 illustrates never-ending iteration for the exchange of the same object pair in the matrix shown in figure 3. As shown in table 5, without any subtraction, the proposed algorithm is not terminated. Thus, the proposed assignment algorithm employs the mechanism to prevent non-termination of algorithm. Usually, the amount of a subtractive value affects the speed of algorithm. The bigger a value is, the faster the speed is. However, the bigger a value is, the worse the quality of solution gets.

	1	2
1	$4 \Leftarrow ci_1$	5
2	4	$5 \Leftarrow ci_2$

Figure 3. Similarity coefficient matrix for a two-object problem

Iteration	Cell Indicator	Max. Increment
0	$ci_1 = \{1\}$ and $ci_2 = \{2\}$	$mv = d_1^2 + d_2^1 = 1 + (-1) = 0$
1	$ci_1 = \{1, 2\}$	$mv = d_1^1 + d_2^2 = (-1) + 1 = 0$
2	$ci_1 = \{1\}$ and $ci_2 = \{2\}$	$mv = d_1^2 + d_2^1 = 1 + (-1) = 0$
3	$ci_1 = \{1, 2\}$	$mv = d_1^1 + d_2^2 = (-1) + 1 = 0$
…	…	…

Table 5. Illustration of non-termination for successive exchanges of the same objects based on a similarity coefficient matrix in figure 3

In the subtraction part, first, a bigger difference value, so called the maximum difference, out of two differences calculated from the best object pair is calculated. Then, the maximum difference is subtracted from coefficients in the column corresponding to a new assigned object with a bigger difference value out of objects selected as the best pair. The procedure of subtraction can be described as follows:

Step 4.1 Two differences of two objects in the best pair are compared. A bigger one is selected as a subtraction value.

$$\text{Maximum difference} = \max \{d_s^t, d_t^s\},$$

Step 4.2 The maximum difference is subtracted from all coefficients in the column corresponding to the new assigned object with the maximum difference.

3.3.5 Termination Conditions

The proposed assignment algorithm employs two termination conditions in order to increase the objective value and prevent never-ending cycle. These termination conditions are in the selection and subtraction part.

3.3.5.1 Selection Part

The first termination condition is concerned with whether the maximum increment value is negative or not. As mentioned, this algorithm has the mechanism that the objective value, the total sum of similarity coefficients, never decreases. The non-decreasing objective value is updated by adding the maximum increment iteratively. If the maximum increment is negative, the objective value is not improved, that is, a value decreases. Hence, the maximum increment should be not negative.

We can consider three cases for the maximum increment value. First, the maximum increment is positive. This case enables the objective value to be always improved. Thus, the proposed algorithm is implemented in order to search better solutions. Second, the maximum increment is equal to zero. That means that the objective value is not improved. Although a better solution does not exist, alternative solutions can be obtained. Thus, the algorithm accepts solutions with the same objective value. Finally, the maximum increment is negative. The current assignment gives the maximum of total sum of similarity coefficients. Thus, the algorithm is terminated.

3.3.5.2 Subtraction Part

The second termination condition is related to an object with a bigger difference out of two objects assigned as the best pair and employed to prohibit a never-ending cycle which is a state wherein the same assignment pattern is repeated iteratively. In the case of the never-ending cycle, new assignment with the better objective value is not formed. Thus, this state should be prohibited from repeating. The second termination condition is valid under the condition that the maximum increment is not negative. Hence, if the maximum increment, which is equal to the sum of two differences of the best pair, is positive, at least one difference out of two differences should be positive; if the maximum increment is equal to zero, a bigger difference is equal to zero or a positive value. Thus, the second condition has two cases.

Consider the case that the maximum increment is not negative. First, if the bigger difference is positive, the proposed algorithm generates a new assignment. Second, the bigger difference is equal to zero. Although the difference value is subtracted from all

coefficients in the column corresponding to an object with the bigger difference, since the difference is equal to zero, there is no change any more. Thus, a never-ending cycle is created without changing the objective value. From this consideration, we have to consider only the case that the bigger difference is positive.

3.3.6 Update and Repeat

After the completion of any iteration of the proposed assignment algorithm, several data are generated to control and implement the proposed algorithm. The objective value increases by the maximum increment, *mv*, and clusters are renewed by new cluster indicators. A similarity coefficient matrix also is changed by subtracting the maximum difference. The updated data is kept for the next iteration, and the proposed assignment algorithm is repeated from a selection step to an update step until the algorithm is terminated.

3.3.7 Procedure of the Assignment Algorithm

The proposed assignment algorithm was described above. This algorithm consists of three parts, selection, assignment and subtraction, as core parts and two parts, initialization and termination, as supplement parts. The similarity coefficient matrix obtained by modifying the Jaccard's similarity is used as an input data to generate initial seed clusters. Based on the pairwise exchange method, an iterative procedure is employed to obtain better solutions from the given data. At each iteration, the objective value that tends to be improved and the related data are updated and kept. The objective function is to maximize a total sum of similarity coefficients. Object clusters chosen as

the best solution are used as a basis in the entire procedure of the clustering heuristic. The iterative procedure of the assignment algorithm is illustrated in figure 4.

3.4 Incorporation

In order to form better object clusters, initial clusters obtained from the assignment step are merged according to a group similarity coefficient, which is based on Jaccard's similarity coefficient, calculated by an ALC algorithm (Seifoddini and Wolfe 1986, 1987). Employing the incorporation step enables the clustering heuristic to have the flexibility in the decision of the number of clusters or objects within a cluster. Originally, the proposed clustering heuristic does not have a limit to clusters or objects. However, if the number of clusters or objects is constrained, the constraints can be satisfied with an ALC algorithm in the incorporation step. For example, let the number of clusters be limited to three. Then, the incorporation step is conducted until the number of clusters is more than two, and then the entire heuristic is terminated. In the case of singleton, which has only one object within a cluster, the object is first merged to an object-cluster having the largest group similarity coefficient. When the incorporation is not available any more, the entire clustering algorithm is terminated.

3.5 Allocation

In the allocation part, attributes are allocated into object clusters using the maximum density rule that guarantees to generate clusters maximizing a total sum of 'one' entries within clusters. This rule leads the minimization of intercellular movements between clusters. If a tie exists, attributes are arbitrarily allocated.

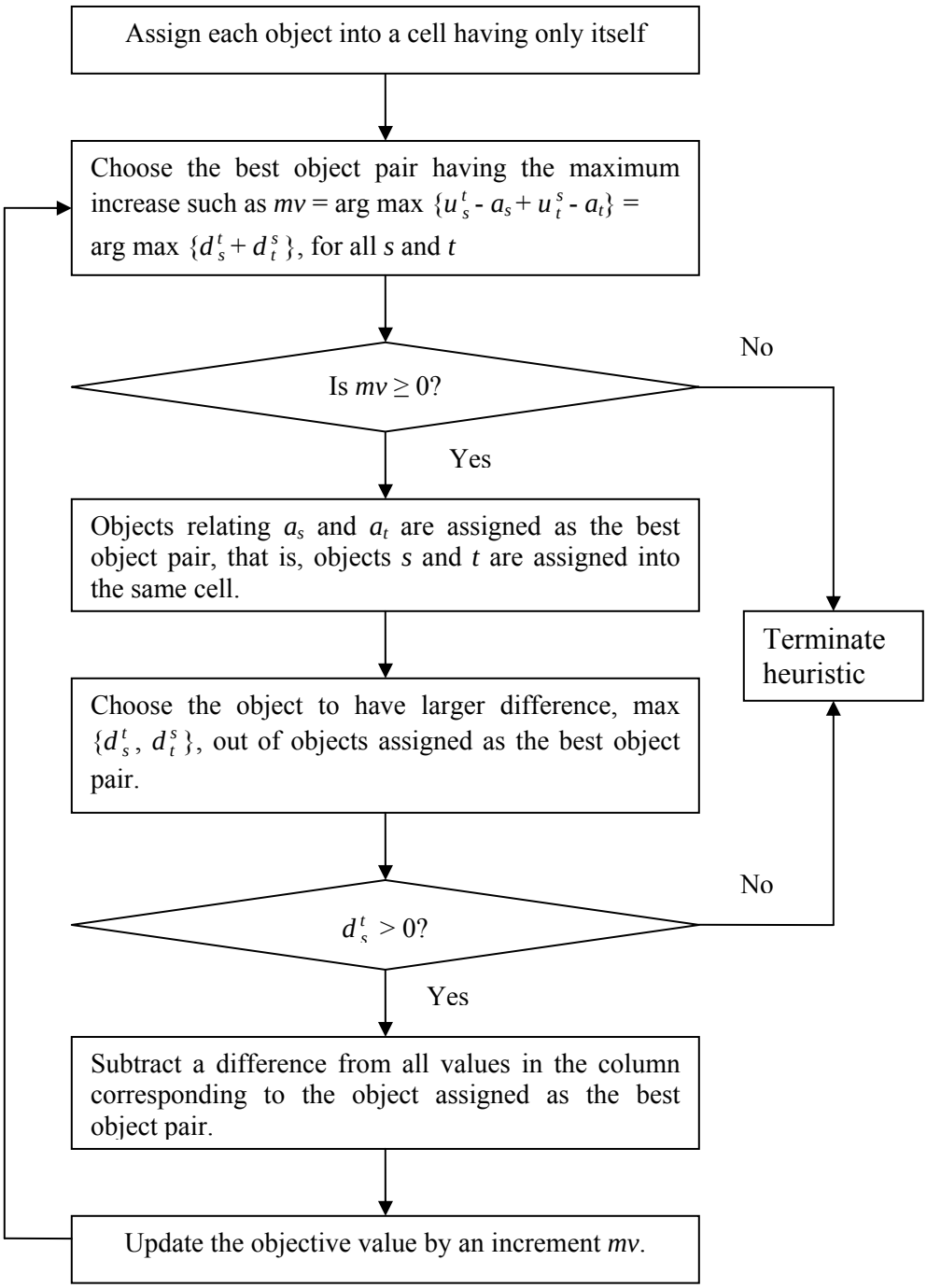

Figure 4. Illustration of the proposed assignment algorithm

- The maximum density rule

    a. For each attribute, compute the sum of ‘one’ entries on each cluster.

    b. Allocate each attribute into an object-cluster having the maximum value.

    c. If the same value exists, allocate arbitrarily attributes into the corresponding object cluster.

3.6 Procedure of the Clustering Heuristic

This section presents the entire procedure of the proposed clustering heuristic which forms object-attribute clusters through an iterative procedure. The clustering heuristic consists of the following four sub-algorithms:

SA1. Development of the similarity coefficient matrix:

A new similarity measure is constructed by modifying the Jaccard similarity coefficient and employing the double centering transformation. A similarity coefficient matrix is used as input data to an assignment algorithm.

SA2. Generation of initial object clusters from the assignment algorithm:

The new assignment algorithm is developed to generate initial object clusters. The new pairwise exchange method is employed to obtain better solutions from the given data through an iterative procedure. At each iteration, the objective value and the related data are updated and kept.

SA3. Incorporation of initial object clusters using the ALC algorithm:

Initial object clusters, which are generated from the assignment algorithm, are

merged and modified to get better solutions.

SA4. Allocation of attributes into object clusters using the maximum density rule:

Attributes are allocated to the merged object clusters by the maximum density rule. Then, the obtained object-attribute clusters are kept as a solution.

Through an iterative procedure, a solution at any iteration is evaluated in the aspect of the number of intercellular moves and compared with a solution at next iteration. From the comparison, better solution is chosen and saved. The heuristic is terminated when an incorporation step is not available. The entire procedure of clustering heuristic is briefly illustrated in figure 5.

3.7 Summary

This chapter proposes the clustering heuristic of which the objective is to minimize the intercellular movements. The proposed clustering heuristic is developed by a new solving structure combining and modifying hierarchical and non-hierarchical clustering methods. A new similarity coefficient is proposed by adding the matching information having the same indicators of ‘zero’ and employing the double centering transformation. And, the relationship between two objects as well as the correlation of all the objects is considered, which makes a new similarity measure have more discriminating ability.

The new assignment algorithm is proposed to generate initial object clusters. In the assignment algorithm, the new pairwise exchange method and the rough-cut concept

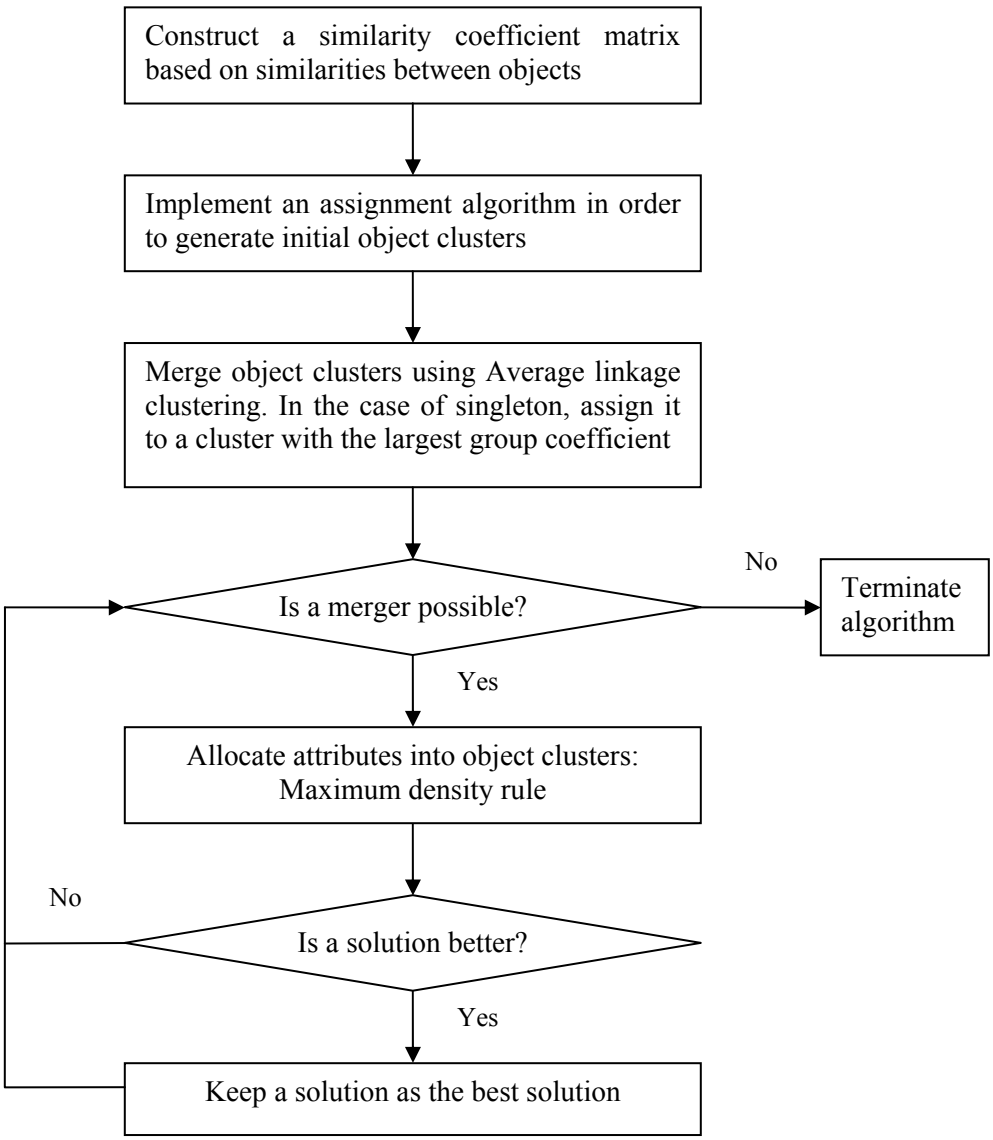

Figure 5. Illustration of the clustering heuristic

are employed to reduce computational time. Two termination conditions are employed to increase the objective value and prevent a never-ending cycle in the selection and subtraction part. In order to merge initial object clusters, the clustering heuristic employs ALC algorithm. Then attributes are allocated to the corresponding object clusters by the maximum density rule. Through an iterative procedure, the proposed clustering heuristic produces the best solution based on the number of intercellular movements.

CHAPTER IV

APPLICATION FOR CELL FORMATION CONSIDERING PROCESSING REQUIREMENT

4.1 Introduction

The objective of this dissertation is to develop an efficient clustering heuristic and apply the proposed heuristic for cell formation which includes the configuration and capacity management of manufacturing cells. Cell formation can be classified into two kinds of problems. One is a problem considering only processing requirement, CFOPR, and another is a problem considering a variety of manufacturing factors, CFVMF.

This chapter provides the application of the proposed clustering heuristic for CFOPR between machines and parts. This problem is a basis for the configuration of manufacturing cells. CFOPR is an approach to group parts and machines by identifying similarities and dissimilarities between all part pairs and machine pairs. In other words, after being calculated the similarity coefficients of all pairs of parts or machines, pairs of parts or machines with the largest similarity coefficient are allocated to the same families or cells, respectively.

The objective of the proposed heuristic is to minimize intercellular movements. However, many evaluation measures have been reported in CFOPR, and many researchers have employed these measures to evaluate their approaches and compare with other methods. Thus, this application also employs a well-known evaluation measure quantifying the goodness of obtained solutions. The heuristic proposed to solve CFOPR adds a feedback step to the clustering heuristic described in Chapter III. A

feedback step enforces the quality of solutions obtained from the clustering heuristic. The feedback step can be described as an opposite concept of the allocation step in the proposed clustering heuristic. Also, the allocation part is modified by adding more specific rules that are efficient to allocate parts. The entire heuristic procedure is iterated until the incorporation step is not available. Before describing the solving procedure, backgrounds for cell formation in CMS are provided.

4.2 Background

4.2.1 Machine-Part Incidence Matrix

In order to solve CFOPR, the processing requirement, which is taken from the route cards for parts, between machines and parts in a MPIM is utilized. A MPIM shows the relationship between machines and parts (King 1980). In other words, a MPIM can be regarded as a given data set of any number of different machines and different parts and denoted by a matrix A_{mxn}, where m is the number of machines and n is the number of parts. Each element in a MPIM is usually represented by x_{ij}, which represents an operation for part j on machine i, and may have the entry of '0' or '1'. A '1' entry indicates that a part requires processing on a machine, and a '0' entry indicates that a part does not require processing. In figure 6 illustrating a MPIM, the elements indicating part 1 on machines C and E are equal to '1'. This means that part 1 requires processing on machines C and E, and the related elements, x_{C1} and x_{E1}, are represented by '1'. However, the coefficients indicating machines that part 1 does not require to be processed on are represented by '0'.

		Parts					
		1	2	3	4	5	6
Machines	A	0	0	1	0	1	0
	B	0	1	1	0	0	0
	C	1	0	0	1	0	0
	D	0	1	1	0	1	0
	E	1	0	0	1	0	1

Figure 6. Machine-part incidence matrix

4.2.2 Block Diagonalization

As mentioned, the purpose of cell formation is to group machines and parts with similar characteristics into the same cell in order to minimize intercellular travel of parts. In order to attain this goal, all the entries of '1' in a MPIM are collected in the diagonal blocks, and all the entries of '0' are collected in the off-diagonal blocks. This can be achieved by attempting a block diagonalization, which has an advantage of visualization in practice, on zero-one elements in the given MPIM. In the process of the block diagonalization coefficients with a '1' value are grouped to form mutually exclusive, independent, clusters, and coefficients with a '0' value are arranged outside these clusters. After the block diagonalization, an initial MPIM is converted into a diagonal arrangement of mutually exclusive clusters. After the block diagonalization of a 5 x 6 matrix in figure 6, figure 7 presents the resulting matrix having two mutually exclusive cells. As a result, CFOPR can be regarded as the problem to solve the original MPIM in order to obtain mutually exclusive part and machine cells. However, it is impossible to

obtain mutually exclusive clusters in practice. Therefore, the objective of the block diagonalization is to change the original MPIM into a matrix form minimizing exceptional elements. Figure 8 presents another version of the diagonal block form. The MPIM shown in figure 8 includes an intercellular movement represented by part 1 on machine A. The operation related to this intercellular movement requires extra cost/time, and it is called an exceptional element.

		Parts					
		1	4	6	2	3	5
Machines	C	1	1	0	0	0	0
	E	1	1	1	0	0	0
	A	0	0	0	0	1	1
	B	0	0	0	1	1	0
	D	0	0	0	1	1	1

Figure 7. Block diagonalized machine-part incidence matrix

		Parts					
		1	4	6	2	3	5
Machines	C	1	1	0	0	0	0
	E	1	1	1	0	0	0
	A	1	0	0	0	1	1
	B	0	0	0	1	1	0
	D	0	0	0	1	1	1

Figure 8. Example of an intercellular movement in a MPIM

4.2.3 Similarity Coefficient

A similarity coefficient shows the degree of resemblance between two parts or machines in the production process. In other words, the resemblance between two parts represents how many identical machines are used in the production processes of the parts, and the resemblance between two machines represents how many identical parts are processed on the machines. A similarity coefficient usually represented by s_{ij}, which indicates the degree of resemblance between parts *i* and *j* or machines *i* and *j*, can be derived from a MPIM. This similarity coefficient is utilized as an important measure to group machines and parts.

Sarker (1996) reviewed and compared the characteristics of a variety of similarity and dissimilarity coefficients from the literature. Moier *et al*. (1997) reviewed a metric form of similarity coefficients and classified them as assignment, weighted assignment, and processing order information. Khan *et al*. (2000) evaluated performance on existing similarity coefficients using matching properties and proposed a new similarity coefficient and grouping model. Yasuda and Yin (2001) mentioned the main deficiencies of the Jaccard similarity coefficient and commonality score (Wei and Kern 1989) and proposed a new dissimilarity coefficient called an average voids value. In cell formation, similarity coefficients are dealt with as the form of a square matrix that is illustrated by SM_{mxm}, where *m* is the number of machines.

4.2.4 Goodness of Clustering

In addition to an exceptional element, another factor considered for the cell formation in binary version is the number of a void such as a zero element indicating

part 2 on machine A in figure 8. Suppose all parts and machines in figure 8 are grouped into one cluster in order to remove exceptional elements. In the aspect of the utilization of machines, since machines should process required parts as well as unnecessary parts, the utilization of machines decreases naturally. For example, in the case that machine E includes parts 1, 4 and 6 to process as well as parts 2, 3 and 5, the utilization of machine E decreases more than when processing parts 1, 4, and 6. Usually, a void ruins the machine utilization within a cell. Thus, the primary objective of the cell formation problem is to minimize intercellular moves and maximize machine utilization. As mentioned, the formation of clusters with completely independent part and machine cells is not practical. Thus, in order to minimize the intercellular moves between cells as well as to maximize the machine utilization within a cell, a trade-off between two considerations is always needed.

4.3 Clustering Heuristic for Solving CFOPR

4.3.1. Procedure of Heuristic

This section proposes the modified clustering heuristic for solving CFOPR. The entire procedure is similar to the clustering heuristic proposed in Chapter III. In order to efficiently deal with CFOPR, the original heuristic is modified by adding the feedback step that enforces the goodness of the obtained solution. In the procedure of the proposed heuristic, initial clusters generated from the assignment algorithm, which uses the similarity coefficient matrix based on processing requirement as input data, are updated and merged to obtain better solutions. After initial machine clusters are incorporated by the ALC algorithm, parts are allocated into merged machine cells by allocation rules.

Then clusters are evaluated by using a popular measure called the grouping efficacy (Kumar and Chandrasekharan 1990) in this field. The obtained machine-part clusters and a grouping efficacy value are saved to compare with other solutions. If a better solution exists, a feedback step is implemented. Otherwise, an incorporation step is repeated to merge other clusters. When an incorporation step is no longer available, the entire clustering heuristic for solving CFOPR is terminated. The proposed clustering heuristic consists of six sub-algorithms as follows:

SA1. Construction of similarity coefficient matrix

SA2. Assignment to obtain an initial machine cells

SA3. Allocation of parts into the corresponding machine cells

SA4. Incorporation of initial machine cells using the ALC algorithm

SA5. Evaluation of obtained solutions

SA6. Feedback to adjust machine cells based on part families

In order to form better machine-part clusters, this algorithm employs the iterative procedure so that the clusters obtained at any iteration are compared by an evaluation measure and the best result is kept. Figure 9 illustrates the procedure of the clustering heuristic proposed to solve CFOPR.

4.3.2 Similarity Coefficient Matrix

The clustering heuristic proposed to solve CFOPR starts from constructing the similarity coefficient matrix based on the MPIM taken from the route card for parts. Since CFOPR considers only the relationship of processing requirement between parts

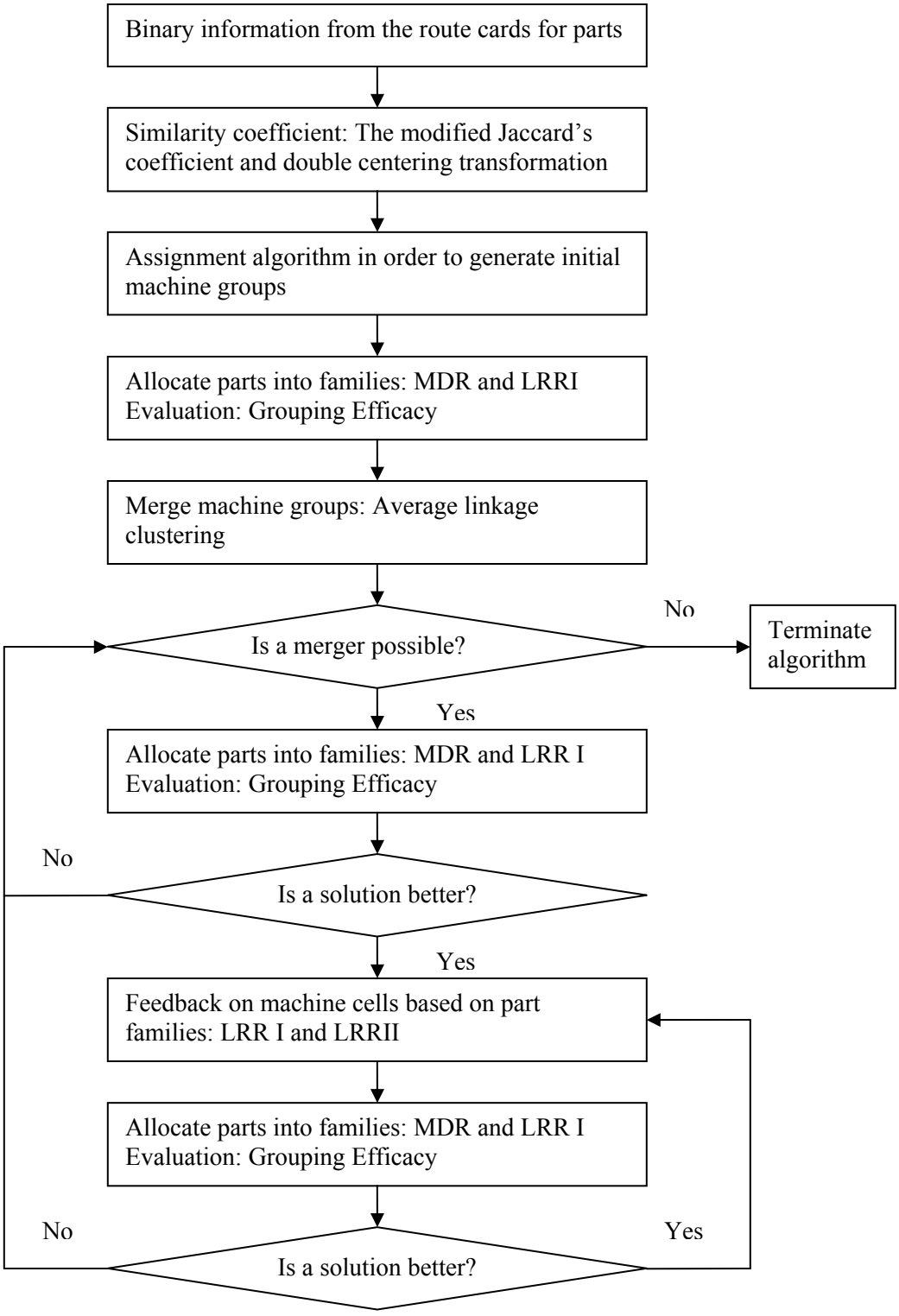

Figure 9. The conceptual illustration of the clustering heuristic for CFOPR

and machines, a similarity measure can be calculated from data set represented by zero or one. A '1' entry indicates that a part requires processing on a machine, and a '0' entry indicates that a part does not require processing. In this case, the definition of similarity coefficient is similar to that of the clustering heuristic defined in Chapter III. However, CFOPR does not consider the relationship of objects and attributes, but machines and parts. Thus, the similarity measure defined in the clustering heuristic can be modified as follows:

- Index and notation

i and j	Index of machines
k	Index of parts
p	The number of parts
s_{ij}	Similarity coefficient between machines i and j

- Binary variables

x_i^k	1, If part k is processed on machine i 0, otherwise
u_{ij}^k	1, $x_i^k = x_j^k = 1$ 0, otherwise
v_{ij}^k	1, $x_i^k = 1$ or $x_j^k = 1$ 0, otherwise
w_{ij}^k	1, $x_i^k = x_j^k = 0$ 0, otherwise

Thus, the new similarity coefficient is defined as follows:

$$s_{ij} = \frac{\sum_{k=1}^{p} u_{ij}^{k} + \sum_{k=1}^{p} w_{ij}^{k}}{\sum_{k=1}^{p} v_{ij}^{k}}$$

The similarity coefficients in the diagonal location have to be replaced with the value of zero in order to prevent a self-loop. Then this similarity measure is transformed by the double centering method. Similarity coefficients are dealt with as the form of a square symmetric matrix, called a similarity coefficient matrix.

4.3.3 Assignment

The new assignment algorithm already described in the clustering heuristic is employed to generate initial machine clusters using similarity information for machine pairs as input data. The assignment algorithm, which is based on the pairwise exchange method, the rough-cut concept and an iterative procedure, consists of three parts: the selection of two machines to be exchanged, the assignment of two selected machines into the same cluster, and the subtraction of margin for the selected machine. Based on the obtained initial machine clusters, initial machine-part clusters are obtained by allocating parts into part families in the following step.

4.3.4 Allocation

Parts are allocated into part families using the maximum density rule and the largest ratio rule I. Most existing cell formation methods allocate parts or machines into the corresponding cells or families by computing only the maximum density that guarantees the minimization of exceptional elements. However, when a tie exists, parts

or machines are arbitrarily assigned. In this heuristic, the largest ratio rule I is employed as a more exact method for the existence of the same density. The largest ratio rule I is employed to evaluate the efficiency of clusters related to the machine utilization that represents the workload of machines for the given machine-part clusters. Thus, parts are allocated to a machine cluster having the smallest number of machines among clusters having the same majority. The allocation step is described as follows:

• Procedure of the allocation

i) The maximum density rule is first adapted.

a. For each part, compute the number of operations on each machine cell.

b. Allocate each part into a machine cell having the maximum value.

ii) If the same value exists, the largest ratio rule I is used.

a. For each part, compute the ratio corresponding to each machine cell as follows:

$$R = \frac{\text{The number of operations on each machine cell}}{\text{The number of machines on each machine cell}}$$

b. Allocate each part into a machine cell having the largest ratio.

iii) If the same value exists, arbitrarily allocate parts into the corresponding machine cell.

4.3.5 Incorporation

In order to form better machine clusters, initial machine clusters obtained from the assignment step are merged to assign machines having similar characteristics into the same cluster. At this time, a clustering method using a group similarity coefficient calculated by the ALC algorithm is employed. The ALC algorithm was adapted by

Seifoddini and Wolfe (1986 and 1987) to group similar clusters. Seifoddini (1988 and 1989) suggested the use of the ALC algorithm in order to overcome a chaining problem in the SLC algorithm. An average similarity coefficient based on the Jaccard's similarity coefficient is represented as follows:

$$s_{uv} = \frac{\sum_{i \in u} \sum_{j \in v} s_{ij}}{N_u \times N_v}$$

Where N_u is the number of machines in a machine cluster u, N_v is the number of machines in a machine cluster v and s_{ij} is the Jaccard's similarity coefficient shown in the section 3.2.1.

In the incorporation step, the number of clusters or machines within a cluster can be limited by inserting the constraint into the proposed heuristic. If a singleton, which has only one machine within one cluster, exits in the initial machine clusters, the machine is merged into a cluster with the largest group similarity coefficient. When this step is not available, the entire heuristic is terminated.

4.3.6 Evaluation

Grouping efficacy, which is introduced by Kumar and Chandrasekharan (1990), is the most notable measure in the literature on CFOPR. This measure has been popularly used to evaluate the proposed algorithm and compare it with other clustering methods. Grouping efficacy, which attempts to reduce exceptional elements and voids, quantifies the degree of the goodness on clusters of machines and parts. The objective of CFOPR is to minimize the number of intercellular movements and maximize the

machine utilization. In order to reduce intercellular moves and increase the machine utilization, the number of exceptional elements and voids should be reduced. Thus, grouping efficacy is very useful as a measure to evaluate the goodness of a solution in CFOPR. This measure is represented as follows:

$$\text{Grouping efficacy} = \frac{(e - e_e)}{(e + e_v)}$$

Where,

e = the number of operations

e_e = the number of exceptional elements

e_v = the number of voids

At all the iterations, a grouping efficacy value based on the obtained machine-part clusters is calculated and compared with the best solution in the previous iteration. From the comparison of two values, a better grouping efficacy value and the corresponding machine-part clusters are kept.

4.3.7 Feedback

A feedback step is employed to get the improved solution for machine-part clusters obtained from the allocation step. By iterating the feedback procedure, the heuristic proposed to solve CFOPR can generate the assignment based on machine cells as well as part families. Thus, the machine-part clusters that are acceptable and well-structured in the two aspects of machine cells and part families are obtained.

Based on part families, the feedback step tests machine cells on how well machine cells are grouped. In other words, a feedback algorithm assigns each machine into part families with the largest number of operations related to the machine, which is opposite to the part allocation that implements allocation of parts into machine cells. Then, parts are again allocated into the newly constructed machine cells in order to obtain better machine-part clusters. In this step, the largest ratio rule I and II are employed to regroup machines into machine clusters. But the largest ratio rule I is modified by exchanging a machine cell with a part family. The largest ratio rule II is related to the utilization of a machine-part cluster instead of the machine utilization in the largest ratio rule I for a machine.

• Procedure of the feedback

i) The largest ratio rule I is used to test the efficiency of machine cells.

a. For each machine, compute the ratio as follows:

$$R = \frac{\text{The number of operations on each part family}}{\text{The number of parts on each part family}}$$

b. Allocate each machine into part families having the largest ratio.

ii) If the same value exists, the largest ratio rule II is used.

a. For each machine, compute the ratio as follows:

$$R = \frac{\text{The number of operations on each part cell}}{\text{The number of parts on each part cell} * \text{the number of machines on each part cell}}$$

b. Allocate each machine into part families having the largest ratio.

iii) If the same value exists, arbitrarily allocate parts into the corresponding machine cell.

The obtained clusters are evaluated and compared through an iterative procedure. If a better solution exists, the feedback step is repeated until none exist. Otherwise, the algorithm goes back to the incorporation step in order to merge the remaining possible clusters.

4.4 Performance Evaluation

In order to demonstrate the performance of the clustering heuristic proposed to solve CFOPR, this dissertation utilizes two kinds of evaluation methods. These are experimented by evaluating the goodness of clustering machines and parts and the computational complexity of the proposed heuristic. One is to compare machine-part clusters obtained from the clustering heuristic with other notable cell formation methodologies for test problems taken from the literature in the aspect of an evaluation measure. Another is to evaluate the computational complexity for various sized MPIMs having two levels of difficulties and the computing time spent in obtaining solutions from the given test problems.

As results of the comparison with other approaches and the evaluation of the computational complexity, the clustering heuristic proposed to solve CFOPR shows that the heuristic has an ability to find good quality machine-part clusters in a short time in the configuration of CFOPR in CMS.

4.4.1 Test Data Set

In order to evaluate the goodness of clusters obtained from the clustering heuristic for CFOPR, 23 problems taken from the literature were tested. These data sets include a variety of sizes, a range from 5 machines and 7 parts to 40 machines and 100 parts, difficulties, and well structured and ill structured matrices. Table 6 represents the sizes and sources of data sets taken from the literature in order to compare with other cell formation methods. These data can be easily accessed from the references mentioned in table 6.

Boctor (1991) and Chandrasekharan and Rajagopalan (1989) provided a variety of data sets to evaluate the effect of the well and ill structured MPIM for GT approaches. In this research, their data are employed to evaluate the clustering heuristic for CFOPR. Problems 8 to 14 are equivalent to problems 2 to 6, 8 and 9 in Boctor's paper, and problems 18 to 22 are equivalent to problems 1 to 3, 5 and 6 in Chandrasekharan and Rajagopalan's paper. In order to evaluate the performance for the large-sized cell formation problem, a problem including 40 machines and 100 parts taken from Chandrasekharan and Rajagopalan (1987) was tested.

4.4.2 Evaluation Measure

The clustering heuristic for CFOPR selects the best solution based on a grouping efficacy value that is an evaluation measure employed in this dissertation. Thus, the test problems are evaluated by grouping efficacy in order to show the performance of the proposed heuristic. In this section, another evaluation measure known in this field is introduced as a supplemental measure to test the given problems. Grouping efficiency

No.	Size	Reference
1	5 × 7	King and Nakornchai (1982)
2	7 × 11	Kusiak and Chow (1987)
3	8 × 12	Seifoddini and Wolfe (1986)
4	8 × 20	Chandrasekharan and Rajagopalan (1986a)
5	8 × 20	Chandrasekharan and Rajagopalan (1986b)
6	10 × 15	Chan and Milner (1982)
7	14 × 24	Stanfel (1985)
8	16 × 30	Boctor (1991)
9	16 × 30	Boctor (1991)
10	16 × 30	Boctor (1991)
11	16 × 30	Boctor (1991)
12	16 × 30	Boctor (1991)
13	16 × 30	Boctor (1991)
14	16× 30	Boctor (1991)
15	16 × 43	Burbidge (1975)
16	20 × 35	Boe and Cheng (1991)
17	20 × 35	Carrie (1973)
18	24 × 40	Chandrasekharan and Rajagopalan (1989)
19	24 × 40	Chandrasekharan and Rajagopalan (1989)
20	24 × 40	Chandrasekharan and Rajagopalan (1989)
21	24 × 40	Chandrasekharan and Rajagopalan (1989)
22	24 × 40	Chandrasekharan and Rajagopalan (1989)
23	40 × 100	Chandrasekharan and Rajagopalan (1987)

Table 6. Test data sets

introduced by Chandrasekharan and Rajagopalan (1986) is employed as an alternative performance measure based on a quantitative criterion. It is defined by the weighted sum of two functions, $n1$ and $n2$, that are related to the number of ones in the clusters and zeros in the off-clusters as follows:

$$\text{Grouping efficiency} = q*n1 + (1-q)*n2$$

Where,

$$n1 = \frac{\text{Number of ones in the clusters}}{\text{Total number of elements in the clusters}}$$

$$n2 = \frac{\text{Number of zeros in the off-clusters}}{\text{Total number of elements in the off-clusters}}$$

q = A weighting factor having a value between zero and one

This measure has a scale of zero for the most ill-structured matrices to one for perfect-structured matrices, but even a very ill-structured matrix shows a grouping efficiency value of around 75%. Thus, in practice, the value of grouping efficiency ranges from about 75% to 100%. It means that grouping efficiency has a low discriminating power for ill-structured matrices. Then, grouping efficacy has higher priority to grouping efficiency in aspect of the performance evaluation for various-sized MPIMs with various difficulties. However, since grouping efficiency is also a notable evaluation measure in this field, the results of grouping efficiency for test problems is given in this dissertation, which can help readers to easily analyze the results obtained from the clustering heuristic for CFOPR. Thus, this dissertation provides the results obtained from measures such as the grouping efficacy as well as the grouping efficiency

in order to compare the proposed heuristic with other existing approaches for solving CFOPR. However, grouping efficiency values reported in this dissertation may not be the best because those values obtained from a heuristic are computed on machine-part clusters having the best grouping efficacy.

4.4.3 Computational Results

Seven alternative methodologies for solving CFOPR are selected in the literature in order to be compared with the proposed clustering heuristic. Seven approaches have been frequently employed to evaluate and be compared with new solving approaches in this field. These selected approaches are taken from non-hierarchical clustering, meta-heuristic and a statistical method called data mining as follows:

- ZODIAC (Chandrasekharan and Rajagopalan 1987)
- GRAFICS (Srinivasan and Narendran 1991)
- MST-GRAFICS (Srinivasan 1994)
- GA-TSP (Cheng *et al*. 1998)
- GA (Onwubolu and Mutingi 2001)
- GP-SLCA (Dimopoulos and Mort 2001)
- ARI (Chen 2003)

In order to present detailed results of the proposed heuristic, table 7 summarizes basic information such as the number of cells, operations, exceptional elements and voids and values of performance evaluation measures such as grouping efficiency and

grouping efficacy for test data sets. As mentioned, since the proposed heuristic depends on grouping efficacy, results shown in table 7 present solutions having the largest grouping efficacy. Thus, the provided grouping efficiency values may not be the best. In order to obtain the best grouping efficiency, the clustering heuristic should be modified by a simple change such that an evaluation measure, grouping efficacy, is exchanged with grouping efficiency. From this simple modification, any evaluation measure that is required to be adopted can easily be employed in the proposed clustering heuristic, which brings the flexibility to the proposed heuristic.

The proposed heuristic does not allow singleton grouping, but GP-SLCA, GA, and GA-TSP include singletons in their approaches. In the aspect of grouping efficacy, singletons make grouping performance increase. However, the practical CMS does not allow singleton grouping except for particular situations. Thus, grouping efficacy values for grouping including singletons are not presented in table 8 in order to fairly compare with other approaches. In order to compare the proposed heuristic with other approaches, this dissertation uses the results taken from previous papers and related literature. Table 8 presents the comparison of the proposed clustering heuristic with the existing seven methods in the aspect of grouping efficacy. A blank in table 8 represents that the related value is not available in the literature.

The result shows that the proposed clustering heuristic is superior to other existing approaches. Among results obtained by the proposed heuristic, the best solution is reported in 18 out of 23 data sets, and the second best solution is reported in the remaining five problems of which the grouping efficacy values are very close to those of

No.	# of cells	# of operations	# of exceptional elements	# of voids	Grouping efficiency	Grouping efficacy
1	2	16	2	3	0.8562	0.7368
2	3	23	6	9	0.7681	0.5313
3	3	35	7	6	0.8553	0.6829
4	3	61	9	0	0.9583	0.8525
5	2	91	27	18	0.7172	0.5872
6	3	46	0	4	0.9600	0.9200
7	5	61	9	14	0.8773	0.6933
8	4	106	9	53	0.8097	0.6101
9	4	92	8	28	0.8641	0.7000
10	6	111	45	25	0.8048	0.4853
11	4	107	11	25	0.8814	0.7273
12	5	101	10	17	0.9079	0.7712
13	3	114	11	64	0.7908	0.5787
14	4	118	12	19	0.9071	0.7737
15	6	126	31	43	0.8160	0.5621
16	4	153	35	57	0.8038	0.5619
17	4	136	2	41	0.8810	0.7571
18	7	131	0	0	1.0000	1.0000
19	7	130	10	11	0.9520	0.8511
20	7	131	20	20	0.9116	0.7351
21	9	131	47	31	0.8374	0.5185
22	10	131	58	31	0.8171	0.4506
23	10	420	36	37	0.9510	0.8403

Table 7. Performance of the proposed clustering heuristic

No.	Clustering Heuristic	ARI (2003)	GP-SLCA (2001)	GA (2001)	GA-TSP (1998)	MST-GRAFICS (1994)	GRAFICS (1991)	ZODIAC (1987)
1	0.737*						0.737	0.737
2	0.531*			0.5000	0.4688		0.531	0.391
3	0.683*						0.683	0.683
4	0.853*	0.852	0.852	0.853	0.852		0.852	0.852
5	0.587*		0.587	0.559	0.583		0.581	0.583
6	0.920*		0.920		0.920		0.920	0.920
7	0.693*			0.635	0.674		0.656	0.656
8	0.610*	0.571				0.508	0.534	0.586
9	0.700**	0.708	0.700			0.644	0.675	0.686
10	0.485*	0.478				0.407	0.449	0.267
11	0.727*	0.727	0.727			0.727	0.691	0.727
12	0.771*	0.766				0.760	0.771	0.764
13	0.579**	0.579	0.595			0.530	0.579	0.320
14	0.774*	0.774	0.774			0.774	0.774	0.774
15	0.562**	0.549	0.568		0.539	0.471	0.544	0.538
16	0.562*	0.527		0.444	0.551			0.511
17	0.757*	0.751		0.663	0.753		0.751	0.751
18	1.000*	1.000	1.000	1.000	1.000		1.000	1.000
19	0.851*	0.851	0.851		0.851		0.851	0.851
20	0.735*	0.735	0.735				0.735	0.730
21	0.519**	0.520		0.376	0.494	0.446	0.433	0.204
22	0.451*			0.348	0.447	0.439	0.445	0.182
23	0.840**	0.842	0.840	0.839	0.840		0.839	0.839

‘*’ indicates the best grouping efficacy in a set of test data.
‘**’ indicates the second best grouping efficacy in a set of data.

Table 8. Comparison with the cell formation methodologies

the best solution. In the comparison with ARI (Chen 2003), the latest approach for solving CFOPR, the proposed clustering heuristic has better results in 7 and the same results in 6 out of a total of 16 problems, but ARI is better for only 3 test problems. GP-SLCA (Dimopoulos and Mort 2001), which is based on a meta-heuristic, has better results in 2 test problems out of 12 problems, but it has a disadvantage of high computational complexity (Chen 2003). However, the proposed heuristic can obtain the solutions in a considerably shorter amount of time as shown in table 9. In table 8, other solving approaches except for ARI and GP-SLCA are apparently inferior to the proposed heuristic in the aspect of grouping efficacy. From the comparison with other solving approaches, this dissertation reports that the clustering heuristic for CFOPR can generate good quality solutions for well structured as well as ill-structured MPIMs.

4.4.4 Computational Complexity

In order to evaluate the computational complexity of the clustering heuristic for solving CFOPR, this research experiments how well the proposed algorithm is performed on the cell formation problems having various sizes and difficulty levels. First, the computational time, the CPU time, required to solve the test data sets given in table 6 is computed by running the proposed heuristic. The clustering heuristic for CFOPR is coded in MATLAB and run on a personal computer with a 1.6 GHz Intel Pentium M processor. Table 9 summarizes the number of machines, parts and operations and the CPU time for the given data sets. Most test problems required a computational time within 0.4 seconds of obtaining final solutions, which means that the clustering heuristic has an ability to obtain final clusters in a considerably shorter amount of time. Besides,

the proposed heuristic requires the computational time of around one and a half seconds even for the largest test problem. Thus, the proposed heuristic is very effective for solving large CFOPR problems.

Another method to evaluate the computational performance of an algorithm is to test various problems from small to large-sizes with different difficulty levels. Unfortunately, it is extremely difficult to obtain actual data for real manufacturing systems because most are proprietary. In order to overcome this limitation, the cell formation problems having five different sizes, from 10 machines and 10 parts to 50 machines and 50 parts, are randomly generated to evaluate the computational complexity of the proposed heuristic. Data sets to be tested depend on two difficulty levels, ill and well-structured. In this research, ill-structured and well-structured MPIMs are defined as data sets having grouping efficacy values of around 0.55 and grouping efficacy values of around 0.85, respectively.

Figures 10 and 11 illustrate the relationship of data sets with different sizes for two difficulty levels. The comparison of results in figures 10 and 11 reports that the computational complexity of the clustering heuristic for CFOPR does not depend on the difficulty levels of data sets. In other words, the results show that the proposed heuristic is relatively well implemented no matter what the difficulty of data sets is, ill or well-structured. As the number of machines and parts increase, the CPU time increases and the trend of the computational complexity increases exponentially. However, the clustering heuristic for solving CFOPR can obtain solutions around three seconds for the case of 50 machines and 50 parts. Although the computational complexity of heuristic

No.	# of machines	# of parts	# of operations	Computational time (sec.)
1	5	7	16	0.01
2	7	11	23	0.01
3	8	12	35	0.02
4	8	20	61	0.02
5	8	20	91	0.03
6	10	15	46	0.04
7	14	24	61	0.07
8	16	30	106	0.12
9	16	30	92	0.14
10	16	30	111	0.11
11	16	30	107	0.12
12	16	30	101	0.11
13	16	30	114	0.11
14	16	30	118	0.11
15	16	43	126	0.12
16	20	35	153	0.23
17	20	35	136	0.20
18	24	40	131	0.32
19	24	40	130	0.37
20	24	40	131	0.41
21	24	40	131	0.31
22	24	40	131	0.30
23	40	100	420	1.62

Table 9. Computational time of the clustering heuristic for CFOPR

increases exponentially, the computational time of the clustering heuristic is relatively short in comparison with other meta-heuristic methods.

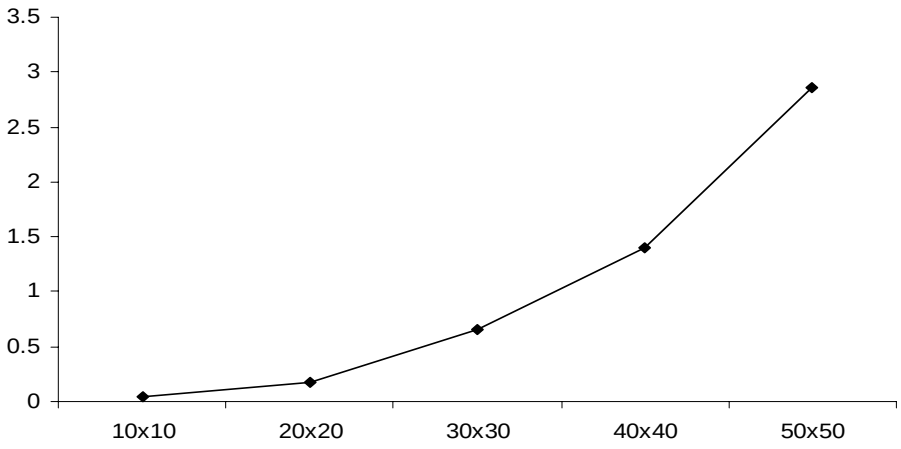

Figure 10. Computational complexity of the clustering heuristic for a well-structured MPIM

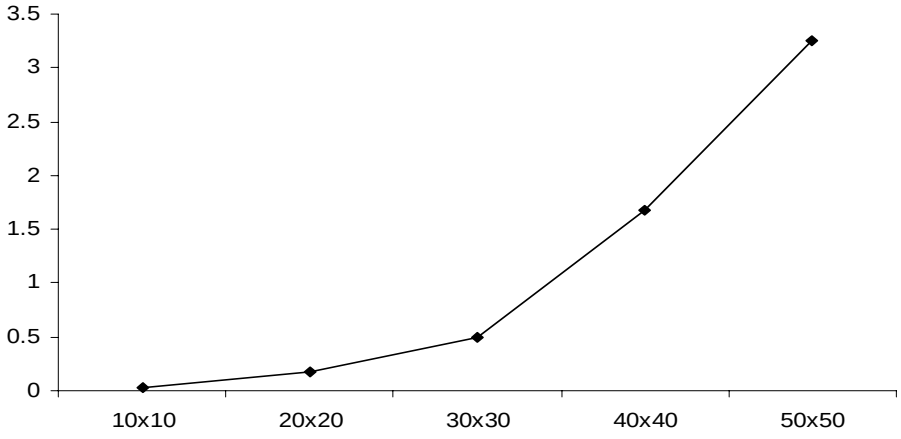

Figure 11. Computational complexity of the clustering heuristic for an ill-structured MPIM

4.5 Summary

In this chapter, the clustering heuristic for solving CFOPR in CMS is proposed. The clustering heuristic proposed in Chapter III is modified by adding a new feedback step and new specific allocation rules. This modification makes the proposed heuristic produce better solutions for CFOPR. Machine-part clusters obtained from the clustering heuristic described in Chapter III are fortified in a feedback step. Machine-part clusters are evaluated by grouping efficacy, and the solution having the largest grouping efficacy obtained through an iterative procedure is selected as the best.

The clustering heuristic for CFOPR is evaluated on cell formation problems, which range from 5 machines and 7 parts to 40 machines and 100 parts, taken from the literature. In order to test the performance of the proposed heuristic, seven well-known approaches are taken from non-hierarchical clustering, meta-heuristic and a statistical method called data mining. In the comparison with other approaches, the proposed heuristic produces the same or best solutions when compared with others in 18 out of 23 test problems and the second best solutions for all the remaining problems. In the aspect of computational complexity, the proposed heuristic obtains a solution in around one and a half seconds even for the largest test problem having 40 machines and 100 parts. Also, the computational complexity of the clustering heuristic does not depend on the difficulty level of data sets, but the number of machines and parts.

The performance evaluation shows that the clustering heuristic for CFOPR produces good solutions in a short amount of time, which means that the heuristic is very adequate to CFOPR.

CHAPTER V

APPLICATION FOR CELL FORMATION CONSIDERING MANUFACTURING FACTORS

5.1 Introduction

This chapter provides the algorithm for solving the cell formation problem considering a variety of manufacturing factors, CFVMF. In practice, manufacturing systems require much data to control and redesign production lines. In the CMS, cell formation is a basic step to estimate the goodness of the entire system. Most reported approaches have been concerned with CFOPR. Approaches based on a binary version, which deal with the processing requirement of parts, use the binary MPIM in order to obtain machine-part clusters.

However, Nair and Narendran (1999) discussed drawbacks of the binary MPIM. Wu and Salvendy (1993) argued that various manufacturing factors should be included in the design step of CMS. If manufacturing factors are not considered in a basic step, which is related to the creation of machine-part clusters, but in later steps, manufacturing systems may have improper and unfeasible production lines. Thus, a variety of manufacturing factors should be considered in the design step, called cell formation in CMS. However, due to the complexity of cell formation, it is unattainable to develop an approach considering all the manufacturing factors. A few researchers have proposed methods considering a part of various manufacturing factors in the configuration of the CMS, but not all of them.

This study considers various factors such as unit processing time, production volume, operation sequence, multiple identical machines, machine available time, lot size and setup time. In particular, a setup time issue, which is barely explored in existing studies, is discussed and involved in the solving procedure. The objective of this problem is to minimize intercellular movements traveling between machine-part clusters satisfying machine capacity. In order to solve CFVMF, five issues are considered as follows: i) Machine capacity, ii) Material flow, iii) Setup time, iv) Assignment of operations, and v) Operation splitting and the duplication of machines. The procedure for solving CFVMF consists of four sub-algorithms as follows:

SA1. Machine capacity management

SA2. Construction of similarity coefficient matrix

SA3. The clustering heuristic to obtain machine-part clusters

SA4. Evaluation of obtained solutions

The entire procedure of the clustering heuristic for solving CFVMF is based on the clustering heuristic described in Chapter III. The heuristic proposed here is developed by adding the machine capacity algorithm, which is a part to ensure the machining time required to process all the operations, to the original clustering heuristic. The clustering heuristic proposed to solve CFVMF is implemented by a new solving procedure introducing a duplicated machine-part matrix as a new matrix form. Also, a new similarity measure considering manufacturing factors is introduced to be used as the input data of the assignment algorithm. From the assignment algorithm, initial machine-

part clusters are generated, and through the iterative procedure of the clustering heuristic described in Chapter III, the resulting clusters obtained at all the iterations are compared by an evaluation measure, and the best result is kept.

5.2 Preliminary

5.2.1 Machine Capacity

In the configuration of manufacturing cells considering real production data such as production volume, processing time and machine available time, the proper management of machine capacity is required to ensure the time consumed to process operations required by all the parts. If the available time for any machine type is insufficient to process all the operations on the machine type, the manufacturing systems should supplement additional machines for any machine type having an insufficient capacity in order to make systems feasible. Thus, the replenishment of identical machines for that machine type can bring the reduction of the number of intercellular movements. However, the addition of identical machines for an insufficient machine type also entails extra cost. Thus, the proper number of identical machines should be considered in the process of the addition.

Heragu (1994) and Heragu and Gupta (1994) emphasized that machine capacity is the most important factor, and the number of each machine type should be first calculated before solving the cell formation problem. Wu and Salvendy (1999) mentioned that the duplication of any machine type should be allowed to ensure the machine capacity and multiple identical machines can be assigned into different clusters. They depicted the advantage of duplication for some machine types, which exceed the

machine capacity, in the solving procedure of the cell formation problems. Xambre and Vilarinho (2003) allowed operations to be split between multiple identical machines. They regarded split operations as different batches of the same part. From comments above, this research has three considerations for solving CFVMF.

- In order to ensure the machine capacity that is able to process operations required by all the parts, the number of machine types is first calculated before solving the cellular manufacturing problems.
- The duplication of machines and operations splitting are employed as tools to manage the machine capacity.
- Multiple identical machines are assigned into different clusters, and split operations are assigned into different identical machines

5.2.2 Operation Sequence

Another important manufacturing factor is operation sequence that can be defined as an ordering of the machines on which the part is sequentially processed (Vakharia and Wemmerlov 1990). In CMS, the operation sequence for all the parts is known as *a priori*. A binary MPIM represents the relationship between machines and parts based on the processing requirement that indicates only the existence of operations to be processed on machines. Since the sequences that parts are processed are not considered in a binary MPIM, material flow between all the pairs of machines for each part type are equal. In other words, the same amount of production volume for each part type is treated as material flow related to the part.

However, in practice parts sequentially move on machines to be processed and different part types are processed by different processing orders that indicate the sequence of the machine types required by part types. In the case considering the operation sequence for parts, since the material flow of an operation on the first or last machine in the sequence is transferred to a successive machine required by a part or from a previous machine respectively, these operations require only one movement. But operations traveling between machines require two movements, since the material flow moves from and to immediate machines. Based on the discussion above, an actual material flow should be calculated by considering operation sequence of parts, not processing requirement.

Gupta and Seifoddini (1990) emphasized the incorporation of operation sequence in the clustering decision process and developed the similarity coefficient involving operation sequence. Sarker and Xu (1998) surveyed the operation sequence-based similarity/dissimilarity coefficients and the cell formation methods based on operation sequence. In that study it was mentioned that operation sequence can affect the machine capacity and the material flow of parts, but considering only the machine capacity cannot influence the material flow.

5.2.3 Survey on the Machine-part Matrix

In the cell formation problems to attain the minimization of intercellular movements, many researchers employed matrix forms to calculate intercellular movements. The matrix form helps researchers to easily calculate the number of intercellular movements and readers to easily understand through the visualization of a

solving procedure. The existing MPIM is constructed by considering only processing requirement between machines and parts. Therefore, this type of matrix is not adequate to represent an actual intercellular flow affected by operation sequence.

Cheng et al. (1996) dealt with the assignment of multiple identical machines using the form of MPIM, but they did not consider operation sequence. Harhalakis et al. (1990) argued that the binary MPIM can not represent operation sequence and intercellular moves. They developed a component-machine incidence matrix represented by operation sequence instead of binary information, but this type of matrix cannot deal with production volume of parts. Wu (1998) developed a machine-machine relation matrix to overcome the drawback of the binary MPIM. Xambre and Vilarinho (2003) provided an interoperation flow matrix to consider the flow volume between operations. However, both the machine-machine relation matrix and the interoperation flow matrix do not represent actual intercellular flow of parts traveling between machine clusters. Won and Lee (2001) emphasized that operation sequence and production volume should be incorporated to calculate the intercellular part flow in CFVMF. They proposed a new production data-based part machine incidence matrix that can represent actual intercellular flow considering operation sequence and production volume, but they did not consider the machine capacity issue.

Table 10 summarizes studies on the matrix form proposed to calculate actual intercellular movements. Among the studies mentioned in table 10, there is no study considering all three factors such as operation sequence, production volume and multiple identical machines. Since these three factors directly affect the material flow of parts on

machines, it is necessary to combine and consider three factors into a matrix form in order to calculate the exact intercellular movements of parts.

In this research, the new version of a machine-part flow matrix, which is called the duplicated machine-part flow matrix (DMPFM), is constructed by incorporating manufacturing factors such as operation sequence, production volume and multiple identical machines. In DMPFM, rows and columns indicate duplicated machines and parts, respectively, and coefficients represent material flow of parts. Thus, coefficients shown in DMPFM represent the amount of actual material flow for movements of parts

Study	Matrix form	Consideration	Drawback
Harhalakis et al. (1990)	Component-machine incidence matrix	• Operation sequences • Multiple identical machines	Don't consider production volumes
Cheng et al. (1996)	Binary machine-part incidence matrix	• Multiple identical machines	Don't consider operation sequences
Wu (1998)	Machine-machine relation matrix	• Operation sequences • Production volumes • Multiple identical machines	Don't represent actual intercellular flow of parts
Won and Lee (2001)	Production data-based part machine incidence matrix	• Operation sequences • Production volumes	Don't consider multiple identical machines
Xambre and Vilarinho (2003)	Interoperation flow matrix	• Operation sequences • Production volumes • Multiple identical machines	Don't represent actual intercellular flow of parts

Table 10. Summary of studies on the matrix form

to and from multiple identical machines in each machine type. In the proposed clustering heuristic for solving CFVMF, DMPFM plays an important role in a procedure to ensure the machine capacity required by all the operations.

5.3 Machine Capacity Management

5.3.1 Duplicated Machine-part Flow Matrix

As mentioned, this research introduces a new machine-part flow matrix that represents actual flow of parts for duplicated machines. This matrix is expanded in the comparison with the existing binary machine-part matrix as follows:

- Coefficients in the proposed matrix represent actual flow of parts, not processing requirements.
- Duplicated machines for each machine type are represented.
- Operation sequences and production volumes are incorporated to calculate material flow of parts.

In the comparison with the existing matrix forms which are summarized in the section 5.2.3, DMPFM has advantages in the process for solving CFVMF as follows:

- Operation sequences, production volumes and multiple identical machines are incorporated into a matrix.
- Actual flow of parts for multiple identical machines in each machine type is represented.
- Assignment of operations to duplicated machines is described.

• The workload for multiple identical machines in each machine type is balanced, not the workload for machine types.

5.3.1.1 Material Flow of Parts

Material flow of parts used in the proposed matrix is based on operation sequence and production volume of parts. As discussed in the section 5.2.2, the material flow on a machine processing the first or last operation in the sequence moves only one time, and the operations processed on machines other than the first or last in the sequence require two movements of the material flow. The amount of actual material flow of parts traveling between machines can be calculated by considering the order in operation sequence and production volume.

• Index and definition

i	Machine
k	Parts
s	Order in operation sequence, $s = 1, \ldots, o_k$
o_k	the total number of machines required by part k in operation sequence
pv_k	Production volume of part k
f_{ik}	Material flow of part k on machine i
n_{iks}	Frequency of part trips in the operation sequence

Then, a material flow of part k on machine i is calculated as follows:

$$f_{ik} = n_{iks} * pv_k$$

where,

$$n_{iks} \begin{cases} 1, \text{ if } s = 1 \text{ or } o_k \\ 2, \text{ if } s = 2,\ldots, o_k - 1 \\ 0, \text{ otherwise} \end{cases}$$

5.3.1.2 Duplication of Machines

The duplication of machines occurs when the total time required to process operations assigned to any machine is greater than the available time of the machine. In this research, each machine type can be duplicated to ensure the machine capacity.

- Definition

p	The number of parts
nm_i	The required number of duplicated machine i
wt_i	Total working time assigned on machine i
cr_i	Capacity ratio of machine i
awt_{ik}	Working time to be assigned on machine i to process part k
at_i	Available time of machine i

The number of duplicated machines for machine type i is calculated as follows:

$$wt_i = \sum_{k=1}^{p} awt_{ik}$$

$$cr_i = wt_i / at_i$$

$$nm_i = \lceil cr_i \rfloor$$

where, $\lceil cr_i \rfloor$ indicates the nearest integer that is greater than cr_i .

For example, let the available time of any machine type be equal to 300 minutes and the total working time to process operations assigned to that machine type be equal to 450 minutes. Then, the number of duplicated machines in the machine type is equal to $nm = \lceil 450/300 \rfloor = 2$. That means that the machine type needs 2 identical machines in order to ensure the machining time required to process all operations.

5.3.1.3 Assignment of Operations to Duplicated Machines

Duplicated machines are assigned to different clusters and process the portion of workload allocated to the machine type. In the duplicated machine-part matrix, the workload allocated to machines is represented by data such as the total working time and the material flow assigned to parts processed on multiple identical machines in any machine type. In this section, machine-part matrix forms having two types of data sets are introduced. One is a duplicated machine-part time matrix (DMPTM) for a total assigned working time and another is DMPFM for a material flow. The total working time for a duplicated machine is calculated by the summation of machining times occurred by processing operations assigned to the duplicated machine. The material flow for a duplicated machine can be calculated by allocating production volume of the same pattern as total working times assigned to the duplicated machine.

In order to implement the allocation of workload, this algorithm employs the largest-first rule that an operation having the largest processing time is assigned to the first available machine. The first available machine means the machine having the shortest total assigned working time. The usage of the largest-first rule can minimize the largest out of the total workloads assigned to identical machines. Thus, the longest out of

machining times of multiple identical machines in any machine type can be minimized. The assignment of operations to duplicated machines is conducted by the following procedure:

• Definition

d	Index of a duplicated machine
upt_{ik}	Unit processing time for part k processed on machine i
pt_{ik}	Total processing time to process a production volume of part k on machine i
awt_i	Working time to be assigned on machine i
wt_i^d	Total working time assigned to a duplicated machine d in any machine type i
max_pt_i	The longest processing time among parts processed on machine type i
af_{ik}^d	Total material flow of part k assigned to a duplicated machine d in any machine type i

• Procedure

i) Calculation of the processing times required to process part k on machine type i

$$pt_{ik} = upt_{ik} * pv_k$$

ii) Selection of the longest processing time

$$max\_pt_i = arg\ max\{\ pt_{ik}\ \},\ \text{for all parts } k \text{ in a machine type } i$$

iii) Selection of a machine having the shortest assigned working time

$$arg\ min\{\ wt_i^d\ \},\ \text{for all duplicated machines } d \text{ in a machine type } i$$

iv) Assignment of the processing time of part k to the selected duplicated machine d in a machine type i

$$awt_i = max\_pt_i$$

$$wt_i^d = wt_i^d + awt_{ik}$$

v) Allocation of material flow corresponding to the assigned part *k* to the selected duplicated machine *d* in a machine type *i*

$$af_{ik}^d = af_{ik}^d + f_{ik}$$

For example, suppose that the number of duplicated machines in any machine type is 2, the assigned working times of duplicated machine 1 and 2 are 100 and 150 minutes, the processing times required to process the production volume of part 1 and 2 are 250 and 300 minutes, and the material flow of part 1 and 2 are 60 and 50, respectively. By the procedure explained above, *max_pt* is 300 minutes in part 2 and an identical machine having the shortest assigned working time is duplicated machine 1. Thus, part 2 is assigned to duplicated machine 1 and total working time of duplicated machine 1 is updated to 400 = 100 + 300. Also, the material flow, 60, of part 2 is added to the flow on duplicated machine 1.

5.3.2 Split of Operations

This section deals with the case that the machine capacity is not ensured after operations are assigned to duplicated machines by the procedure described in section 5.3.1.3. In this case, in order to completely process operations exceeding the capacity of the duplicated machines, the production volume of a part can be divided into lots called batches, which means that an operation processing the part is split into multiple identical machines. Xambre and Vilarinho (2003) mentioned that operation splitting is practical in

real industry, and the reduced-size production volumes by the operation splitting may bring flexibility and efficiency to manufacturing systems.

The next example provides the reason for the installation of operation splitting when operations are assigned to duplicated machines. Let the processing time of an operation to be assigned to a machine type be 200 minutes and the available machine time be 300 minutes. The machine type needs two duplicated machines to ensure the machine capacity, and the working times already assigned to the duplicated machine 1 and 2 are 200 and 150 minutes. By the procedure in the section 5.3.1.3, the operation should be processed to duplicated machine 2. However, the total working time of duplicated machine 2 after the operation is assigned is equal to 350 minutes which exceeds the available time, 300 minutes, of the machine type. If the split of operations is not considered, one extra duplicated machine should be purchased, which means additional cost.

Under an environment that operation splitting is allowed, the processing time assigned to duplicated machine 2 can be divided to ensure the machine capacity. In other words, the assigned operation is split. As a result, 150 minutes out of the processing time of the part is assigned to duplicated machine 2, and the remaining 50 minutes is assigned to duplicated machine 1. The resulting working times of duplicated machine 1 and 2 are 250 and 300 minutes respectively. That means that the machine capacity is satisfied. Thus, we don’t need to purchase an extra machine. The consideration of the operation splitting is inevitably to ensure the machine capacity when the total assigned working time exceeds the machine available time.

5.3.3 Setup Time Issue

As mentioned, this research introduces the setup time issue that affects the selection of duplicated machines to which operations are assigned when the required machining time exceeds the machine available time. This research defines a setup time as the preparation time required before processing a batch of any part type on a machine. So, a setup time is not considered as a manufacturing factor included in a unit processing time and having the same value. As studied in the previous sections, the selection of parts to be assigned to duplicated machines depends on only the amount of a time in a period consumed to completely process production volumes of the parts.

However, since the setup time is considered as an independent factor in this section, the total time consumed to process a part needs to be newly defined. The total consumed time mentioned in this research indicates the production time defined as the summation of processing time and setup time for a part processed on a machine. When setup times of parts are different according to part types, the assignment condition mentioned in the previous sections should be extended by introducing the setup time issue. Besides, when the setup time of a part is relatively big in comparison with the processing time of that part, the influence of setup time on the selection of duplicated machines gets bigger.

In the case employing setup time, we need to consider two cases. i) Total working time does not exceed the machine available time. ii) Total working time exceeds the machine available time. In the first case, parts are assigned to identical machines by the largest-first rule based on the production time, not the processing time

of parts on the machine. However, in the second case, the split of operations should be considered to ensure the machine capacity. The split of operations is conducted by allocating the proper portion of workload on the identical machine that exceeds the machine capacity to other identical machines that do not exceed the machine capacity. In the case considering the setup time, operations having the smallest setup time are split by the corresponding lot size. As already defined, the setup time occurs before processing parts on a machine. Thus, if an operation is split into two sub-operations, two setups will be required. In order to reduce the completion time that indicates the time consumed to finish all the parts processed on a machine type, the smallest setup time should be selected, because the processing time for all the parts is always constant. Thus, in the case exceeding the machine capacity, operations are assigned by the following procedure. Operations of parts having the non-smallest setup time are first assigned to duplicated machines by the largest-first rule based on production time. Then, an operation of a part having the smallest setup time is split by lot size in order to ensure the machine capacity.

In order to describe the assignment of parts to duplicated machines in the case considering the setup time, four cases stemmed from two cases, non-exceeding and exceeding the machine available time, are considered. Case 1 and 2 on a non-exceeding case, and case 3 and 4 on an exceeding case are based on table 11, which presents the data set for illustrating the assignment of operations when the setup time is considered.

- Case 1: The assignment of parts in the case considering only the processing time.

Figure 12 illustrates the case that operations related to parts are assigned to duplicated machines by the largest-first rule based on processing time. Part 3 having the

largest processing time and part 2 having the second largest processing time are assigned to duplicated machines 1 and 2 respectively. Then, since duplicated machine 1 has smaller total assigned working time than duplicated machine 2, part 1 is assigned to duplicated machine 1.

• Case 2: The assignment considering the production time in the case without the limit in the machine available time.

In the figure 13, part 1 having the longest production time is assigned to duplicated machine 1 and other parts, 2 and 3, are assigned to duplicated machine 2. Although part 1 has the shortest processing time, since the setup time of part 1 is relatively big in comparison with other parts, part 1 with the longest production time is assigned at first. This case shows the influence of the setup time in the allocation of parts to duplicated machines.

	Part 1	Part 2	Part 3
Unit processing time	0.8	1.0	1.2
Production volume	100	100	100
Setup time	90	50	20
Lot size	10	10	10
Total processing time	0.8 * 100 = 80	1.0 * 100 = 100	1.2 * 100 = 120
Production time	80 + 90 = 170	100 + 50 =150	120 + 2 = 140

Table 11. The data for illustrating cases considering a setup time

• Case 3: The assignment of the case selecting a part with the shortest setup time

By the allocation procedure of operations in the case exceeding the machine capacity, part 1 and 2 having the non-shortest setup time is assigned to duplicated machine 1 and 2, respectively. Then, in order to reduce the completion time, part 3 having the shortest setup time is split as shown in figure 14. Let the capacity of duplicated machines be 250 minutes. In figure 13, since the completion time of duplicated machine 2 is 290 minutes, we need to reduce the processing time of part 3 by 40 minutes. As mentioned, since operations are split by the lot size, a multiple of the value obtained by multiplying the unit processing time, 1.2 minutes, by the lot size, 10, of part 3 is subtracted from the completion time of duplicated machine 2 until machine

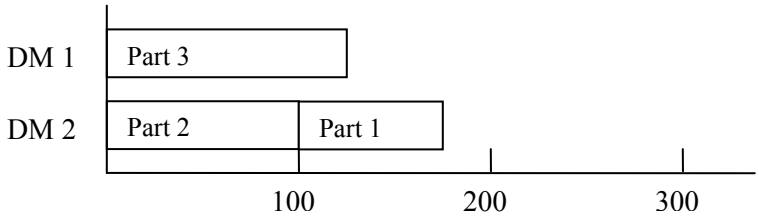

Figure 12. The completion time in the case considering only the processing time

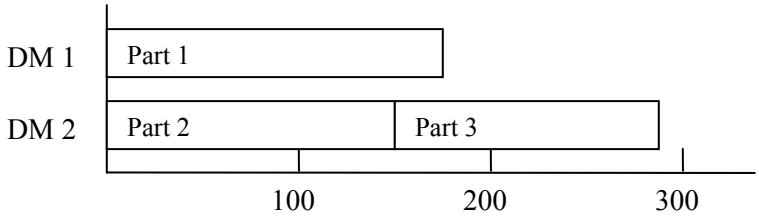

Figure 13. The completion time in the case considering the production time

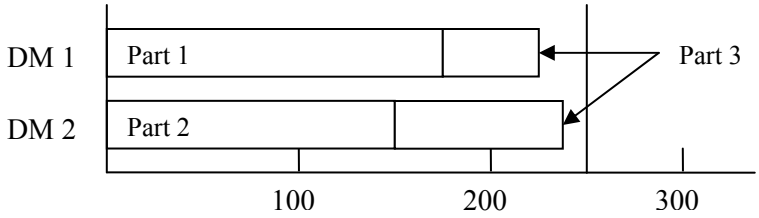

Figure 14. The completion time in the case with the smallest setup time

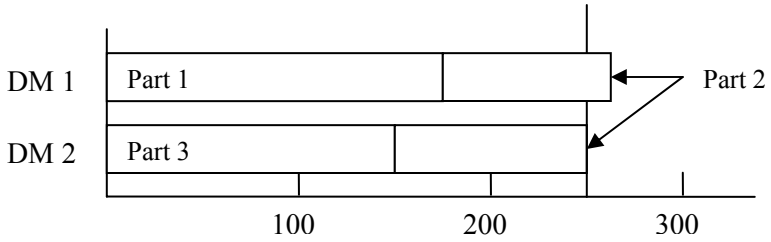

Figure 15. The completion time in the case with the non-smallest setup time

capacity is satisfied. In this example, 48 minutes is subtracted from 290 minutes and the setup time, 20 minutes, of part 3 and 48 minutes are added to the completion time, 170 minutes, of duplicated machine 1. As results, the completion times of duplicated machine 1 and 2 are equal to 238 and 242 minutes. Thus, the capacities of duplicated machine 1 and 2 are satisfied.

• Case 4: The assignment of the case selecting a part with the non-shortest setup time

Figure 15 illustrates the example to assign a part having the non-shortest setup time in the third case. The selection of part 2 as an operation to be split produces longer completion time than the example described in figure 14. Besides, this case shows that the machine capacity is not ensured in duplicated machine 1. Thus, when the total

working time exceeds the machine available time, the part having the shortest setup time should be selected as the operation to be split.

The amount of material flow of a part assigned to a duplicated machine depends on the portion of processing time assigned to the duplicated machine. In the case of part 3 in figure 14, since 40% of the total processing time is allocated to duplicated machine 1, 40% of production volume for part 3 should be assigned as a material flow on machine 1. Then, the amount of flow for part 3 on duplicated machine 1 and 2 are equal to 40 and 60, respectively. Also, the amount of material flow for part 1and 2 are equal to 100 in duplicated machine 1 and 2. Thus, total material flow allocated to duplicated machine 1 and 2 are equal to 140 and 160.

5.4 Similarity Measure Considering Manufacturing Factors

For solving CFVMF, the proposed similarity measure is developed by incorporating various manufacturing factors such as unit processing time, production volume and operation sequence. Since these data have non-binary values, we have to deal with the characteristics of manufacturing factors with attention. The proposed similarity measure is based on that of binary information, but modified to represent the characteristics of manufacturing factors. A new similarity measure has advantages as follows:

- Processing requirement, as well as manufacturing factors such as production volume, operation sequence and unit processing time, is incorporated.
- A high value is assigned to a similarity coefficient of machines having a high workload.

Thus, those machines having a high workload have a high priority in the clustering process.

- The material flow of parts as well as the workload of machines is considered.
- One similarity factor contributes only one similarity measure. Thus, the goodness of clustering can be evaluated by a similarity measure considering all factors as well as each similarity measure.
- Since the proposed measure has the product form of three similarity measures, the priority for each similarity measure can be defined by the weight form.

The proposed similarity measure consists of three similarities. Figure 16 illustrates the relationship of three similarity factors to develop a new similarity measure having characteristics mentioned above. These similarity factors include similarities considering processing requirement, material flow and machine workload which are important factors that have to be considered in the design step of manufacturing systems due to the following reasons:

- Processing requirement indicates whether a part is processed on any machine or not. This factor is basic information in a manufacturing system design because the cell formation is to form machine-part clusters based on the relationship between parts processed on machines and the machines.
- The common objective in the configuration of manufacturing cells is to minimize intercellular movements. Material flow, which means flow of parts between machines, should be known to compute the number of intercellular movements.

• In CFVMF, machine capacity is a factor that is considered at first. In this research, after the capacities of machines are ensured, machine-part clusters are formed. Machine workload represents the degrees of jobs allocated to machines. In order to ensure machine capacities, workload on machines should be calculated.

Three similarity factors are calculated by considering processing requirement, operation sequence, unit processing time and production volume. These manufacturing factors, which are the most common data generated from manufacturing systems, are usually used to analyze and control manufacturing systems. Thus, a new similarity measure should incorporate the three similarity factors mentioned above in order to indicate influences of these manufacturing factors.

5.4.1 Similarity Considering Processing Requirement

The first similarity to consider is based on processing requirements between parts

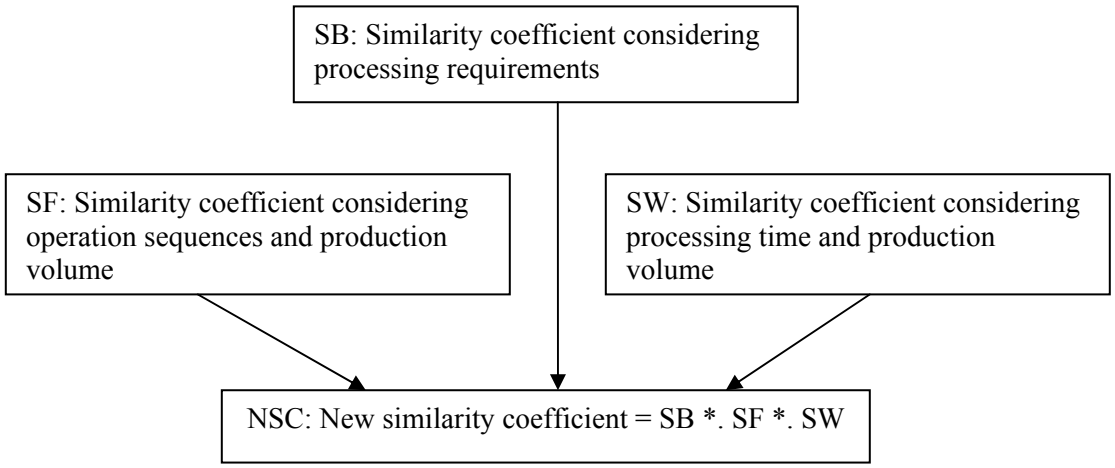

Figure 16. The structure of a new similarity coefficient

and machines. This similarity is the same as the similarity measure considering only binary information. The definition of a similarity coefficient required in this section is equivalent to the similarity measure referred in the original clustering heuristic in Chapter III.

5.4.2 Similarity Considering Material Flow

This similarity is based on DMPFM representing material flow between machines and parts obtained by considering operation sequence and production volume. In order to reduce intercellular movements of parts, machines to process a part having a large production volume should be grouped into the same cluster. Gupta and Seifoddini (1990) depicted the influence of production volume in the grouping process. It was mentioned that a production volume out of three manufacturing factors such as production volume, operation sequence and unit processing time has a dominant power.

The similarity coefficient considering material flow is defined as the ratio of the sum of flow of parts having non-zero flow on both machines to the sum of flow of parts having non-zero flow on both or either of two machines. The proposed similarity coefficient has a value ranging from 0 to 1. A large similarity coefficient indicates that parts having large production volumes are processed on both machines. Thus, a machine pair having a larger flow for parts tends to be grouped by the proposed similarity coefficient. In order to develop a similarity measure considering material flow of parts between machines, the similarity measure defined in the clustering heuristic is modified. In the similarity for material flow, since the ‘0-0’ match in the machine-part flow matrix

means that there is no flow for any part on both machine, we do not need to consider '0-0' matching information. Similarity coefficient can be defined as follows:

- Notation

sf_{ij}	Similarity coefficient based on the material flow between machine i and j
f_{ik}	Flow for part k on machine i

- Binary variables

x_i^k	1, $f_{ik} > 0$ 0, otherwise
u_{ij}^k	1, $x_i^k = x_j^k = 1$ 0, otherwise
v_{ij}^k	1, $x_i^k = 1$ or $x_j^k = 1$ 0, otherwise

Thus, the new similarity coefficient based on material flow is defined as follows:

$$sf_{ij} = \frac{\sum_{k=1}^{p} \left(f_{ik} + f_{jk}\right) * u_{ij}^k}{\sum_{k=1}^{p} \left(f_{ik} + f_{jk}\right) * v_{ij}^k}$$

5.4.3 Similarity Considering Workload

A similarity measure for workload is related to a total processing time consumed to process production volumes of parts on each machine. In order to describe a total processing time, let the unit processing time for part k on machine i be 2 minutes and the

production volume for part *k* be 100. Then, a total processing time on machine *i* for part *k* is 200 minutes. This similarity measure is defined as the ratio of the total processing time for parts on both machines to the total processing time for parts on both or either of the machines. By the definition of the similarity coefficient considering the workload assigned to a pair of machines, a machine pair having a larger workload, a longer total processing time, for parts tends to be grouped into the same machine cluster. In this similarity, '0-0' matches do not need to consider because '0-0'matches means that there is no operation for any part on a machine. The similarity measure considering the workload of machines is similar to the form of the similarity considering the material flow of parts and can be defined as follows:

- Notation

  sw_{ij} Similarity coefficient based on the workload between machine *i* and *j*

  pt_{ik} Total processing time for part *k* on machine *i*

- Binary variable

$$x_i^k \quad \begin{cases} 1, & pt_{ik} > 0 \\ 0, & \text{otherwise} \end{cases}$$

$$u_{ij}^k \quad \begin{cases} 1, & x_i^k = x_j^k = 1 \\ 0, & \text{otherwise} \end{cases}$$

$$v_{ij}^k \quad \begin{cases} 1, & x_i^k = 1 \text{ or } x_j^k = 1 \\ 0, & \text{otherwise} \end{cases}$$

Thus, the new similarity coefficient based on machine workload is defined as follows:

$$sw_{ij} = \frac{\sum_{k=1}^{p} \left(pt_{ik} + pt_{jk}\right) * u_{ij}^{k}}{\sum_{k=1}^{p} \left(pt_{ik} + pt_{jk}\right) * v_{ij}^{k}}$$

5.4.4 Similarity Measure Considering Manufacturing Factors

The resulting similarity measure consists of three parts such as similarity coefficients considering processing requirement, material flow and machine workload. The three similarity factors described in the previous sections are combined by the product form as follows:

$$nsc_{ij} = sb_{ij} * sf_{ij} * sw_{ij}$$

where, nsc_{ij} is a new similarity coefficient considering manufacturing factors, sb_{ij} is a similarity coefficient related to processing requirement, sf_{ij} is a similarity coefficient related to material flow and sw_{ij} is a similarity coefficient related to workload between machine *i* and *j*.

5.5 Clustering Heuristic for Solving CFVMF

5.5.1 Introduction

The solving procedure based on the clustering heuristic described in CHAPTER III is proposed to solve CFVMF. Data such as operation sequence, processing time, production volume, setup time, machine available time, lot size and frequency of part trips in operation sequence should be collected before the design step. Unit processing time and production volume influence the workload of machines and total working time.

Operation sequence, frequency of part trips and production volume are used to calculate actual flow of parts on machines. In order to ensure the machine capacity, manufacturing factors as machine available time, unit processing time, production volume, setup time, and lot size are considered. The clustering heuristic proposed in Chapter III is employed to obtain final machine-part clusters.

The entire procedure of the algorithm for solving CFVMF is similar to the heuristic proposed to solve CFOPR. However, the procedure of heuristic for the second application is more complicated, because a variety of manufacturing factors used in real industry are considered in the solving procedure. The objective of this problem is to minimize intercellular movements of parts traveling between machine-part clusters satisfying the machine capacity. The proposed heuristic first ensures the capacities of machines and then generates the machine-part cluster.

The proposed algorithm is based on two new matrices such as the DMPFM representing the material flow and the DMPTM representing the processing time. In the step of the machine capacity management, machines are duplicated and operations are split in order to ensure the capacity of machines, which is attempted through the manipulation of DMPTM. Then, the minimization of intercellular movements is conducted through DMPFM. Data sets produced from the implementation of the algorithm are stored and updated in the two duplicated matrices. After implementation of the heuristic for CFVMF, duplicated machine-part clusters satisfying capacities for duplicated machines are obtained, and actual material flow of parts on duplicated

machines are calculated. The clustering heuristic proposed to solve CFVMF focuses on considerations as follows:

- Construction of duplicated machine-part matrix
- Assignment of multiple identical machines to different clusters
- Split of operations and duplication of machines in order to ensure the machine capacity
- Consideration of setup time
- Proposal of new similarity coefficient incorporating various manufacturing factors
- Minimization of intercellular movements between machine-part clusters
- Calculation of actual material flow of parts on duplicated machines
- Creation of duplicated machine-part clusters

5.5.2 Summary on Heuristic

The procedure of the clustering heuristic for solving CFVMF consists of four sub-algorithms as follows:

SA1. Machine capacity management:

In order to meet the machine capacity, various schemes such as the duplication of machines, the split of operations and the largest-first rule are employed. In particular, two kinds of new matrices are proposed. The DMPTM deals with the machine working time, and the DMPFM represents material flow of parts on duplicated machines. After the capacities for all the machines are satisfied, material flow on machines are allocated

to the corresponding machine types or duplicated machines in any machine type according to the pattern of workload assigned to machines.

SA2. Similarity coefficient matrix:

A similarity measure in CFVMF should incorporate various manufacturing factors. The proposed similarity coefficient measure consists of three parts. The first similarity is related to processing requirements between machines and parts represented by DMPIM. The second similarity is based on DMPFM representing flow between machines and parts obtained by considering operation sequence and production volume. The third similarity, which considers processing times and production volumes, is related to the ratio of workloads between two machines. The third measure can be calculated by DMPTM. Three similarity measures mentioned above are incorporated by multiplying them. This new similarity coefficient matrix is used as input data of the clustering heuristic in the next step. The similarity coefficient matrix has the form of a symmetric square matrix.

SA3. Clustering heuristic:

Originally, the objective of this research was to develop an efficient heuristic based on cluster analysis to configure manufacturing cells. The clustering heuristic proposed in Chapter III is applied to CFVMF. In the procedure of the proposed clustering heuristic, the assignment algorithm produces initial machine clusters, and the ALC method (Seifoddini and Wolfe 1986 and 1987) is employed to merge initial machine clusters. However, since this case deals with the non-binary data set, the ALC

method is not adequate to represent the characteristic of manufacturing factors. Thus, in the heuristic for solving CFVMF, a new grouping similarity coefficient is introduced. In this step, the similarity coefficient used to calculate a group similarity value is equivalent to the similarity measure, sf_{ij}, considering material flow in DMPFM.

After machine clusters are obtained, parts should be allocated to the proper machine clusters to complete the configuration of manufacturing cells. Since the objective of the second application is to minimize intercellular movements between clusters, parts should be also allocated to minimize intercellular movements, which is the same concept as the maximization of within-cluster flows. Thus, the maximum density rule is used to grantee the minimization of intercellular movements. When a tie exists, the proposed algorithm considers the relationship of the number of operations and the workload of machines within a cluster.

SA4. Evaluation of obtained solutions:

Solutions obtained from the proposed heuristic are evaluated in the aspect of the amount of intercellular flow between machine-part clusters. The solution having the smallest intercellular movements becomes the best in comparison with others. However, if we consider only the minimization of intercellular movements, only one cluster including all machine types and parts will be the optimal solution, because one cluster produces zero intercellular movement. But in the aspect of the goodness of grouping, the solution having only one cluster is a very poor solution as well as a non-practical situation in industry. Thus, we need to consider the goodness of grouping for the obtained machine-part clusters.

In Chapter IV, grouping efficacy is employed to evaluate solutions in CFOPR. This evaluation measure is also used to select the best solution in CFVMF. However, the trade-off between the number of intercellular movements and the goodness of grouping may be required by employing grouping efficacy. Thus, we need to make a decision on the preference for the selection of the best solution and the limit of the number of clusters. This research chooses no preference, which means that the selection of preferences is reserved for readers. So readers can get the duplicated machine-part clusters based on the preference they want. Also, readers can employ various evaluation measures instead of the measures that this dissertation adopts, which brings flexibility to the proposed clustering heuristic. When grouping efficacy is considered, the machine-part clusters obtained at any iteration are evaluated and compared with the best solution at the previous iteration. A solution having the highest grouping efficacy is considered as the best solution, and the number of intercellular movements in the best solution is calculated and kept.

5.5.3 Procedure of Heuristic

In this section, the entire procedure of the clustering heuristic proposed to solve CFVMF is described. The proposed heuristic consists of four sub-algorithms that are divided into six steps. For the convenience of the description of the proposed heuristic, the first sub-algorithm, machine capacity management, is divided into three steps, and the last two sub-algorithms, the clustering heuristic and evaluation, are merged into step 6. The procedure of the clustering heuristic for CFVMF can be described as follows:

• Objective: the minimization of intercellular movements traveling between duplicated machine-part clusters satisfying the capacities of duplicated machines

• Known data: Operation sequence, Processing time, Production volume, Setup time, Machine available time, Lot size, Frequency of part trip in operation sequence

• Index and notation

i and j	Machine
k	Part
s	Order in operation sequence, $s = 1, \ldots, o_k$
d	Duplicated machine
u and v	Machine cluster
N_u	Total number of machines in machine cluster u
p	Total number of parts
o_k	Total number of part k in operation sequence
ft_{ik}	Frequency of trips of part k visiting machine i
n_{iks}	Frequency of part trips in the operation sequence
pv_k	Production volume of part k
f_{ik}	Material flow of part k on machine i
nm_i	The required number of duplicated machine i
cr_i	Capacity ratio of machine i
at_i	Available time of machine i
upt_{ik}	Unit processing time for part k processed on machine i

pt_{ik} Total processing time to process a production volume of part k on machine i

awt_{ik} Working time to be assigned on machine i to process part k (= pt_{ik})

wt_i Working time to be assigned on machine i

dwt_i^d Working time already assigned to a duplicated machine d in any machine type i

max_pt_i The longest processing time among parts $\in$ PA_i

af_{ik}^d Total material flow of part k assigned to a duplicated machine d in any machine type i

sb_{ij} Similarity coefficient between machine i and j based on production requirement after the double centering

sf_{ij} Similarity coefficient between machine i and j based on material flow

sw_{ij} Similarity coefficient between machine i and j based on workload

dfm_{ik} Coefficient of part k on machine i in the duplicated machine-part flow matrix

dim_{ik} Coefficient of part k on machine i in the duplicated machine-part incidence matrix

st_{ik} Setup time of part k on machine i

gs_{uv} Group similarity coefficient for two machine clusters u and v

min_st_i The shortest setup time among parts $\in$ PA_i

- Set and matrix

PA_i Set of parts to be processed on machine type i

OC Set of machines that exceed the capacity of that machine

DMPIM Duplicated machine-part incidence matrix

DMPTM Duplicated machine-part time matrix

DMPFM Duplicated machine-part flow matrix

MPIM Machine-part incidence matrix

NSM New similarity coefficient matrix

• Procedure

Step 1. Machine Duplication

1.1 Calculation of a working time to process part k on machine i

$$awt_{ik} = pv_k * upt_{ik} * ft_{ik} + st_{ik}$$

1.2 The capacity ratio for each machine type i

$$wt_i = \sum_{k=1}^{p} awt_{ik}$$

$$cr_i = wt_i / at_i$$

1.3 The number of duplicated machines for a machine type i

$$nm_i = \lceil cr_i \rfloor$$

where, $\lceil cr_i \rfloor$ indicates the nearest integer that is greater than cr_i .

Step 2. Duplicated machine-part time matrix

2.1 Allocation of parts to identical machines by the Largest-first rule for each machine type

2.1.1 Calculation of the total processing time required to process part k on machine type i

$pt_{ik} = upt_{ik} * pv_k * ft_{ik} + st_{ik}$, for a part $k \in \mathrm{PA}_i$

2.1.2 Selection of the longest total processing time

$$max\_pt_i = arg\ max\{\ pt_{ik}\ \},\ \text{for all parts } k \in PA_i$$

2.1.3 Selection of a duplicated machine having the shortest assigned working time

The selection of duplicated machine $d \leftarrow arg\ min\{\ dwt_i^d\ \}$, for all duplicated machines in a machine type i

2.1.4 Assignment of the processing time of operations to the selected duplicated machine d in any machine type i

$$wt_i = max\_pt_i$$

$$dwt_i^d{}_{(n)} = dwt_i^d{}_{(n-1)} + wt_i$$

where, the initial value of dwt_i^d is equal to zero.

2.1.5 If all parts in a set PA_i are assigned, go to step 2.2. Otherwise, remove the assigned part from a set PA_i and repeat 2.1.1 to 2.1.4

2.2 If all machine types are considered, go to step 3.1. Otherwise, continue step 2.1

Step 3. Duplicated machine-part flow matrix

3.1 Actual parts flow

3.1.1 Actual frequency of parts flow

$$n_{iks} \begin{cases} 1, & \text{if } s = 1 \text{ or } o_k \\ 2, & \text{if } s = 2,\ldots, o_k - 1 \\ 0, & \text{otherwise} \end{cases}$$

3.1.2 Actual flow on machine i for part $k \in PA_i$

$$f_{ik} = n_{iks} * pv_k$$

If multiple trips occur in the sequence, the sum of n_{iks} for each trip is considered.

3.2 Assignment of an actual part flow to duplicated machine d by the same pattern as

the assignment of processing times in DMPTM

$$af_{ik}^{d}{}_{(n)} = af_{ik}^{d}{}_{(n-1)} + f_{ik}$$

where, the initial value of af_{ik}^{d} is equal to zero.

3.3 If all parts in a set PA_i and machine types are considered, go to step 4. Otherwise, repeat step 3.1 to 3.2

Step 4. Machine balancing (when working time to be assigned on machine *i* exceeds the capacity of that machine)

4.1 Selection of an identical machine to which lot size of a part is allocated

4.1.1 Selection of machine types that exceed the machine available time

$$wt_i > at_i \rightarrow \text{Machine type } i \in \text{OC, for all machine types}$$

4.1.2 Selection of a part with the shortest setup time

$$min\_\ st_i = \arg\min \{ st_{ik} \} \text{ for part } k \in PA_i$$

4.1.3 Selection of an identical machine with the shortest assigned working time

The selection of identical machine $d \leftarrow arg\ min\{ dwt_i^d \}$, for all duplicated machines in any machine type *i*

4.2 Assignment of the corresponding values, flow and time, to the selected identical machine

4.2.1 Allocation of a lot size for a part having the shortest setup time to the selected duplicated machine *d* in the DMPFM

4.2.2 Allocation of the sum of the setup time and the machine working time required to process lot size of the part having the shortest setup time to the selected identical machine *d* in DMPTM

4.2.3 If the assigned working time of the selected identical machine still exceeds the machine available time, repeat step 4.2. Otherwise, go to step 4. 3

4.3 If all machine types are considered, go to step 5. Otherwise, repeat step 4

Step 5. New similarity coefficient matrix

5.1 Similarity coefficient based on processing requirement

5.1.1 Construct a DMPIM considering duplicated machines

$$dim_{ik} \begin{cases} 1, & \text{if } dfm_{ik} > 0 \\ 0, & \text{otherwise} \end{cases}$$

5.1.2 Calculate a similarity coefficient between duplicated machines *i* and *j* based on a DMPIM

$$s_{ij} = \frac{\sum_{k=1}^{p} u_{ij}^{k} + \sum_{k=1}^{p} w_{ij}^{k}}{\sum_{k=1}^{p} v_{ij}^{k}}$$

5.1.3 Apply the double centering method

$$sb_{ij} = s_{ij} - s_{i\cdot} - s_{\cdot j} + s_{\cdot\cdot}$$

Where, s_{ij} is a similarity coefficient between duplicated machines *i* and *j*, $s_{i\cdot}$ is an average for row *i* in the similarity coefficient matrix, $s_{\cdot j}$ is an average for column *j* in the similarity coefficient matrix, $s_{\cdot\cdot}$ is an average for all the coefficients in the similarity coefficient matrix

5.2 Material flow ratio between duplicated machines *i* and *j*

$$sf_{ij} = \frac{\sum_{k=1}^{p} \left(f_{ik} + f_{jk}\right) * u_{ij}^{k}}{\sum_{k=1}^{p} \left(f_{ik} + f_{jk}\right) * v_{ij}^{k}}$$

5.3 Machine workload ratio between duplicated machines *i* and *j*

$$sw_{ij} = \frac{\sum_{k=1}^{p} \left(pt_{ik} + pt_{jk}\right) * u_{ij}^{k}}{\sum_{k=1}^{p} \left(pt_{ik} + pt_{jk}\right) * v_{ij}^{k}}$$

5.4 New similarity coefficient between duplicated machines *i* and *j*

$$nsc_{ij} = sb_{ij} * sf_{ij} * sw_{ij}$$

5.5 If all duplicated machine pairs are considered, go to step 6. Otherwise, repeat step 5.1 to 5.4

Step 6. Clustering heuristic

6.1 Assignment algorithm for initial machine clusters

6.2 Allocation of parts to machine clusters

6.2.1 Maximum density rule

6.2.1.1 Calculate total material flow for each of the merged clusters

6.2.1.2 Allocate parts to machine clusters with the maximum flow

6.2.1.3 If tie exists, go to step 6.2.2 Otherwise, go to step 6.2.4

6.2.2 Largest number of operations

6.2.2.1 Calculate the number of operations for the merged clusters

6.2.2.2 Allocate parts to clusters with the largest number of operations

6.2.2.3 If tie exists, go to step 6.2.3 Otherwise, go to step 6.2.4

6.2.3 Largest ratio for machine workload

6.2.3.1 Calculate the ratio as follows

$$\frac{\text{The number of operations within a cluster}}{\text{The number of machines within a cluster}}$$

6.2.3.2 Allocate parts to clusters with the largest ratio

6.2.2.3 If tie exists, break arbitrarily Otherwise, go to step 6.2.4

6.2.4 If the number of machine clusters is not equal to the number of part families and a singleton exists, go to step 6.4. Otherwise, go to step 6.3

6.3 Evaluation

6.3.1 Calculate the grouping efficacy

6.3.2 Calculate intercellular movements for the merged clusters

6.3.3 If clusters to be merged exist, go to step 6.4. Otherwise, keep a solution having the highest grouping efficacy as the best, and terminate an algorithm.

6.4 Merge initial clusters using average group similarity

6.4.1 Calculate the group similarity coefficient

$$gs_{uv} = \frac{\sum_{i \in u} \sum_{j \in v} sf_{ij}}{N_u \times N_v}$$

6.4.2 Merge initial clusters with the largest average group similarity coefficient. If a singleton exists, first merge the singleton to group with the largest average group similarity coefficient.

6.4.3 Go to step 6.2

5.6 Performance Evaluation

5.6.1 Illustration of the Proposed Heuristic

Before providing the performance of the heuristic proposed to solve CFVMF, a simple example is illustrated to show the solving procedure of the heuristic. Table 12 shows the sample data to illustrate an example having 6 parts to be processed on 4 machine types. The available time for every machine type is 250. The unit related to time data is a minute. The procedure of the clustering heuristic proposed to solve CFVMF is described as follows.

Example procedure of the heuristic for CFVMF

Step 1. Calculate the number of duplicated machines of machine types. 1 for machine type 1, 2 for machine type 2, 3 and 4.

Part no.	Operation sequences	Processing times	Setup times	Production volumes	Lot sizes
1	m1-m4-m1	.5-.8-.5	10-14-10	100	10
2	m2-m3	.7-.8	15-10	80	10
3	m4-m2	.7-.9	18-9	120	10
4	m2-m3-m2	.6-.7-.6	15-20-15	90	10
5	m4-m3-m2-m3	.9-.6-.7-.6	15-18-14-18	70	10
6	m2-m1-m4	.8-.7-.9	18-15-20	80	10

Table 12. The data for a simple example to illustrate the heuristic

Step 2. Construct DMPTM by assigning workloads to duplicated machines following the largest-first rule. Table 13 represents DMPTM in step 2.

Step 3. Construct DMPFM by assigning material flow of the same pattern as workloads in the DMPTM. Table 14 shows DMPFM in step 3.

Step 4. Ensure the capacity of all machine types. The duplicated machine 1 in the machine type 2 exceeds the capacity. Then, a part 5 with the smallest setup time is selected to change the workload and the material flow. Tables 15 and 16 show

	P1	P2	P3	P4	P5	P6
M1	110	0	0	0	0	71
M2(d1)	0	71	0	123	63	0
M2(d2)	0	0	117	0	0	82
M3(d1)	0	0	0	0	102	0
M3(d2)	0	74	0	83	0	0
M4(d1)	0	0	102	0	78	0
M4(d2)	94	0	0	0	0	92

Table 13. The duplicated machine-part time matrix for an example

	P1	P2	P3	P4	P5	P6
M1	200	0	0	0	0	160
M2(d1)	0	80	0	180	140	0
M2(d2)	0	0	120	0	0	80
M3(d1)	0	0	0	0	210	0
M3(d2)	0	80	0	180	0	0
M4(d1)	0	0	120	0	70	0
M4(d2)	200	0	0	0	0	80

Table 14. The duplicated machine-part flow matrix for an example

DMPTM and DMPFM after ensuring the machine capacity.

Step 5. Construct new similarity coefficient matrix based on three factors such as process requirement, material flow and machine workload. Table 17 represents the resulting similarity matrix.

Step 6. Implement the proposed clustering heuristic to obtain initial machine-part clusters. Based on the similarity coefficient matrix in table 17, initial machine clusters are produced as follows:

	P1	P2	P3	P4	P5	P6
M1	110	0	0	0	0	71
M2(d1)	0	71	0	123	56	0
M2(d2)	0	0	117	0	21	82
M3(d1)	0	0	0	0	102	0
M3(d2)	0	74	0	83	0	0
M4(d1)	0	0	102	0	78	0
M4(d2)	94	0	0	0	0	92

Table 15. The duplicated machine-part time matrix after balancing

	P1	P2	P3	P4	P5	P6
M1	200	0	0	0	0	160
M2(d1)	0	80	0	180	130	0
M2(d2)	0	0	120	0	10	80
M3(d1)	0	0	0	0	210	0
M3(d2)	0	80	0	180	0	0
M4(d1)	0	0	120	0	70	0
M4(d2)	200	0	0	0	0	80

Table 16. The duplicated machine-part flow matrix after balancing

MC1 = {M1, M4(d2)}

MC2 = {M2(d1), M3(d2)}

MC3 = {M2(d2)}

MC4 = {M3(d1), M4(d1)}

Since a singleton cluster exists, the initial clusters are merged by the average group similarity as follows:

MC1 = {M1, M4(d2)}

MC2 = {M2(d1), M3(d2)}

MC3 = {M2(d2), M3(d1), M4(d1)}

By the maximum density rule, the assignment of part 1 and 6 to machine cluster 1, part 2 and 4 to machine cluster 2 and part 3 and 5 to machine cluster 3 is obtained. Based on the clusters above, grouping efficacy is 0.8125, and the number of intercellular movements is equal to 210. Since a grouping efficacy

	M1	M2(d1)	M2(d2)	M3(d1)	M3(d2)	M4(d1)	M4(d2)
M1	0	0	-0.0989	0	0	0	7.0760
M2(d1)	0	0	-0.0075	0.1096	1.7993	-0.0464	0
M2(d2)	-0.0989	-0.0075	0	0.0713	0	1.0867	-0.0861
M3(d1)	0	0.1096	0.0713	0	0	1.8728	0
M3(d2)	0	1.7993	0	0	0	0	0
M4(d1)	0	-0.0464	1.0867	1.8728	0	0	0
M4(d2)	7.0760	0	-0.0861	0	0	0	0

Table 17. The similarity coefficient matrix for an example

value is the highest, the obtained clusters are considered as the best solution that has 210 intercellular movements. Table 18 represents DMPFM showing the material flow of the best solution for an illustrative example.

In the illustrative example above, we obtain the duplicated machine-part clusters with the best grouping efficacy, 0.8125, and 210 intercellular movements. In the aspect of the grouping efficacy, the resulting clusters should be the best solution. However, in the aspect of the number of intercellular movements, the alternative solution with smaller intercellular movements may exist as shown in the table 19. The alternative solution for the same example has better intercellular movements, 160, but the poor grouping efficacy value, 0.6667. Thus, as mentioned, the trade-off between two evaluation measures is required to decide the final solution.

5.6.2 Performance Evaluation

The clustering heuristic proposed to solve CFVMF is evaluated through the comparison with other approaches for the test problems taken from the literature. However, there is no existing example that is an exact fit for the proposed cell formation problem in the literature, because studies including all the manufacturing factors considered in this research do not exist in the existing literature. Therefore, test examples from existing studies with the objective of the minimization of intercellular movements are selected to evaluate the proposed heuristic. The cell formation problems taken from the literature do not consider setup time, but similar manufacturing factors and the same objective. In order to evaluate the performance of the proposed heuristic, after setup time

	P1	P6	P2	P4	P3	P5
M1	200	160	0	0	0	0
M4(d2)	200	80	0	0	0	0
M2(d1)	0	0	80	180	0	130
M3(d2)	0	0	80	180	0	0
M2(d2)	0	80	0	0	120	10
M3(d1)	0	0	0	0	0	210
M4(d1)	0	0	0	0	120	70

Table 18. The duplicated machine-part flow matrix of the best solution

	P1	P6	P2	P4	P5	P3
M1	200	160	0	0	0	0
M4(d2)	200	80	0	0	0	0
M2(d1)	0	0	80	180	130	0
M3(d1)	0	0	0	0	210	0
M3(d2)	0	0	80	180	0	0
M2(d2)	0	80	0	0	10	120
M4(d1)	0	0	0	0	70	120

Table19. The duplicated machine-part flow matrix of the alternative solution

is ignored in the procedure of the clustering heuristic for solving CFVMF, the number of intercellular movements obtained from the proposed heuristic is compared with other approaches. Two test examples are taken from examples presented by Wu and Salvendy (1999) and Wu (1998).

The first test problem taken from Wu and Salvendy (1999) has 6 parts and 8 machine types. In the result obtained from their paper, two identical machines for

machine type 1 and 8 and only one machine for others are reported. The result obtained from the proposed heuristic also has the same result. In table 20 showing the results of two approaches, two approaches produce zero intercellular movement in the case of two duplicated machine-part clusters. However, if the goodness of grouping is considered, the solution with 4 clusters should be selected, because a grouping efficacy value, 0.8421, in 4 clusters is the best. In this example, the trade-off problem between the number of intercellular movements and the goodness of grouping is illustrated.

The second test problem taken from Wu (1998) has 13 parts and 13 machine types. The proposed heuristic generates three identical machines for machine type 5, two for machine type 1, 7 and 8 and one for others, which is the same as the result in Wu (1998). From the two approaches, 560 intercellular movements occur as shown in figure 17 representing DMPFM for the second example. However, a grouping efficacy value for the resulting clusters is 0.5926. Table 21 shows the results according to the number of clusters with different grouping efficacy values. In the case of 4 machine-part clusters,

Approach	The number of clusters	Intercellular movements	Grouping efficacy
Wu and Salvendy (1999)	2	0	0.5938
Clustering heuristic	2	0	0.5938
	3	85	0.7727
	4	265	0.8421

Table 20. The comparison of two approaches for the first test problem

the best grouping efficacy, 0.8421, is obtained, but 265 intercellular movements occur. The second test problem also requires a trade-off between 3 and 4 clusters.

5.7 Summary

This chapter proposes the clustering heuristic for solving CFVMF in CMS. The clustering heuristic described in Chapter III is modified by adding the machine capacity management step. Figure 18 represents the overall procedure of the proposed

	P5	P6	P7	P8	P9	P12	P2	P4	P10	P13	P1	P3	P11
M1(d1)	180			2200	430								
M3		120	200		860	150			560				
M4		240		2200	860								
M5(d3)	360		600										
M8(d1)	360		400										
M10	360	120			860	300							
M11	180				430	150							
M1(d2)							310		280	90			
M2							620	350	560	180			
M6								700	560				
M7(d1)									280				
M7(d2)							310	350		270			
M5(d1)											1200		1560
M5(d2)												1250	
M8(d2)											800	5000	1040
M9											800	3750	
M12											800	2500	1040
M13											400	2500	520

Figure 17. The duplicated machine-part flow matrix for the second test problem with 3 clusters

Approach	The number of clusters	Intercellular movements	Grouping efficacy
Wu (1998)	3	560	0.5936
Clustering heuristic	3	560	0.5926
	4	1120	0.6765

Table 21. The comparison of two approaches for the second test problem

clustering heuristic for CFVMF. In the proposed heuristic, the capacity of machines is first ensured, and then manufacturing cells are configured to minimize intercellular movements. In order to ensure the capacity of all machine types, the duplication of machines and the split of operations are allowed.

The new matrix concept such as DMPIM representing processing requirement, DMPFM representing material flow and DMPTM representing production time on duplicated machines, is introduced. Operations required by parts are assigned by the largest-first rule in DMPTM and material flow of parts is assigned by the same assignment pattern as DMPTM in DMPFM. The clustering heuristic for CFVMF involves setup time in a solving procedure and defines a new processing time, so called production time, including setup time.

The proposed heuristic also develops the new similarity measure incorporating processing requirement in DMPIM, material flow in DMPFM and machine workload in DMPTM. In order to merge initial duplicated machine clusters, a new group similarity measure based on material flow in DMPFM is used instead of the average similarity in

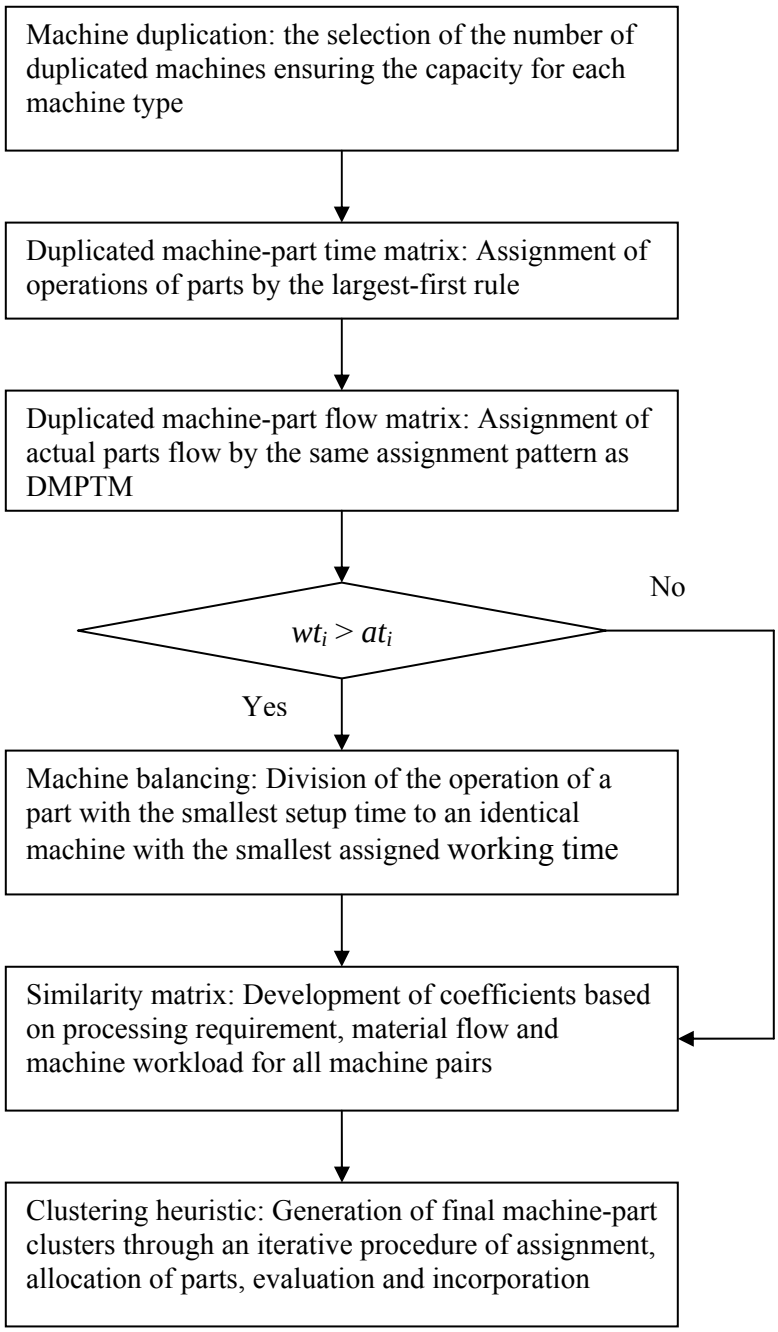

Figure 18. The conceptual illustration of the clustering heuristic for CFVMF

the ALC algorithm. For a more precise allocation of parts to duplicated machine clusters, this research introduces new allocation rules considering the maximum density as well as the number of operations and the workload of machines.

On the selection of the best solution, this chapter proposes the trade-off problem between the number of intercellular movements and the goodness of clustering. Two test problems are taken from the literature in order to evaluate the clustering heuristic proposed to solve CFVMF. The same results as the existing approaches are obtained, which shows the efficiency of the proposed clustering heuristic for the configuration and machine capacity management of manufacturing cells. Also, the result shows that the clustering heuristic for CFVMF is applicable to real manufacturing systems.

CHAPTER VI

CONCLUSION AND FUTURE STUDY

6.1 Conclusion

This dissertation presents the configuration and capacity management of manufacturing cells using cluster analysis. A new clustering heuristic based on cluster analysis are proposed to solve the given problem, so called cell formation. Thus, the objective of the dissertation is to study how well the proposed clustering heuristic is performed on cell formation in CMS.

In CHAPTER I, the objective of this dissertation and the description of the problem to be solved are provided. CHAPTER II presents a survey of the literature related to methodologies on cluster analysis and cell formation in CMS. Also, the classification and comparison of methodologies are given. CHAPTER III develops a new clustering heuristic and proposes a new similarity coefficient and assignment algorithm based on the pairwise exchange method. CHAPTER IV proposes a clustering heuristic, which is modified by adding a feedback step and more allocation rules for solving CFOPR and provides the performance evaluation. CHAPTER V presents a clustering heuristic considering the machine capacity for solving CFVMF. The conclusion and future work is mentioned in CHAPTER VI.

This research proposes a new clustering heuristic having a simple structure to overcome the high complexity of the cell formation problem. The proposed heuristic produces the best solution from one time run of algorithm and explores only one basis to

obtain the best solution. In the proposed clustering heuristic, a new similarity coefficient measure is developed by modifying the Jaccard similarity and employing the double centering transformation, which makes a similarity measure consider the relationship between two objects as well as the correlation with other objects. A new assignment algorithm is based on the new pairwise exchange method and the rough-cut concept, which makes an algorithm generate initial object clusters in a short time, but near optimal solutions. Initial object clusters are merged by the ALC algorithm to generate better clusters. Attributes are allocated to object clusters, and then the object-attribute clusters having the smallest intercellular movements are selected as the best solution through iterative procedure.

The proposed clustering heuristic is applied to CFOPR by adding a new feedback step and more precise new allocation rules. These additions make the quality of the clusters fortified. The obtained machine-part clusters are evaluated by grouping efficacy which quantifies the goodness of grouping. In order to evaluate the clustering heuristic for solving CFOPR, computational complexity is tested for the cell formation problems randomly generated from small sizes to large sizes with different difficulty levels. The result shows that the proposed heuristic is relatively well implemented no matter what the difficulty of data sets is, ill or well-structured. Also, the proposed heuristic obtains the solution in around one and half seconds, even for the largest test problem. The result of the computational performance reports that the proposed clustering heuristic is very efficient for CFOPR. Another common evaluation method is to implement the proposed heuristic for the test data sets taken from the literature and then compare it with other

approaches. From the comparison with other approaches for the existing 23 data sets in the aspect of grouping efficacy, the proposed heuristic produces the same or better solutions than others in 18 out of the total of the test data sets. Thus, we can know that the proposed heuristic is very adequate for CFOPR.

Another application of the proposed clustering heuristic is CFVMF. Only a few studies dealt with various manufacturing factors such as operation sequence, processing time, production volume, machine available time and lot size that the clustering heuristic for CFVMF considers. In particular, the setup time, which is barely discussed in the literature, is involved. In this dissertation, the following five issues are studied to solve CFVMF:

- Machine capacity should be ensured to process operations required by all the parts.
- A new similarity measure incorporating a variety of manufacturing factors should be constructed.
- Setup time should be considered in the procedure of algorithm.
- Operations should be assigned to multiple identical machines with proper rules
- The split of operations and the duplication of machines should be considered to meet machine capacity.

The solving procedure can be widely divided into two sub-algorithms. The capacity of machines is first ensured, and then manufacturing cells are configured to minimize intercellular movements. The proposed algorithm is based on three new matrices such as DMPIM, DMPFM and DMPTM. New matrices represent actual

material flow of parts and the assignment of operations on multiple identical machines. In order to ensure the capacity of machine types, the duplication of machines and the split of operations are allowed, and operations are assigned by the largest-first rule. A new similarity measure is constructed by incorporating processing requirement, material flow and machine workload.

Since no studies coincidently considered all manufacturing factors mentioned in this dissertation, two test problems that do not include setup time, but similar manufacturing factors and the same objective are taken from the literature in order to evaluate the performance of the clustering heuristic for CFVMF. In two test problems, the trade-off between the number of intercellular movements and the goodness of grouping is required to select the best solution. In the case of the same number of clusters as solutions found in the literature, the proposed heuristic produces the same intercellular movements.

The significant contributions of this dissertation, which studies the design of the heuristic based on cluster analysis for CFOPR and CFVMF, are summarized as follows:

- The new heuristic based on cluster analysis is developed in order to minimize intercellular movements.
- A new solving structure incorporating structures of hierarchical and non-hierarchical clustering methods is introduced.
- A new assignment algorithm based on the pairwise exchange method, an iterative procedure and a rough-cut concept, is proposed.

- The proposed clustering heuristic is applied to two types of fundamental problems, CFOPR and CFVMF, in the cell formation in CMS. Different algorithms for solving two problems are not required.
- New similarity measures considering processing requirement as well as a variety of manufacturing factors are introduced.
- Setup time, which is barely considered in existing research, is involved in the procedure of the heuristic. It establishes a more practical means for the configuration of manufacturing systems.
- A new machine-part matrix representing multiple identical machines and material flow is constructed. More detailed assignment of operations is available.
- Intercellular moves after considering machine duplication are calculated. Actual material flow of parts on duplicated machines is calculated.
- Various evaluation measures, i.e. grouping efficacy, grouping efficiency and so on, can be employed when a solution is selected. Various preferences for taking the desirable solution can be introduced in the procedure of the heuristic.
- The constraints, i.e. the limitation of the number of clusters, machines within a cluster and so on, can be easily limited. The proposed clustering heuristic is easily modified, which shows the flexibility of the heuristic and the easy application for various manufacturing systems.
- Test problems taken from the literature are implemented to evaluate the proposed clustering heuristic and compared the obtained solutions with other existing approaches. The performance evaluation of the clustering heuristic shows that the proposed heuristic

is applicable, adequate and efficient for two types of the cell formation problems, CFOPR and CFVMF.

6.2 Future Study

This research can be extended in several ways. The proposed clustering heuristic employs the similarity measure in order to obtain initial clusters from the assignment algorithm. A similarity measure with the better discriminate power guarantees a better solution. The similarity coefficient proposed in this research has many advantages to for achieving very good solutions. However, by employing various similarity or dissimilarity measures proposed in the literature in the proposed heuristic, the bigger domain of solutions can be explored, and those measures can be compared with each other.

The solutions obtained from the heuristic are evaluated and compared with other solutions in the aspect of the given evaluation measure. In CFVMF, this research proposes the trade-off between the number of intercellular movements and the goodness of grouping in the selection of the best solution. The compromise for this trade-off is reserved for readers. However, the proposal of the evaluation measure that can quantify the incorporation of two preference measures makes the analysis of solutions obtained from the proposed heuristic clearer and easier.

This research tests the performance of the clustering heuristic for test data taken from literature and provides the comparison of results obtained from the proposed heuristic and other approaches, which enables the proposed heuristic to directly compare with other approaches for the same test problems. From the comparison, we agree that

the proposed heuristic is a reliable and adequate approach to the application in CMS. However, by implementing the data collected in real manufacturing systems, the clustering heuristic can have an opportunity to be upgraded and be recognized as a more reliable approach.

In CFVMF, the clustering heuristic considers a variety of manufacturing factors such as unit processing time, production volume, operation sequence, setup time and so on. After careful search of the literature, no methodology for solving the cell formation considering all the manufacturing factors mentioned in this research has been found. As it is known, it is impossible to solve the cell formation considering all production factors used in real manufacturing systems. However, in order to make the proposed heuristic closer to real systems, more factors, i.e. alternative routings of parts, layout of machines and so on, can be incorporated in the solving procedure.

This dissertation studies the application of the clustering heuristic for two kinds of the cell formation problems such as CFOPR and CFVMF in CMS. The proposed heuristic based on cluster analysis is applied to these problems with only little modification, which shows the flexibility of the heuristic that enables the heuristic to be easily applied to other manufacturing systems. In the literature, various applications of GT concept have been reported. Thus, the proposed clustering heuristic, which is relatively well applied for the cell formation problems in CMS, should be extended by being applied to other manufacturing areas, i.e. printed circuit board assembly, layout of machines, the detection of wafer defects in integrated circuit manufacturing and so on, as well as science and social areas.

REFERENCES

ADIL, G. K., RAJAMANI, D. and STRONG, D., 1997, Assignment allocation and simulated annealing algorithms for cell formation. *IIE Transactions*, **29**, 53-67.

BOCTOR, F. F., 1991, A linear formulation of the machine-part cell formation problem. *International Journal of Production Research*, **29**, 343-356.

BOE, W. and CHENG, C. H., 1991, A close neighbor algorithm for designing cellular manufacturing systems. *International Journal of Production Research*, **29**, 2097-2116.

BURBIDGE, J. L., 1963, Production flow analysis. *Production Engineer*, **42**, 742-752.

BURBIDGE, J. L., 1971, Production flow analysis. *Production Engineer*, **50**, 139-152.

BURBIDGE, J. L., 1975, *The Introduction of Group Technology* (New York, NY: Wiley).

BURBIDGE, J. L., 1991, Production flow analysis in planning group technology. *Journal of Operations Management*, **10**, 5-27.

CARRIE, S., 1973, Numerical taxonomy applied to group technology and plant layout. *International Journal of Production Research*, **11**, 399-416.

CHAN, H. M. and MILNER, D. A., 1982, Direct clustering algorithm for group formation in cellular manufacture. *Journal of Manufacturing Systems*, **1**, 65-74.

CHANDRASEKHARAN, M. P. and RAJAGOPALAN, R., 1986a, MODROC: an extension to rank order clustering for group technology. *International Journal of Production Research*, **24**, 1221-1233.

CHANDRASEKHARAN, M. P. and RAJAGOPALAN, R., 1986b, An ideal-seed non-hierarchical clustering algorithm for cellular manufacturing. *International Journal of Production Research*, **24**, 451-464.

CHANDRASEKHARAN, M. P. and RAJAGOPALAN, R., 1987, ZODIAC: an algorithm for concurrent formation of part families and machine-cells. *International Journal of Production Research*, **25**, 835-850.

CHANDRASEKHARAN, M. P. and RAJAGOPALAN, R., 1989, Groupability: analysis for concurrent formation of part families and machine cells. *International Journal of Production Research*, **27**, 1035-1052.

CHEN, J. and HERAGU, S. S., 1999, Stepwise decomposition approaches for large scale cell formation problems. *European Journal of Operational Research*, **113**, 64-79.

CHEN, M. C., 2003, Configuration of cellular manufacturing systems using association rule induction. *International Journal of Production Research*, **41**, 381-395.

CHENG, C. H., GYPTA, Y. P., LEE, W. H. and WONG, K. F., 1998, A TSP-based heuristic for forming machine groups and part families. *International Journal of Production Research*, **36**, 1325-1337.

CHENG, C. H., MADAN, M. S. and MOTWANI, J., 1996, Designing cellular manufacturing systems by a truncated tree search. *International Journal of Production Research*, **34**, 349-361.

DIMOPOULOS, C. and MORT, N., 2001, A hierarchical clustering methodology based on genetic programming for the solution of simple cell-formation problems. *International Journal of Production Research*, **39**, 1-19.

DIMOPOULOS, C. and MORT, N., 2004, Evolving knowledge for the solution of clustering problems in cellular manufacturing. *International Journal of Production Research*, **42**, 4119-4133.

GONCALVES, J. F. and RESENDE, M. G. C., 2004, An evolutionary algorithm for manufacturing cell formation. *Computers and Industrial Engineering*, **47**, 24-273.

GUPTA, T. and SEIFODDINI, H., 1990, Production data based similarity coefficient for machine-component grouping decision in the design of a cellular manufacturing system. *International Journal of Production Research*, **28**, 1247-1269.

HALKIDI, M., BATISTAKIS, Y. and VAZIRGIANNIS, M., 2001, On clustering validation techniques. *Journal of Intelligent Information Systems*, **17**, 107-145.

HANSEN, P. and JAUMARD, B., 1997, Cluster analysis and mathematical programming. *Mathematical Programming*, **79**, 191-215.

HARHALAKIS, G., NAGI, R. and PROTH, J. M., 1990, An efficient heuristic in manufacturing cell formation for group technology application. *International Journal of Production Research*, **28**, 185-198.

HASTIE, T., TIBSHIRANI, R. and FRIEDMAN, J., 2001, *The Elements of Statistical Learning; Data Mining, Inference, and Prediction* (New York, NY: Springer-Verlag).

HERAGU, S. S., 1994, Group technology and cellular manufacturing. *IEEE Transactions on systems, Man, and Cybernetics*, **24**, 203-214.

HERAGU, S. S. GUPTA, Y. P., 1994, A heuristic for designing cellular manufacturing facilities. *International Journal of Production Research*, **32**, 125-140.

JOHNSON, R. A. and WICHERN, D. W., 1998, *Applied Multivariate Statistical Analysis 4th edition* (Upper Saddle River, NJ: Prentice Hall).

KHAN, M., ISLAM, S. and SARKER, B., 2000, A similarity coefficient measure and machine-parts grouping in cellular manufacturing systems. *International Journal of Production Research*, **38**, 699-720.

KING, J. R., 1980, Machine-component grouping in production flow analysis: an approach using a rank order clustering algorithm. *International Journal of Production Research*, **18**, 213-232.

KING, J. R. and NAKORNCHAI, V., 1982, Machine-component group formation in group technology: review and extension. *International Journal of Production Research*, **20**, 117-133.

KUMAR, C. S. and CHANDRASEKHARAN, M. P., 1990, Grouping efficacy: a quantitative criterion for goodness of block diagonal forms of binary matrices in group technology. *International Journal of Production Research*, **28**, 603-612.

KUSIAK, A., 1987, The generalized group technology concept. *International Journal of Production Research*, **25**, 561-569.

KUSIAK, A. CHOW, W. S., 1987, Efficient solving of the group technology problem. *Journal of Manufacturing Systems*, **6**, 117-124.

KUSIAK, A. CHOW, W. S., 1988, Decomposition of manufacturing systems. *IEEE Transactions on Robotics and Automation*, **4**, 457-471.

LINGRAS, P. and HUANG, X., 2005, Statistical, evolutionary, and neurocomputing clustering techniques: cluster-based vs object-based approaches. *Artificial Intelligence Review*, **23**, 3-29.

LOGENDRAN, R., 1991, Impact of sequence of operations and layout of cells in cellular manufacturing. *International Journal of Production Research*, **29**, 375-390.

MACQUEEN, J. B., 1967, Some methods for classification and analysis of multivariate observations. *Proceedings of the 5th Symposium on Mathematical Statistics and Probability*, University of California, Berkeley, CA, pp. 281-297.

MCAULEY, J., 1972, Machine grouping for efficient production. *Production Engineer,* **51**, 53-57.

MCCORMICK, W. T., SCHWEITZER, P. J. and WHITE, T. W., 1972, Problem decomposition and data reorganization by clustering techniques. *Operations Research*, **20**, 993-1009.

MILTENBURG, J. and ZHANG, W., 1991, A comparative evaluation of nine well-known algorithms for solving the cell formation problem in group technology. *Journal of Operations Management*, **10**, 44-72.

MOIER, C. T., YELLE, J. and WALKER, G., 1997, Survey of similarity coefficient based methods as applied to the group technology configuration problem. *OMEGA, International Journal of Management Science*, **25**, 65-79.

NAIR, G. J. and NARENDRAN, T. T., 1998, CASE: a clustering algorithm for cell formation with sequence data. *International Journal of Production Research*, **36**, 157-179.

NAIR, G. J. and NARENDRAN, T. T., 1999, ACCORD: a bicriterion algorithm for cell formation using ordinal and ratio-level data. *International Journal of Production Research*, **37**, 539-556.

NG, S., 1996, On the characterization and measure of machine cells in group technology. *Operations Research*, **44**, 735-744.

ONWUBOLU, G. C. and MUTINGI, M., 2001, A genetic algorithm approach to cellular manufacturing systems. *Computers and Industrial Engineering*, **39**, 125-144.

SARKER, B. R., 1996, The resemblance coefficients in group technology: a survey and comparative study of relational metrics. *Computers and Industrial Engineering*, **30**, 103-116.

SARKER, B. R., 2001, Measures of grouping efficiency in cellular manufacturing systems. *European Journal of Operational Research*, **130**, 588-611.

SARKER, B. R. and MONDAL, S., 1999, Grouping efficiency measures in cellular manufacturing: a survey and critical review. *International Journal of Production Research*, **37**, 285-314.

SARKER, B. R. XU, Y., 1998, Operation sequences-based cell formation methods: a critical survey. *Production Planning and Control*, **9**, 771-783.

SEIFODDINI, H., 1988, Comparison between single linkage and average linkage clustering techniques in forming machine cells. *Computers and Industrial Engineering*, **15**, 210-216.

SEIFODDINI, H., 1989, Single linkage vs. average linkage clustering in machine cells formation application. *Computers and Industrial Engineering*, **16**, 419-426.

SEIFODDINI, H. and WOLFE, P. M., 1986, Application of the similarity coefficient method in group technology. *AIIE Transactions*, **18**, 271-277.

SEIFODDINI, H., and WOLFE, P. M., 1987, Selection of a threshold value based on material handling cost in machine-component grouping. *AIIE Transactions*, **19**, 266-270.

SELIM, H. M., ASKIN, R. G. and VAKHARIA, A. J., 1998, Cell formation in group technology: review, evaluation and directions for future research. *Computers and Industrial Engineering*, **34**, 3-20.

SHTUB, A., 1989, Modeling group technology cell formation as a generalized assignment problem. *International Journal of Production Research*, **27**, 775-782.

SRINVASAN, G., 1994, A clustering algorithm for machine cell formation in group technology using minimum spanning trees. *International Journal of Production Research*, **32**, 2149-2158.

SRINVASAN, G. and NARENDRAN, T. T., 1991, GRAFICS-a non-hierarchical clustering algorithm for group technology. *International Journal of Production Research*, **29**, 463-478.

SRINVASAN, G., NARENDRAN, T. T. and MAHADEVAN, B., 1990, an assignment model for the part-families problem in group technology. *International Journal of Production Research*, **28**, 145-152.

VAKHARIA, A. J. and WEMMERLOV, U., 1990, Designing a cellular manufacturing system: a materials flow approach based on operation sequences. *IIE Transactions*, **22**, 84-97.

VENUGOPAL, V. and NARENDRAN, T. T., 1992, Cell formation in manufacturing systems through simulated annealing; an experimental evaluation. *European Journal of Operational Research*, **63**, 409-422.

WANG, J. and ROZE, C., 1997, Formation of machine cells: a modified *p*-median model and a comparative study. *International Journal of Production Research*, **35**, 1269-1286.

WEI, J. L. and KERN, G. M., 1989, Commonality analysis: a linear clustering algorithm for group technology. *International Journal of Production Research*, **27**, 2053-2062.

WEMMERLOV, U. and HYER, N.L., 1989, Cellular manufacturing in the US industry: a survey of users. *International Journal of Production Research*, **27**, 1511-1530.

WEMMERLOV, U. and JOHNSON, D. J., 1997, Cellular manufacturing at 46 user plants: implementation experiences and performance improvements. *International Journal of Production Research*, **35**, 29-49.

WON, Y. and LEE, K. C., 2001, Group technology cell formation considering operation sequences and production volumes. *International Journal of Production Research*, **39**, 2755-2768.

WON, Y. and LEE, K. C., 2004, Modified p-median approach for efficient GT cell formation. *Computers and Industrial Engineering*, **46**, 495-510.

WU, N., 1998, A concurrent approach to cell formation and assignment of identical machines in group technology. *International Journal of Production Research*, **36**, 2099-2114.

WU, N. and SALVENDY, G., 1993. A modified network approach for the design of cellular manufacturing systems. *International Journal of Production Research*, **31**, 1409-1421.

WU, N. and SALVENDY, G., 1999, An efficient heuristic for the design of cellular manufacturing systems with multiple identical machines. *International Journal of Production Research*, **37**, 3519-3540.

XAMBRE, A. R. and VILARINHO, P. M., 2003, A simulated annealing approach for manufacturing cell formation with multiple identical machines. *European Journal of Operational Research*, **151**, 434-446.

XU, R. and WUNSCH II, D., 2005, Survey of clustering algorithm. *IEEE Transactions on Neural Networks*, **16**, 645-678.

YASUDA, K. and YIN, Y., 2001, A dissimilarity measure for solving the cell formation problem in cellular manufacturing. *Computers and Industrial Engineering*, **39**, 1-17.

VITA

NAME	Young Hak Shim
EDUCATIONAL BACKGROUND	B.S. Industrial Engineering 1999
	Kyunghee University
	M.S. Industrial Engineering 2001
	Texas A&M University
	Ph.D. Industrial Engineering 2006
	Texas A&M University
PERMANENT ADDRESS	Hyundai Morningside APT. 104-1302
	Shinhyun-ri, Gwangju-si, Kyeunggi-do
	South Korea, 464-895