DOKUZ EYLÜL UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

DESIGN OF CELLULAR MANUFACTURING SYSTEM WITH WORKER ASSIGNMENT

by

Müge AKPINAR

March, 2013

İZMİR

DESIGN OF CELLULAR MANUFACTURING SYSTEM WITH WORKER ASSIGNMENT

A Thesis submitted to the

Graduate School of Natural and Applied Sciences of Dokuz Eylül University

In Partial Fulfillment of the Requirements for the Degree of Master of Science in Department Of Industrial Engineering

by

Müge AKPINAR

March, 2013

İZMİR

M.Sc THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"DESIGN OF CELLULAR MANUFACTURING SYSTEM WITH WORKER ASSIGNMENT"** completed by **MÜGE AKPINAR** under supervision of **ASSIST. PROF. DR. ÖZCAN KILINÇCI** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

ASSIST. PROF. DR. ÖZCAN KILINÇCI

Supervisor

ASSIST. PROF. DR. GÖKALP YILDIZ

(Jury Member)

ASSOCIATE PROF. DR. ZEKİ KIRAL

(Jury Member)

Prof.Dr. Ayşe OKUR
Director
Graduate School of Natural and Applied Sciences

ACKNOWLEDGEMENTS

I am appreciated to my family for their support on me at whole my life, but especially on this study.

And also I feel myself much more lucky by having a guide as Assist. Prof. Dr. Özcan Kılınçcı.

And at the end, I would like to thank to TÜBİTAK for their monetary support.

Müge AKPINAR

DESIGN OF CELLULAR MANUFACTURING SYSTEM WITH WORKER ASSIGNMENT

ABSTRACT

Cellular manufacturing is one of the most effective approaches in manufacturing area, when the system contains similar products by shapes, functions or production methods. Cell Formation Problem is the main problem in cellular manufacturing systems.

In this thesis, a specific cell formation problem, which consists of alternative routes of parts, different process times at different machines for each part, manufacturing cost, different inter-cell material handling cost for different parts, different machine complexity degree, different worker talent degree, multifunctional workers and worker training cost. To solve the problem, an algorithm based on Simulated Annealing algorithm is presented. The number of the cells is determined in the presented algorithm using Kaiser's Rule by Bashir & Karaa (2008). To test the proposed algorithm, three cell formation problems are randomly generated, the small sized consists of five machines and seven parts; the medium sized consists of ten machines and fifteen parts; and the large sized consists of eighteen machines and thirty parts. The results show that the proposed algorithm produces good results giving the minimum total cost which includes manufacturing cost, inter-cell material handling cost, and worker training cost. Also effects of the inter-cell material handling cost and the worker training cost on the total cost are analyzed by changing each cost.

Keywords: Cell formation, cellular manufacturing, heuristics, human issue, worker training, worker assignment, simulated annealing.

ÇALIŞAN ATAMALI HÜCRESEL İMALAT SİSTEMİ TASARIMI

ÖZ

Hücresel imalat sistemleri, imalat alanında çok etkili bir yöntemdir, özellikle şekil, fonksiyon veya üretim yöntemi açısından benzer parçalar söz konusu olduğu zaman. Hücresel imalat sisteminin temel problemi Hücre Oluşturma Problemi'dir.

Bu tezde, alternatif rota, makinalarda parça bazlı farklı imalat süreleri, üretim maliyeti, hücreler arası parça bazlı taşıma maliyeti, makine bazlı zorluk derecesi, operatör bazlı yetenek derecesi, çok fonksiyonlu operatörler, ve çalışan eğitim maliyeti bilgilerini içeren nitelikleri belli bir hücre oluşturma problemi çalışılmıştır. Problemi çözmek için Tavlama Benzetimi algoritmasına dayanan bir algoritma sunulmuştur. Hücre sayısı belirlemek için sunulan algoritmanın içinde Bashir ve Karaa (2008) tarafından bulunan Kaiser Kuralı kullanılmıştır. Oluşturulan algoritmayı denemek için rastgele üç hücre oluşturma problemi geliştirilmiştir, küçük boyda olan 5 makine ve 7 parçadan, orta boyda olan 10 makine ve 15 parçadan ve büyük boyda olan 18 makine ve 30 parçadan oluşmuştur. Sonuçlar, sunulan algoritmanın, üretim maliyeti, hücreler arası taşıma ve çalışan eğitim maliyetinden oluşan minimum toplam sistem maliyetini veren iyi sonuçlar ortaya çıkarmıştır. Ayrıca eğitim maliyeti ve hücreler arası taşıma maliyetinin toplam maliyet üzerindeki etkisi, birim maliyetler değiştirilerek analiz edilmiştir.

Anahtar kelimeler: Hücre oluşturma, hücresel imalat, sezgiseller, insan faktörü, çalışan eğitimi, çalışan ataması, tavlama benzetimi.

CONTENTS

Page

LIST OF FIGURES

Page

LIST OF TABLES

Page

CHAPTER ONE
INTRODUCTION

1.1 Cellular Manufacturing System

A manufacturing system consists of labor, machine and raw materials. When they are brought together to produce a product in a same place, the system gets ready to work. Many types of manufacturing systems exist in the real world. For example; assembly lines, flexible manufacturing systems, project-based (make to order) manufacturing systems, batch type manufacturing systems, continuous (make to stock) manufacturing systems, and cellular manufacturing systems. Cellular manufacturing system is studied in this thesis.

Cellular manufacturing systems (CMS) are not included in traditional systems which have been used for the several years. CMS is based on similarity of parts and functions of machines. This system works by bringing some machines together with some parts in a common area. Some rules or assumptions are used while this integration is carried out. First, the similarity between parts should be determined. Similar parts form "families", when they get together. This similarity is generally determined by their relationship with machines in the production system.

Then machines, which are intensively related with families, are assigned to these families. At the end of this assignment, the group, formed by interrelated parts and machines, is called as "cell".

A rule should be discussed for cell formation problems. The Kaiser's Rule is a good guide to form cells. This rule uses within-group correlations and calculates the optimal number of cells in a production system with several parts and machines. After that, cell compositions and families can be determined easier.

As Cellular Manufacturing system provides grouping similar parts into part families and the corresponding machines into cells, it constructs small imaginary manufacturing plants which are responsible for only themselves in the whole manufacturing system. Thus, gaining ascendancy over these small plants gets easier and a better control approach is provided on manufacturing area in this way. This new system provides reduced paper work, reduced labor, better supervisory control, reduced tooling, reduced setup time, reduced delivery time, reduced lead time, reduced rework and scrap materials, reduced lot size, reduced work in process, reduced inventory, reduced material handling, easier scheduling and improved quality efficiency and flexibility in manufacturing system.

After configuring a cellular manufacturing system, there are many issues that should be taken into consideration. For example; capacity and quantity of machines, routes, types and quantities of parts, abilities of workers, part carrying costs, etc. Many methods are used to solve cell formation problems in the literature. They have both advantages and disadvantages. These approaches could be mainly classified into three groups,

1)Part oriented approaches
2)Process oriented approaches
3)Visual inspection method

Visual inspection method generally does not work effectively. The separation of parts depends on the visual ability of worker. That means it relies on personal experience and carefulness.

The part-oriented approaches use the shapes or functionalities of parts to form families and groups by some classification and coding methods. However the configuration of cell can not completely be done by these techniques.

Process oriented approaches work by manufacturing similarities as the similarities of parts' routes. The advantage is, these approaches only use machines which are required for the part.

A detailed classification for these approaches is shown in figure 1.

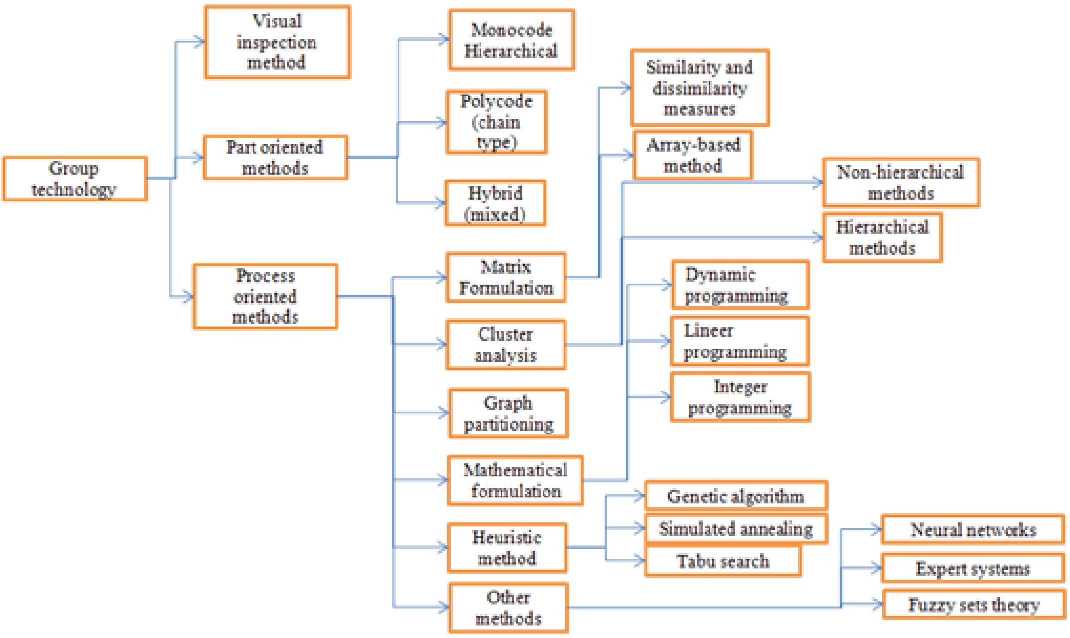

Figure 1.1 A detailed classification of cell formation methods

1.2 Framework Of The Thesis

Various studies on cellular manufacturing systems and their main points have been evaluated and discussed for this thesis. It is understood that the implementing cellular manufacturing system provides many advantages in manufacturing areas. First, the possibility of applying cellular manufacturing system to current system should be analysed. After that, an effective method should be found to form cells and to implement cellular manufacturing system. The point is, whole characteristics of the existing manufacturing system should be included by implemented system. Many approaches are defined in searched studies. Some of them have several assumptions, and some of them are suitable just for more stable systems. However the issue, related with human or workers, is always ignored in studies. Because this topic is hard to study with its generally behavioural based structure. Human issue has a big importance on manufacturing systems. Because human is the main component of the system.

It can be seen that few studies exist on literature for human issue, which are not appropriate to simulate real systems. Because of that, it is decided to study cell formation problem with human issue in this thesis. Technical skills and trainability is used for workers. This means workers can be multi-functional in this study. Because a worker should be suitable for a cell which we assigned to. In real life situations, training is usually used and assignments are done by workers’ skills. In this study, it is tried to be as close as it gets to a real system. Some assumptions are also used.

A part may have alternative routes in a real life manufacturing system. This provides flexibility in the system. Because of that, alternative routes are included in this study. It is also called as “alternative machines for a part”, in the study.

Operations are carried out for multi-period in a real life manufacturing system. But multi-period calculations make system more complicated and hard to solve. Because of that, a cellular manufacturing system for single period is modelled.

Simulated Annealing(SA) heuristic is used to solve the derived problems. It can be seen from literature survey that SA heuristic is easier and faster than other heuristics to solve a model.

A couple of numerical examples are derived for a 5x7, 10x15 and 18x30 (machine-part) dimensioned problem.

In problems, this study tried to form cells which include workers with specified number of machines and parts, by minimizing the objective cost function using Matlab R2008a.

1.3 Outline Of The Thesis

This thesis consists of chapters. Chapter one gives some basic informations about Cellular Manufacturing Systems and applications of solving cell formation problems. Also framework of the thesis is explained in this part.

The components of the problem are explained and a detailed literature review is done for Cellular Manufacturing in Chapter two. Everything about Cellular Manufacturing (benefits, applications, hardnesses, deficiencies, studied and unstudied issues in literature, studies, etc.) is explained in this chapter.

The components of the problem and algorithm are explained in Chapter three. Heuristics have an important role on solving problems in an acceptable time interval and with an acceptable performance. So we need to learn and use them to have better and applicable results. Also the characteristics of the derived problem are explained in this chapter. Proposed Algorithm which is constructed for the derived problem is shown in this chapter.

In Chapter four, computational results of the problem are explained with all points. The problem is explained by numerical examples, and results.

In Chapter five, the conclusion is explained and interpreted.

CHAPTER TWO
CELLULAR MANUFACTURING SYSTEM AND CELL FORMATION PROBLEM

2.1 Introduction

Cellular manufacturing system (CMS) is an implementation of Group Technology (GT) to the manufacturing area. Group technology is a manufacturing philosophy that has established the potential to contribute positively in batch-type production, and it endeavours to combine the flexibility of the job production system with the high productivity of the flow production system (Ham, Hitomi, & Yoshida, 1985). Mosier & Taube (1985), and Shunk (1985) define GT as;

"...a disciplined approach to identify things such as parts, processes, equipment, tools, people or customer needs by their attributes, analyzing those attributes looking for similarities between and among the things; grouping the things into families according to similarities; and finally increasing the efficiency and effectiveness of managing the things by taking advantage of the similarities. "

The most common two incorporation methods for determining part families and cells are called classification and coding. Some rules are used while determining part families and cells. It was explained in chapter one, that at the beginning of cell formation the similarity between parts should be determined. Coding is used in here. Cell determination, which is the main point of CMS, is carried out after that similarity determination. Then some other usual problems may occur as in traditional systems. The layout of machines and cells should be decided. Actually, it is as important as forming cells. Cells should be well emplaced to avoid unnecessary traffic in manufacturing area. An inefficient designed layout would cause worse results.

The aim in CMS is minimizing the overall cost, while maximizing the effectiveness in manufacturing area. This cost is called Objective Function in models. Models are used to solve real systems. For example; mathematical models, heuristic algorithms, simulation models etc. A model of the real system is constructed before the execution. Trials will cause a big cost in real manufacturing systems. We can’t change places of machines and routes of parts for several times in a manufacturing area to find the right decision. Because of that, models of real systems are used. They help us to find the optimal arrangement without any changes in the manufacturing area. At the end, we can apply the optimal result to the real manufacturing system at once.

The components of Objective Function are explained in the next section.

2.2 The Cost Components Of Cellular Manufacturing Systems

The objective function consists of costs which occur in the manufacturing system. These costs are; manufacturing cost, inter-cellular material handling cost, setup cost, hiring-firing costs and training cost.

The manufacturing cost denotes the value of cost that occurs by machines which are visited by parts. A matrix, which shows the relation between parts and machines, is needed for a manufacturing cost calculation. This matrix is called part-machine incidence matrix, and formed by 1’s and 0’s. 1(one) means, that part visits that machine. 0(zero) means that part has no operation by that machine. If the sequence of operations is important in a system, 1’s and 0’s turn to order numbers of operations. Sequence may change cost calculation, family formation and lead time of the system. The reason is, difference in the similarity calculation and assignment of parts to machines. The existence of sequence for parts may change the objective value, and the cell configuration. Another issue that changes the cost calculation is importance level of parts. In some manufacturing systems, an importance level is assigned to products. This level is shown by weight parameters. A part may have higher weight parameter according to production volume or operation time of that part (Yin & Yasuda, 2006). Furthermore importance level may be used for some other issues. These issues show characteristics of the manufacturing system. This also changes the objective value because of different cell formations.

Another issue, that changes the manufacturing cost, is alternative routes. A part may have more than one route to accomplish its process. This method is usually used in real manufacturing systems. Because this gives flexibility to the system and decreases the lead time.

The second part of objective function is inter-cellular material handling cost. This is the cost caused by the traffic between cells. If a part needs an operation out of a cell, then it should generate a move out from this cell. This move is called inter-cellular material handling. Inter-cellular movements are caused by exceptional elements or operations which are carried out by some machines in some other cells, basically. However this does not mean that intra-cell movements are not important for cost control. The layout problem, for both intra-cell and inter-cell systems, is another problem of implementing Cellular Manufacturing systems. When we have a better cell configuration, we will have lower inter-cellular material handling cost. The companies which deal with unreliable customer demands, and want to survive, have to briskly adapt themselves to changes and organise production system in accordance with these changes.

The third part of objective function is setup cost. Setup cost occurs when a machine operates several parts which is also called multi-functionality. This ability may avoid opportunity cost. The part selection is an important issue. More similar parts will cause less setup cost. Because a machine will need less setup. Setup cost also occurs when multiple period is analysed in manufacturing system. Demand rates change for each period. When different demand rates occur, machines will need setup to be able to satisfy customer demand. Cell configuration is also effective for this issue. When cells in the manufacturing system are well configurated according to all periods, less changes of cells and machine setups will be needed.

The fourth part of objective function is hiring-firing cost. Number of workers is important when forming a cellular manufacturing system. Each cell should have a worker. When we don't have enough workers or have more than needed, workers are hired or fired. But this causes cost as fine or redundant wages.

The last part of objective function is training cost. A trained worker can operate various types of machines. This supplies flexibility to the manufacturing system on scheduling. Training and cross-training also eliminate monotony and support motivation. The main point in forming a system is, positioning the worker to the appropriate cell and train if needed. However this brings cost. Because of that, cell formation methods try to construct a balance between training a worker and forming a cell. When a machine is more complex than talent level of worker in the cell, that worker should be trained to that complexity level. A new trained worker would not be as efficient as a high talented worker in real life.

Actually, all components of Objective Function cost equation mainly depend on formed cells in the manufacturing area. Different cell formations may cause different cost calculations. Because of that, cell formation methods have a big role on cost calculation of the whole system.

The most common cell formation method is similarity determination. It is also a very easy method to use. However it is not capable to real systems. Because this method does not let use constraints and any other attributes of the real systems except weights or sequences of parts. Many researchers have many formulations for similarity calculations. However the characteristics of the system, which are explained above, change these formulations. After the similarity determination, the machines which have highest similarity values, form a cell. And cellular manufacturing system is constructed. However this method does not form healthy manufacturing cells. Because this calculations does not show the optimal number of cells, and different combinations should be tried or a specified number should be given to the system. An overview for these different formulations of different characteristics is shown in the Table 2.1.

The formulas in the Table 2.1 denote;

a: number of parts visit both machines;

b: number of parts visit machine i but not j;

c: number of parts visit machine j but not i;

d: number of parts visit none of machine i and j.

Table 2.1 Similarity coefficient formulas for general purpose (Yin & Yasuda, 2006)

Similarity Coefficient	Definition S_{ij}	Range
Jacard	a/(a + b + c)	0-1
Hamann	[(a + d) -(b + c)]/[(a + d) + (b + c)]	-1 to 1
Yule	(ad - bc)/(ad + bc)	-1 to 1
Simple matching	(a + d)/(a + b + c + d)	0-1
Sorenson	2a/(2a + b + c)	0-1
Rogers and Tanimoto	(a+d)/[a+2(b+c) + d]	0-1
Sokal and Sneath	2(a + d)/[2(a + d)+ b + c)	0-1
Rusell ind Rao	a/(a + b + c + d)	0-1
Baroni-Urbani and Buser	$[a + (ad)^{1/2}]/[a + b + c + (ad)^{1/2}]$	0-1
Phi	$(ad - bc)/[(a + b)(a + c)(b + d)(c + d)]^{1/2}$	-1 to 1
Ochiai	$a/[(a + b)(a + c)]^{1/2}$	0-1
PSC	$a^2/[(b + a)*(c + a)]$	0-1
Dot-product	a/(b + c + 2a)	0-1
Kulczynski	1/2[a/(a + b) + a/(a + c)]	0-1
Sokal and Sneath 2	a/[a + 2(b + c)]	0-1
Sokal and Sneath 4	1/4[a/(a + b) + a/(a + c) + d/(b + d) + d/(c + d)]	0-1
Relative matching	$[a + (ad)^{1/2}]/[a + b + c + d + (ad)^{1/2}]$	0-1

In Table 2.1, similarity coefficient formulas for general purpose, which are used in the literature surveys, are seen. But there exist more than these formulas. And some formulas are derived from these general purposed ones. Hwang & Ree (1996), Gupta (1993), Won & Kim (1997), and Won (2000) constructed similarity coefficient formulations with alternative routes. Gupta’s formulation also includes operation sequences, production volumes and operation times.

Vakharia & Wemmerlöv (1990), Selvam & Balasubramanian (1985) constructed formulations with operations’ sequences. Seifoddini (1988) used Jaccard similarity coefficient. His formulation also uses operation volumes. Choobineh (1988) and Tam(1990) are other researchers worked on operations’ sequences. The difference is, they worked operations’ sequences by distances. Balasubramanian & Panneerselvam (1993) studied on operation sequences, additional cell arrangements, production volume, over moves and costs of them.

It was told that weight parameters were used in cell formation methods. It is also used in similarity coefficient determination. The basis is general purposed similarity coefficient formulations. The adaptation is done by injecting the weight into the general purposed formulation.

The another main point in our study is Worker Assignment. This issue is discussed in the next section.

2.3 Worker Assignment In Cellular Manufacturing Systems

In previous parts, it was told that the worker was one of components of Cellular Manufacturing System. Although it is a very important issue, worker assignment is always eliminated in many studies. Because analyzing human in manufacturing systems is difficult. Assignment should be carried out by skills of workers. These skills can be divided into technical and human skills. But making this classification is also hard. Studies are done to be able to useful for real manufacturing areas. The more we eliminate issues when modeling the system, the less studies match with real life. Forming a suitable manufacturing system is the beginning of effectiveness. Using resources effectively is as important as choosing the right system. Cellular manufacturing is both an advantageous and an easy system to use. However forming a cell is not easy as using it. Also worker assignment makes it harder. Because of this hardness, many heuristics are used to build the system in the literature.

We know, a cell is composed from a family and related machines. On the other hand, worker is main point of a cell. When we form a cell, we should supply suitable machines and suitable workers for the family of parts. When machine or worker on the hand is not available, multi-functional machines and training or cross training for workers, may be a solution for the system. But this brings extra cost.

Many attributes of workers affect the system. For example, motivation, education, trainability, multi-functionality, assiduousness, ability, etc. Generally multi-functionality, ability and trainability attributes are used in system modeling studies.

When modeling a real life problem, we can apply human issues in eight broad areas: worker assignment strategies, skill identification, training, communication, autonomy, reward/compensation system, teamwork, and conflict management. (Bidanda, Ariyawongrat, Needy, Norman, & Tharmmaphornphilas, 2005)

Assignment is the most important part of manufacturing systems. It is also so difficult. Ability, trainability, suitability of worker and needs of the manufacturing area should be analysed. A wrong decision will cause an extra cost of training, material handling or re-work. Assignment can be done by mathematical models or some other heuristics.

Skill identification should be done properly. It is used to compare with the task and assignment will be carried out by this identification. As told above, it can be divided to human and technical skills. Technical skills are generally some skills on accomplishing the task. Human skills are about personal communication, harmony with the team or motivation. Training or cross-training becomes a part of assignment in here. When worker is not suitable for the task, training will be a solution for the problem. But this brings extra cost for the manufacturing system. And also ability of worker should be analysed before this method. Because a training would not give the same result for different workers. The worker should be able to talent-upgrade. Everybody has some capacity, but not same as each other. Also the amount of training would not be same for everyone. An analyse should be done for both worker and task, before a training application.

Communication ability of workers in a manufacturing area is important. Communications within workers and between workers and management are important for task definitions and assignments, problem definitions, and solutions of these problems. Also communication has a significant importance on training.

Type of autonomy should be well analysed. In a manufacturing system the control can be given to the worker for a work area, or everything can be managed from the top management. In cellular manufacturing, workers have their own responsibilities in their cells. Cellular manufacturing system uses advantages of this autonomy type. When workers have responsibility of their cells, they may be able to solve problems occur in the cell, they can maintain machines in the cell, they can manage their cell. Manufacturing area will be divided into small manufacturing organisations that each have a keeper inside. Because of that, this system needs multi-functional workers.

Motivation is a very important issue in manufacturing systems. It affects to many issues on humans, especially carefulness. Reward and compensation systems are used for motivation of workers. It is a remarking system for both positively and negatively; positively to support the good actions, and negatively to eliminate the bad and wrong actions.

Generally workers work in teams in manufacturing systems. A team may consist of personnel from many areas. That means communication within the team and acting through team's destination is an important issue in manufacturing systems to achieve success. Team's success means system's success. This will be achieved by the harmony of team members. The harmony is for both technical and personal skills.

In a manufacturing system, there may work many workers in same area. And one will affect the other by his own function in the manufacturing system. Or some different ideas may be occur in the system. Conflict management have a significant role in this situation. Many workers work together with many different skills and positions in the same area. Conflict is sometimes an unavoidable situation. It is important to turn this situation to a helpful and an useful situation. This is called conflict management and headmen take role in this position.

Worker assignment affects the manufacturing system both by cost and productivity. It is an inter-related issue with cell formation in a cellular manufacturing area. This issue is as important as part and machine similarity. Because the compatibility of worker with machines may be strengthen or weaken the stability of a cell. A literature review is carried out in the next section on Cellular Manufacturing System Solution Methods and Human Issue in this system.

2.4 The Literature Review On Cellular Manufacturing Systems

A general literature review is carried out in the next part for cellular manufacturing systems. And a literature review for human issue is carried out after that part.

2.4.1 Cellular Manufacturing Systems

In Cellular Manufacturing (CM), in each cell, some operations are done on parts by machines, so that the main objective is maximizing the intra-cell operations while minimizing the number of inter-cell movements (Saeedi, Solimanpur, Mahdavi, & Javadian, 2010).

Some researchers analyzed cellular manufacturing systems by their studies. Guerrero, Lozano, Smith, Canca, & Kwok (2002) studied cellular manufacturing by weighted similarity coefficients, a new self-organizing neural network and a linear network flow model. Cell formation has two steps: first, part families are formed and then machines are assigned. Also a Maximum Spanning Tree heuristic is used in their study to compare the results. Self-organizing neural network is used in the main part of the problem; forming part families.

Wu, Chu, Wang, & Yan (2007) studied hierarchical genetic algorithm for cellular manufacturing. In their problem, routing (sequence), work load, machine capacity, demand, batch size, and layout type are searched. Cell formation and layout design are carried out simultaneously. First, a mathematical model is constructed. Then genetic algorithm is used for cell formation problem. Crossover and mutation are both used. Dynamic assignment is done in their problem.

Balakrishnan & Cheng (2007) studied cellular manufacturing problem with multi-period. They also use demand and resource uncertainty. This manufacturing system is harder than single-period to construct. Also the demand and resource are not known. A mathematical model is constructed.

It is seen that many studies were done for single period, which demand and some other values were constant. Because multi-period systems are hard to solve. But it does not occur like this in real life systems.

Safaei, Mehrabad, & Ameli (2008) studied a dynamic cellular manufacturing system with a hybrid SA algorithm. A mixed-integer programming model is also developed in their study. The advantage of their study is, they calculate inter and intra-cell material handling by sequence of operations. That means, both side of traffic between two cells are calculated. And machine replication is allowed in the model. The hybrid system consists of mean field annealing algorithm and simulated annealing algorithm. It was explained before that SA algorithm needed an initial solution to be able to improve it. In this study the mean field annealing algorithm is used for that initial solution which is needed by SA algorithm. Mean field annealing algorithm is a combination of neural networks. It is seen that mean field annealing algorithm increases the performance of the model and speeds up the algorithm to reach the optimal result. Back order is not allowed and a limit is determined for the maximum cell number. The problem is for multi-period. And both intra and inter-cell movements are calculated in the model. But human issue is not included in the study.

Pailla, Trindade, Parada, & Ochi (2010) studied on a comparison between simulated annealing and genetic algorithms. They use an algorithm to improve both simulated annealing algorithm results and genetic algorithm results. This algorithm uses some other crossover rule instead of classic methods. Both results for some numerical examples reach to almost same performance. But SA algorithm finds even better results than previous literature studies for the same problems.

Tavakkoli-Moghaddam, Rahimi-Vahed, Ghodratnama, & Siadat (2009) studied solving a cell formation problem by simulated annealing with two types of cells. One type can produce different types of parts, the other one can produce specific types of products. First, a non-linear mathematical model is formed. Then the model is solved by Simulated Annealing Algorithm. Three objective issues are included in the model.

These issues are; minimizing delay cost of parts, minimization of unproductive times of cells, maximizing the unused capital.

A literature survey was carried out by Saeedi, Solimanpur, Mahdavi, & Javadian (2010), on details of heuristics. Table 2.2, shows that the points which are taken or not taken into consideration by researchers who studied cell formation until 2010.

Table 2.2 Literature survey on heuristics (Saeedi, Solimanpur, Mahdavi, & Javadian, 2010)

Reference	Applied Methodology	Sequence of operation	Production Volume	Exceptional Elements (Voids)	Intercellular Movements
Islier	Ant Algorithm	No	No	No	No
Prabhaharan et al.	Ant Algorithm	Yes	Yes	No	Yes
Mak et al.	Ant algorithm	Yes	No	No	No
Spiliopoulos and Sofianopoulou	Ant algorithm	Yes	No	No	Yes
Kesen et al.	Ant algorithm	Yes	No	No	No
Satolgu and Suresh	Goal Programming	No	No	No	No
Kao and Fu	Clustering Algorithm	No	Nc	No	No
Pandian and Mahapatra	Neural Networks	Yes	No	Yes	Yes
Mahdavi et al.	Genetic Algorithm	Yes	No	Yes	No
Mahdavi and Shirazi	Heuristic Aıgorithm	Yes	No	Yes	No
Arkat et al.	Simulated Annealing	No	Yes	No	No
Ahi et al.	TOPSIS	Yes	No	Yes	No
Wang et al.	Scatter Search	Yes	Yes	No	No
Murugunandam et al.	GA + Tabu Search	Yes	Yes	No	No

We can see that, some researchers did not work with sequence of operations which are related with inter and intra-cell movements. This means, traffic between two parts for one direction or for both directions has same importance. This assumption may affect the result.

We can see another point that, volume of parts are not taken into consideration by some researchers. The costs of material handlings for one part and for many parts are not same. And a part with high quantity can get ahead about reducing cost, instead of a part with low quantity even though unit material handling cost is lower.

Some researchers calculate the affect of exceptional elements but most of them do not. Exceptional elements mean inter-cellular movements. When a part, which is not totally belong to a cell, needs an operation; it should enter that cell. Or if a machine, which a part needs an operation from, takes place out of a cell; part should go out of this cell.

It can be seen in Table 2.2 that, many researchers did not calculate inter-cellular material handling. Inter-cellular movements are the most important cost part of the objective function. A cell formation method which is applied with this assumption would not be realistic. Because the main point in cost calculation and cell formation is minimizing the inter-cellular movements which means trying parts to make stay in their cells.

When we want to use heuristics to solve our problems, we need to have some assumptions to be able to achieve results. If we have fewer assumptions, then our model will respond closely to the real life problems. Also the importance of the assumption for that problem is a point that should be critically determined. If the issue that we make an assumption is a main point of our problem or a performance criteria, then our model would not respond as good as we expect.

2.4.2 Human Factor In Manufacturing

In this part, a literature survey is carried out on human issue in cellular manufacturing systems. And a summary table is done.

Dawis & Mabert (2000) studied on worker assignment and order releasement. They implement two different mathematical models. Instead of productive resources, they decided to study on inventory reducing formulations. Reassignment is allowed in their models. Two issues which they studied, should be in a harmony. So they implement two types of algorithm to achieve that harmony. First one is worker assignment and order releasement are done simultaneously. The second method is sequentially. Then they implement a heuristic, to be able to see the difference of these two methods. At the end of the study it is seen that sequentially calculated model gives better results and it is more sensitive to critical time intervals.

So, worker assignment can be divided into two categories:

1) Post-cell formation worker assignment (Norman, 2002).

2) Simultaneous formation of cells and worker assignment. (Aryanezhad, Deljoo, & Mirzapour, 2009)

Stevens & Campion (1994) have found 14 KSA (knowledge,skill,ability) types in their study. They say that these 14 different types can be used by an assignment problem. Erin, Fitzpatrick, Ronald, & Askin (2005) say that a worker should be analysed not only for technical skills but also for interpersonal skills. They worked on a mathematical model. In their model, it is known which machine is located in which cell, at the beginning of the model. And they studied on a heuristic model that is called Balanced Heuristic Model. When multiple teams of workers needed, this heuristic is used but it does not give good results if we have workers more than needed in the system. Workers are chosen by their inertias to the teams. Also they use Kolbe Conative Index to be able to measure the instinctive behaviour of workers.

Multi-functionality has a big role on assigning workers to the teams. Also training is the main point of it. Slomp, Bokhorst, & Molleman (2005) have a study on cross-training of workers. They used an integer programming model to allocate workers to cells. Their model also decides if that worker should be cross-trained, to be able to balance the work load on them. The objective is minimizing the cost, while allocation is being done. They say that some skill identifications should be done. And it is assumed in their model that, if a worker is cross-trained, his productivity would be lower than a worker which is already able to operate that machine. The model also has some constraints like limit on multi-functionality and machine redundancy. And the training is given only for multi-functionality, not for upgrading workers' productivity. Azizi, Zolfaghari, & Liang (2010) has a study on job rotation. They say that boredom should be eliminated as possible to be able to make workers learn operations. They studied on a mathematical model with skill identification and boredom, and a metaheuristic SAMED-JR for large scaled problems. This metaheuristic is a combination of Simulated Annealing Algorithm(SA) and Genetic Algorithm(GA). They found that the metaheuristic, they used, gave better results than using only SA or only GA.

Multi-functionality is also called as labor flexibility. This is the ability of assigning workers to different operations. Cesani & Steudel (2005) define that as intra-cell operator's mobility. They classify labor strategies according to the machine and operator assignment as dedicated, shared and combined. They used simulation to see the effect of workload balancing. At the end of their study, it can be seen that labor flexibility has a huge effect on productivity of the whole manufacturing system. They used different operator numbers and different labor assignment strategies as told before.

Askin & Huang (2001) made a study on forming effective worker teams. A mixed integer programming was used in the study. They separated workers abilities as technical and administrative. The model that they instructed includes worker assignment and training for multi-functionality. They used meta-heuristics to be able to achieve results for problems with big capacity. At the end of their study, it can be seen that meta-heuristics give good results with reasonable time for NP-hard problems. One of the model they used is Simulated Annealing(SA) algorithm. They solved more complex models with SA. It can be seen that SA could achieve optimal solution in their study. Another study was done by Aryanezhad, Deljoo, Mirzapour, & Al-e-hashem (2009) with multi-functionality of workers and also machines. They proposed a solution by Linear Integer Programming model. Objective function includes manufacturing costs, material handling costs and personnel costs. The main point of the study is, more than one period is included. But this model can be used only for small sized problems. Because a real life problem would convert the system to NP-hard problem.

Corominas, Pastor, & Rodriguez (2006) studied on a real system with multi-functional workers. This study is not based on cellular manufacturing but applied in a real life manufacturing system and investigates multi-functional worker assignment. The difference from other studies is, they assigned tasks to operators. Problem is for multi-period. Therefore, a mathematical model would come out with NP-hard situation. Researchers applied another method. They solved problem for one period and then allowed results as an input for the next period. So they divided the planning horizon.

Another study with multi-period time horizon is done by Mahdavi, Aalaei, Paydar, & Solimanpur (2010). Mathematical model was constructed. All cost issues like hiring, firing, intercellular material handling etc. were used in objective function. But cell number determination was not included in the study. This means, the cell number was specified and allowed to the problem as an input. Model applies flexibility on worker assignment, but does not include the cost. Backorder opportunity exists in the model. But it is NP-hard problem for big scaled problems.

It can be seen that studies for multi-functionality are based on the training and motivation of workers. It is accepted that a worker can be equal to a couple of workers, at least more than one, by cross-training. Same tasks for long periods will bring monotony for workers. This concept usually brings inattentiveness and work-related accidents, too. Demand will also be flexible in a short time interval. In real life situations, companies should be able to respond demands as fast as they change. Multi-functional machines are an one of alternative applications. But still, multi-functionality of workers are needed. A multi-functional machine can be operated by a multi-functional operator, or different operators should be used for different operations. But it is not a realistic application for a real life situation. Because training is costed generally lower than hiring a worker. A decision making position appears in here; which worker should be trained for which operation. This becomes the basic unit of assignment nowadays.

To be able to have feasible work situations with workers, we should analyse their abilities. Suitable tasks for right abilities will upgrade workers' performance and also systems performance. This point is studied in this thesis. When a training capacity is occurred, a worker can be able to work for multiple points that usually brings high motivation. In our study, training and upgrading of talent level is possible. And a worker can work with multiple machines, which called multi-functionality.

Multi-functionality is also has a big role when number of machines are more than number of workers. We have two options in this situation; unemployed machines in the manufacturing area, or cross-training of workers. A worker can operate several machines by cross-training. His talent also should be taken into consideration. The main objective is minimizing the cost of both cross-training and unemployed time of machines. Unemployed machines mean keeping the system away from demand satisfaction. And unsatisfacted demand means receiving lower demand at the next time. Especially nowadays, in a competitive market, time is a kind of money figure. Workers should be work on whole production time, because unproductive time means cost for a manufacturing system. It can't be provided unique duty for a worker every time. Demands may be changed through some time interval. So a worker should operate several machines or be able to do several operations. In this way, a manufacturing system can satisfy demand.

In the Table 2.3, Cesani & Steudel (2005) have listed factors affecting the development and deployment of labor flexibility.

Cesani & Steudel (2005) studied on labor assignment. They say that, although many factors are identified as influential in determining labor flexibility decisions, some of them are qualitative in nature and thus, difficult to model. The model and framework presented in their work, concentrate on those aspects that can be quantified and for which information is readily available or could be determined. The propositions were considered in the framework evolved from the empirical study and their impact on system performance were investigated with the purpose of developing knowledge about the complexities of the labor allocation process in labor limited manufacturing cells.

Table 2.3 Factors affecting the development and deployment of labor flexibility (Cesani & Steudel, 2005)

Factor	**Issues**	
Layout	Equipment proximity Wort flow (organization)	Size of the cell/cellular area Location/size inter-station buffers
Equipment	Level of automation (manual, semi-automatic, CNC machinery)	Age/condition
Type of labor assignments possible	Dedicated assignments	Utilization Combined assignments (shared and dedicated)
Cross-training level of individual operators	Number of operations Proficiency level Quality of training	Bottleneck machines Machine tending requirements Individual cycle times
Workload in the cell	Variations in demand Rush jobs	Relative machine utilization
Job design	Job rotation frequency When and where to move rules Response to load imbalances	Division of activities in the cell Inter-cell vs. intra-cell mobility
Labor aspects	Responsibilities of the operators	Operator's ownership: machine vs. cell
Personnel practices	Type of supervision Wort teams	Job autonomy
Managerial concerns	Effective operator utilization	Effective machine utilization (through wortload sharing)
Administrative aspects	Leveled operator assignments Union restrictions	Focus on bottleneck operations Crosstraining vs. compensation systems

The investigated companies currently do not use formal models such as spreadsheet-based rough-cut analysis, linear programming or simulation to assign operators to machines. Most labor assignments are made based on the experience of the personnel involved with the cells. Many times, particularly at the cell implementation level, labor decisions involve a lot of trial and error and therefore, companies do not take the best use of their labor and machine resources. Management in the companies investigated expressed the desirability for models and guidelines to assist in the labor allocation process since supervisors and operators disagreed on the most appropriate labor allocation strategies. Furthermore, while developing a completely flexible workforce is a goal in both of these companies, neither of them have objective measures to evaluate the impact that increasing operators' cross-training has on cell performance. Thus, cross-training decisions are many times made arbitrarily. (Cesani & Steudel, 2005)

M\W	A	B	C	D	Dm
1	X(12)	X(0)			12
2	X(0)	X(14)			14
3	X(6)	X(4)			10
4			X(10)	X(0)	10
5			X(0)	X(6)	6
6			X(2)	X(6)	8
WL	18	18	12	12	

M\W	A	B	C	D	Dm
1	X(6)	X(6)			12
2	X(7)	X(7)			14
3	X(2)		X(8)		10
4		X(2)		X(8)	10
5			X(3)	X(3)	6
6			X(4)	X(4)	8
WL	15	15	15	15	

Figure 2.1 Example of machine worker matrix (Slomp, Bokhorst, & Molleman, 2005).

In Figure 2.1, Slomp, Bokhorst, & Molleman (2005) have shown an example of multi-functionality of workers. The difference between two matrixes shows the multi-functionality of that worker. For example, in the left-handed figure, it cen be seen that worker can operate machine 1 and 3. In the right handed figure, we can see, worker A has cross-trained and is able to operate machine 2, too. But his total workload gets lower because a worker might be less productive on a new duty.

Several authors presented a hierarchical scheme for work force organisation problems that consists of three phases: (1) planning; (2) scheduling; (3) allocation. The assignment of tasks to multi-functional workers is done during phase (3), once a schedule has been assigned to each worker. (Corominas, Pastor, & Rodriguez, 2006)

Table 2.4 shows the attributes of researches. We can see that almost all researchers used mathematical model for their problems. Some of them are for single period, and the rest are for multiple periods. Some of them used heuristics, but these heuristics have some deficits.

Table 2.4 Literature survey

RESEARCHERS	MATHEMATICAL MODEL	HEURISTIC	ANOTHER	MULTI-FUNCTIONALITY	PERIOD
Dawis and Mabert (2000)	+	Sequentially Assignment, Simultaneously Assignment		-	more than 1
Stevens and Campion (1994)			14 KSA (knowledge,skill,ability) types		
Erin, Fitzpatrick, Ronald, Askin, (2005)	+	Balanced Heuristic Model	Team selection	worker	1
Slomp, Bokhorst, Molleman (2005)	+			worker	1
Azizi, Zolfaghari, Liang, (2010)	+	SAMED-JR		-	1
Cesani and Steudel (2005)			simulation	worker	1
Askin and Huang (2001)	+	Simulated Annealing		worker	1
Aryanezhad, Deljoo, Mirzapour, Al-e-hashem (2009)	+			both	more than 1
Corominas, Pastor, Rodriguez (2006)		Result Is Input For The Next			more than 1
Mahdavi, Aalaei, Paydar, Solimanpur (2010)	+				more than 1

CHAPTER THREE
PROBLEM DEFINITON AND SOLUTION METHODOLOGY

In this thesis, simulated annealing algorithm of a cell formation problem, which number of the cell is determined by Kaiser's Rule (Bashir & Karaa, 2008), is proposed. The cost function, which we want to minimize, consists of inter-cellular movements, training and manufacturing costs. In this problem, products move by batches and demand levels are predetermined. It is studied single period. Alternative routes are included in the problem and which machine produce which part is specified. Some assumptions are included by the problem. They are;

- The problem has one period.
- 0-1 part-machine incidence matrix is used.
- Machines and parts change place in problem. Workers are stable in cells.
- The demand for each part type is known.
- The number of machines in the system is known.
- Training doesn't take any time.
- Trained worker is assumed to be reached same productivity level as high talented worker.
- Processing time for parts are randomly distributed and vary on different machines.
- The inter-cell material handling cost per batch is known and vary for different parts.
- A part may have alternative routes.
- A worker can operate more than one machine (multi-functionality).
- Cost of training depends on levels.
- Parts are moved in batches between cells.
- The machine relocation cost is 0(zero).
- Number of workers are as much as number of cells in the system.

To be able to understand the algorithm, it should be seen indices and input data;

Indices are;

Alternative machine index: a=1; 2; . . . ;A

Cell index: p = 1; 2; . . . ; P

Iteration index: t = 1; 2; . . . ;T

Machine index: k = 1; 2; . . . ;K

Part type index: j = 1; 2; . . . ; J

Talent index: y = 1; 2; . . . ; Y

Worker index: v = 1; 2; ....;V

Input data;

B(k)= operating cost of machines per time unit

BN= a large number

Bs= batch size

C(j)= number of transportation for each part

D(j)= demand vector of parts

G(j)= inter-cell material handling cost by each part

Te(t)=temperature level of SA procedure for each itration

W(v,y-1)= Cost of worker training level skips 1 to 2, 2 to 3,…, (y-1) to y

X(k,j)= part-machine incidence matrix, XЄ(0,1)

Xa(a,j)= alternative machine matrix

Z(k,j)= production times of parts by each machine

Using these datas, the objective function is formed as;

$$obj(t) = (DZXB) + \left(\sum_{j=1}^{J} C(j) x \sum_{j=1}^{J} G(j) x \left(\frac{D(j)}{B_s} \right) \right) + \sum_{y=2}^{Y} W(y-1)$$

(**Equation 1**)

obj(t) = objective function value of iteration t

This equation consists of three parts. The first part is sum of production cost, the second part is sum of intercellular material handling costand the third part is sum of training cost for workers.

First, our algorithm calculates the manufacturing cost of the system. This is shown as;

$$DZXB = D(j)x\left(\sum_{j=1}^{J}\sum_{k=1}^{K} Z(k,j)xX(k,j)\right)xB(k)$$

(**Equation 2**)

Manufacturing cost does not change by different cell formations. For each iteration of alternative part-machine matrix, manufacturing cost remains same. When we use different alternative for part-machine matrix, manufacturing cost changes.

Then algorithm calculates the inter-material handling cost for each cell formation. Each alternative part-machine matrix and each iteration for these matrixes generate different cost function. The steps for inter-material handling cost calculation isshown below;

1. Find which parts need which machines,
2. Check the machines, which cell that they belong to,
3. If they spread out n different cells, set the inter-material handling (n-1)
4. Multiply inter-material handling number for each part by demand and by unit inter-material handling cost. This equation is shown as;

$$Inter - MaterialHandlingCost = \left(handling(i)\right)xG(j)x\left(\frac{D(j)}{B_s}\right)$$

(**Equation 3**)

Inter-Material Handling cost changes by different cell formations.

Finally algorithm calculates the training cost for each cell formation. Each alternative part-machine matrix and each iteration for these matrixes generate different cost function as inter-material hadling cost. Because each combination of the part-machine-cell string cause different talent necessities. It should be determined that how complex machines are and the difference between this complexity and workers' talent degree. Then the cost is calculated by;

$$W_{total} = \sum_{y=2}^{Y} \sum_{v=1}^{V} W(v, y-1)$$

(**Equation 4**)

According to Pinedo (2004), the third step of a local search procedure is the search process within the neighbourhood after neighbourhood design. According to this process; the value, that is tried to be decreased by SA algorithm, is objective value. That value is also manufacturing system cost in this problem. Algorithm checks if the objective value is lower than or equal to the value that is decided to be reached. When we reach to that value, algorithm stops. Otherwise it changes the assignment of machines in cells. This is the acceptance-rejection criterion according to Pinedo (2004), which is the last step of the local search procedure. Neighbourhood is used to change parts and machines in cells by exchangement and mutation. Saeedi, Solimanpur, Mahdavi, & Javadian (2010) say that, some heuristic algorithms like Hill Climbing technique, may found the Local Optimum instead of the Global optimum because the movements leading to a new point worse than the current point are not allowed. SA algorithm allows to choose a worse result with a probability. This method helps to keep solution from local optimum. Because the goal is finding global optimum.

In the next step, an overview is done on the algorithm that is constructed for cellular manufacturing system.

3.1 The Proposed Algorithm And Components

In this part, detailed analyses are done for the problem, which is told above, and the poposed algorithm for this problem. The proposed algorithm tries to minimize the objective cost function with alternative routes and specified number of cells. This specified number is found by Kaiser's Rule. The total cost which is calculated for that specified number of cells, is tried to be minimized by the Simulated Annealing Algorithm. Before the detailed steps of algorithm, two methods are explained below which are used in this algorithm. The Kaiser's Rule and Simulated Annealing Algorithm.

3.1.1 The Kaiser's Rule

The Kaiser's Rule is an approach, which gives the most suitable number *p* for the system to divide. We can call it as *optimal cell number*. The Rule makes it by finding the most similar parts and machines.

There are many approaches to find the similarity between parts and machines which should be calculated to form cells. The similarity coefficient method is always prefered among these approaches. Because this method is easy to use and gives useful results. As told before, Jaccard's similarity coefficient approach is used in our algorithm. It is denoted by S_{ki}. This approach considers the relationship between parts and machines. But it doesn't consider this relationship as a traffic, which has a direction. The formulation is;

$$S(k,i) = \frac{\sum_{j=1}^{J} Xm_{jki}}{\sum_{j=1}^{J} \left(Ym_{ji} + Zm_{jk} - Xm_{jki} \right)} \qquad 1 < i,k < K$$

(**Equation 5**)

J = number of parts.

K = number of machines;

Xm_{jki} = 1 if part j has operation on both machines i and k, and 0 otherwise;

Ym_{ji} = 1 if part j has operation on machine i, and 0 otherwise;

Zm_{jk} = 1 if part j has operation on machine k, and 0 otherwise;

The similarity coefficients matrix is formed by S_{ki}'s. The matrix elements range from 0 to 1. According to the matrix theory, if the similarity coefficient matrix is real symmetric, it has n real eigenvalues. Moreover, the eigenvectors corresponding to these eigenvalues are linearly independent and each eigenvector represents a cell. These cells have low intercorrelations because the eigenvectors are uncorrelated, and therefore there should be low similarities between machines that are associated with different cells (Bashir & Karaa, 2008). This approach is simply called Kaiser's Rule. The equation is;

$$(S - \lambda I)\gamma = 0 \qquad \textbf{(Equation 6)}$$

I denotes; the identity matrix,

S denotes; the similarity coefficient matrix,

λ denotes; the root (eigenvalue) of the equation,

γ denotes; n eigenvector.

Kaiser's Rule says that the number of eigenvectors which are greater than 1 (one), both shows the number of cell in a system that should be and suitability of this system for the cellular manufacturing. If we have more than one eigenvector that fits to that condition, the system is suitable for a cellular manufacturing.

Kaiser's Rule is used with Simulated Annealing Algorithm in this thesis. Instead of trying different cell number alternatives, the right number is given to the problem by Kaiser's Rule. This approach made the algorithm easier and faster to reach to the feasible solution. In our study, it was seen that giving an optimal number (the value that is found by Kaiser's Rule) to the problem as a cell number returned a lower cost(objective function) with same number of iterations. Simulated Annealing Algorithm is explained in the next part.

3.1.2 The Simulated Annealing Algorithm

Simulated annealing (SA) is one of metaheuristics that have been used extensively to solve combinatorial optimization problems. By simulating the phenomenon that takes place in the cooling of pure substances from the liquid to the solid state, SA improves a solution to an optimization problem gradually until it finds the best solution in the search space (Kirkpatrick, Gelatt, & Vecchi, 1983). In each iteration, the algorithm accepts a randomly generated solution in the neighborhood of the current solution directly if it is better or probabilistically if it is worse. So it can be seen that not only better results are accepted, but also worse results are accepted. This provides global optimum solution to the algorithm.

Heuristic methods are formed by simulating the nature. Problem solutions are derived from these simulated systems. Saeedi, Solimanpur, Mahdavi, & Javadian (2010) say that the Simulated Annealing algorithm is derived from metallurgy and thermodynamics which incorporated a temperature parameter into the minimization parameter. A high temperature expands the search space, and with a lower temperature the search space gets smaller. The procedure starts from a high temperature and ends at a low temperature. At each temperature, a number of iterations are done.

It was told that SA algorithm worked by global search algorithm. The basis of global seach algorithm is local search algorithm. After the examination of local search procedure, the main difference is explained below.

Pinedo (2004) says that local search procedures can be compared by the following criterias;

1. The schedule representation needed for the procedure.

2. The neighbourhood design.
3. The search process within the neighbourhood.
4. The acceptance-rejection criterion.

Pinedo (2004) says that the design of the neighbourhood is a very important aspect of a local search procedure. Because it is the main point of algorithms. The coding the system for the neighbourhood and using it in solution steps should be well-calculated. This procedure may be easy for small models, but careness is needed for more complex ones.

We can view a literature survey with local optimization by Wu, Chang, & Chung (2008). They used neighbourhood calculations, which are the basis of SA modelling. They checked performance of their solutions by grouping efficacy calculations. In improvement studies, two types of movements were used for neighbourhood calculations to decrease the objective function value. First one is, just one part may change its cell; second one is, two parts may change their cells simultaneously. This may faster improve the system.

The simulated annealing algorithm gives result at the end of some iterations. The result of each iteration is an input of the next one. Pinedo (2004) explained the formulation. S_k is the solution of iteration k. The best solution found is S_0 at that time. $G(S_k)$ and $G(S_0)$ states the objective function of these solutions. Also $G(S_0)$, is used as comparison criteria. If the result which is found in an iteration, is better than this value, then it is accepted as the new solution. If not, then it is accepted by some probability or not. This probability is calculated as ;

$$P(S_0, S_k)=\exp\left\{\frac{\left(G(S_0)-G(S_k)\right)}{\beta_0}\right\} \qquad \textbf{(Equation 7)}$$

with probability 1 - $P(S_0, S_k)$ schedule S_k is rejected and $S_{k+1} = S_k$. Best solution S_0 does not change because it is better than S_k. The

$$\boldsymbol{\beta_1 \geq \beta_2 \geq \beta_3 \geq \beta_4 \geq \beta_5 \geq \cdots \geq 0} \qquad \textbf{(Equation 8)}$$

are cooling parameters or temperatures which are used to get the solution space smaller.

As shown in equation 8, cooling parameter gets lower by each iteration. This makes the solution space smaller, and the probability of acceptance of the worse result, lower.

Pinedo (2004) presents the SA algorithm as below. Two different stopping criteria are used for the algorithm; specified number of iterations or stopping when no improvement occurs. The algorithm below, uses the first criteria.

The algorithm is formed on calculation, comparison and optimization of the objective function as explained above, in this study. Algorithm starts by checking input datas. An itial solution should be given to the algorithm to start. Kaiser's Rule is used for this initial solution. It is an effective and useful method as a start to find the optimal solution. Kaiser's Rule gives the optimal number of cells that should be formed in the problem. This number is used to form cells on the initial solution. Algorithm first starts calculations by this initial solution. These calculations give inputs for the objective function. It was told that alternative routes existed in this problem. The algorithm runs for each alternative route for a specified number of iterations. And algorithm runs for same number for cell configuration that is given by Kaiser's Rule on same route.

Then each result for iteration number (t-1), is an input for iteration (t). It was told that the objective function consisted of three parts. The first part is sum of production cost, the second part is sum of intercellular material handling cost and the third part is sum of training cost for workers. The sum of these costs is compared with a big number and the neighbourhood solution is created according to the result. This cycle turns till the last iteration number. The best result among overall iterations (which alternative routes are included) is chosen as the solution of problem. The comparison is done by simulated annealing algorithm.

The algorithm is;

Begin

Step 1.

Set k = 1 and select β_1.

Select an initial sequence S_1 using some heuristic.

Set $S_0 = S_1$.

Step 2.

Select a candidate schedule Sc from the neighbourhood of S_k.

If $G(S_0) < G(S_c) < G(S_k)$, set $S_{k+1} = S_c$ and go to Step 3.

If $G(S_c) < G(S_0)$, set $S_0 = S_{k+1} = S_c$ and go to Step 3.

If $G(S_c) > G(S_k)$ generate a random number U_k from a Uniform(0,1) distribution;

If U_k . $P(S_k, S_c)$ set $S_{k+1} = S_c$ otherwise set $S_{k+1} = S_k$ and go to Step 3.

Step 3.

Select $\beta_{k+1} \leq \beta_k$.

Increment k by 1.

If k = N then STOP, otherwise go to Step 2. (Plaquin & Pierreval, 2000)

According to these definitions, the constructed and proposed algorithm is explained in the next part.

3.1.3 The Proposed Algorithm

The proposed algortihm for solving the problem is constructed as shown below.

Set *nnn=0*

Step.1 If nnn $<nn^{np}$Set *nnn=nnn+1* and generate alternative part-machine matrix, else stop.

Step.2 Calculate the Similarity Coefficients of each machine

Step.3 Apply the Kaiser's Rule to find the optimal cell number

Step.4 Build the initial part-machine-cell matrix PMC_i

Step.5 Set *iteration=1:iterationmax*

Step.6 Calculate the manufacturing cost *DZXB* according to alternative part-machine matrix

Step.7 Calculate the inter-cellular material handling cost *handling(i)* according to traffic between cells for each part

Step.8 Find training cost *W(i)* if training needed in cells and add to the total training cost *Wtotal*

Step.9 Calculate the objective function *obj(iteration)*

Step.10 If it is lower than predetermined value (*BN*), accept the *obj(iteration)*, assign new value as *BN*, and generate new part-machine-cell matrix solution

Step.11 If not, reject the solution and generate new part-machine-cell matrix solution from former iteration or accept the solution according to a probability value of SA and generate a new part-machine-cell matrix solution.

Step.12 If iteration < iterationmax, go to step 5, else go to step 1.

The algoritm starts by taking the input datas. All steps run by these input datas. Input datas are formed from quantitative datas and matrixes. Some of them are part-machine matrix, alternative manufacturing matrix, demand matrix, matrix production times, matrix of machine production cost, matrix of inter-cell material handling cost, worker training cost matrix, worker talent matrix (randomly distributed) and so on.

First, algorithm generates the alternative part-machine matrix by alternative routes given as an input. After specified number of iterations, algorithm generates the next alternative part-machine matrix and run for same number of iterations.

The next step is Kaiser's Rule. To learn the optimal number of cells, similarity of machines are calculated and Kaiser's Rule finds the optimal value according to these similarities. This optimal number is not change till the next alternative route. This means, it is used same number of cells for sprecified number of iterations.

Manufacturing cost (*DZXB),* inter-cellular material handling cost (*handling(i))*and training cost (*W(i))* is calculated for each iteration according to alternative part-machine matrix. The sum of these costs is objective function *(obj(iteration)).*

The simulated annealing procedure starts after objective function calculation. If the objective value, that is calculated in this iteration, is lower than the predetermined value (a big number), the new solution is generated from that iterations solution. If it is higher, two options exist. If the probability value is higher than the randomly generated value, the same procedure is done as in lower objective function value than the predetermined value. If this is not occured, the result of earlier iteration is used for a new solution generation.

This continues till the maximum iteration value. At the end, the lowest objective function value and its datas are taken from overall results. This is the result of that route. If any other routes exist, algorithm continues for them and goes to the beginning of algorithm. This cycle turns for nn^{np} times, which *nn* denotes; alternative route number for one part, *np* denotes; the number of parts that have alternative routes.

3.1.4 An Illustrative Example

The steps and details of the proposed algorithm are explained in previous parts. A numerical example for a manufacturing system with 5 parts and 3 machines is shown below to understand better how algorithm works.

Set *nnn=0*
Step.1 If nnn $<nn^{np}$Set *nnn=nnn+1* and generate alternative part-machine matrix, else stop.

nnn=1

$$X_p=\begin{bmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 \end{bmatrix} \quad ;$$

Step.2 Calculate the Similarity Coefficients of each machine

$$S = \begin{bmatrix} 1 & 0 & 0.25 \\ 0 & 1 & 0.3333 \\ 0.25 & 0.3333 & 1 \end{bmatrix};$$

Step.3 Apply the Kaiser's Rule to find the optimal cell number

$$C = \begin{bmatrix} 0.5833 \\ 1.0000 \\ 1.4167 \end{bmatrix};$$

Number of values that are higher than 1 is two. That means Kaiser's Rule offers 2 cells for the manufacturing system.

Step.4 Build the initial part-machine-cell matrix PMC_i

$$PMC_i = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 1 & 2 & 3 & 0 & 0 \end{bmatrix}$$

Step.5 Set *iteration=1:iterationmax*

iteration=1

Step.6 Calculate the manufacturing cost *DZXB* according to alternative part-machine matrix

DZXB = 815

Step.7 Calculate the inter-cellular material handling cost *handling(i)* according to traffic between cells for each part

handling= 20

Step.8 Find training cost *W(i)* if training needed in cells and add to the total training cost *Wtotal*

W_{total}= 2200

Step.9 Calculate the objective function *obj(iteration)* (*BN=1.000.000*) *obj* = 3035

Step.10 If it is lower than predetermined value, accept the *obj(iteration)* and generate new part-machine-cell matrix solution
3035<1.000.000

$PMC_i = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 0 & 0 & 1 & 3 \end{bmatrix};$

Step.11 If not, reject the solution and generate new part-machine-cell matrix solution from former iteration or accept the solution according to a probability value of SA and generate new part-machine-cell matrix solution.

Step.12 If iteration < iterationmax, go to step 5, else go to step 1.
iteration =1
go to step 1.

This example is the first iteration of first route. Algorithm goes on working in this presentation. At the end of all iterations of all routes, the minimum value of objective function and its datas are choosen by algorithm.

CHAPTER FOUR
ANALYZING THE COMPUTATIONAL RESULTS

The cellular manufacturing problem and its solution method, which are studied in this thesis, were explained in previous section. Matlab R2008a version is used to run the proposed algorithm for solving the problem. This study is executed on an Intel Centrino Duo, 1.73 Ghz, and Windows 7, using 1.5 GB RAM. Computations are done for different numerical combinations and different sized matrices of datas. In this chapter, computational results of numerical examples are shown, which are derived for a system with 5 machines and 7 parts, 10 machines and 15 parts, 18 machines and 30 parts.

In this part, cell formation is done which cells include workers, machines and parts, by minimizing the objective cost function. In the problems, machine complexity levels and worker talent levels are included. Each cell has one stable worker who's talent degree is randomly assigned. If machines' complexities are higher than worker's talent in a cell, then this worker is trained to operate these machines. Inter-cell material handling cost is also included in the problem. Number of cells is not specified but with part-machine matrix, optimal number of cells is calculated by Kaiser's Rule. This makes the algorithm faster and provides an advantage on forming cells. A part can go alternative machines (different routes) to be operated for the same operation. A worker can operate more than one machine. This is called multi-functionality. And cost of training depends on complexity levels. A single period is studied in this thesis. The table below, elaborates on the characteristics of problem.

Several numerical examples are solved for each of 5x7, 10x15 and 18x30 sized problems to see and analyze the numerical results. This is called Sensitivity Analyse, and done for each problem to see the effects of objective function cost components on the total cost. These components are; intercellular-material handling cost and training cost. Intercellular-material handling costs and training costs are used for different level of values, to see their effects on results. Detailed analyze of computational results are shown in next sections.

Table 4.1 Characteristics of the problems

Number of machines	specified	5-10-18
Number of parts	specified	7-15-30
Machine complexity degree	specified	Max 3
Worker talent degree	specified	Max 3
Worker talent degree upgrading	by training	Max 2
Period capacity of the problem	Single period	
Material handling cost	Specified	
Number of cells	Unspecified(Calculated by Kaiser's Rule)	
Number of workers	Unspecified(equal to the number of cells)	Equal to the number of cells
Multi-functionality of machines	Enable	A machine can produce several parts
Alternative routing	Enable	
Multi-functionality of workers (trainig is based on the most complex machine in the cell)	Enable(one worker can operate more than one machine)	
Training	Enable (takes 0 time unit)	
Productivity of workers	Trained worker's productivity is equal to the trueborn talented	
Training cost	Changes by talent level	

4.1 Computational Results For 5x7 Sized Part-Machine Matrix

In this part, a manufacturing area is studied with 5 machines and 7 parts. The solutions for all alternative routes are analyzed. Several examples are shown below for different cost component level combinations. A sensitivity analyse is done by these combinations. It can be seen that The Kaiser’s Rule divided the manufacturing area to two cells for each example. Different inputs will form different solutions.

First, we can analyze the table below, which shows results for 8 alternative part-machine matrix of our algorithm with 5 machines and 7 parts.

Numerical results are shown in Table 4.2. We have 8 alternative part-machine matrices, according to the alternative routes. Each coloumn shows the alternative part-machine matrix results. Maximum number of iterations specified as 200 for each alternative matrix which means the the algorithm runs for 200x8=1600 times. Each iteration uses the result of previous one, as an input. In other words, algorithm improves results by each iteration. (Also when algorithm chooses the worse result.) “Minimum objective function value” means the minimum value within the each 200 iterations for each alternative route. We told that each iteration used the result of previous one, as an input. And also we told that Simulated Annealing Algorithm sometimes chose the worse result, not to stuck into local minimum. Within this coloumn, the minimum value is shown that is found in successive iterations. For example, the minimum objective value that is achieved within 200 iterations is *8675* by the 2nd route. Cell number is the number which is found by the Kaiser’s Rule with the inputs given below. Kaiser’s Rule finds the cell number at the beginning of the algorithm, and all iterations for the same route base on this number. In table 4.2, it can be seen that the optimal cell number for the inputs given below, is 2. And “Number of inter-material handling” shows the inter-cellular movements of the manufacturing system within that cell number. For example, algorithm gave 5 times material handling in the system at the end of 200 iterations which costed 210 for the 2nd route. But with same cell number, in the 1st route, algorithm gave 0 material handling in the system at the end of 200 iterations which costed 0. This means, algorithm formed a system with 2 cells for all alternative routes but cells had different combinations of machines and parts because of the different routes.

The input datas for this problem are explained below.

The part-machine incidence matrix is (rows are machines);

$$X_p=\begin{bmatrix}0&1&1&0&0&1&1\\1&0&1&0&1&1&0\\1&1&0&1&0&1&0\\0&0&0&1&1&0&0\\1&0&1&0&1&0&1\end{bmatrix}$$

Alternative machine matrix, which forms the alternative routes, is shown as X_a. Rows are parts and coloumns are machines.

$$X_a=\begin{bmatrix}0&0\\0&0\\2&4\\0&0\\2&5\\1&3\end{bmatrix}$$

Production times of parts at each machine are shown in matrix Z. Rows are machines and coloumns are parts.

$$Z=\begin{bmatrix}1&3&2&3&1&4&3\\3&5&5&1&4&3&2\\7&2&2&4&1&1&3\\1&4&2&3&6&3&4\\8&5&1&1&4&6&3\end{bmatrix}$$

The temperature of Simulated Annaeling Algorithm probability equation is shown as;

$Te=10^8$

The demands of parts are shown as;

D_p=[10 15 5 20 35 15 30]

The operating costs of machines per time unit are;

$$B=\begin{bmatrix}8\\5\\9\\7\\6\end{bmatrix}$$

The batch size of a part is;

Bs=5;

The inter-cell material handling costs for each part are shown as;

G=[10 15 20 10 15 15 10];

Costs of worker training skips between levels 1-2 and 3 are;

W=[1000 600];

Workers' talent levels are;

T=[2 2 3 1 2 2 1 3 3 2]

Part-machine-cell matrix as an initial solution in the problem;

$$PMC=\begin{bmatrix}1 & 2 & 3 & 4 & 5 & 6 & 7\\1 & 2 & 3 & 4 & 5 & 0 & 0\end{bmatrix}$$

Machines' complexity levels are;

M=[2 3 1 3 2];

Our algorithm found the minimum value as 8675 cost unit with 5 times carriage and no training in overall iterations and routes. The rest routes have different combinations of training and inter-material handling and costs of them.

We can analyze some of runs to be able to analyze the whole system.

The details of first route are shown in table 4.3. We can see that algorithm divided the system to 2 cells and complexity level is 3 for both cells. *1200* unit cost spent to make workers suitable for cells. This is the training cost. It can be seen that system doesn't need to any inter-material handlings. And manufacturing cost has the main portion which costed *8375*.

Table 4.2 Computational results of 5x7 part-machine matrix

Alternative matrix	***1***	***2***	***3***	***4***	***5***	***6***	***7***	***8***	***min***	***average***
Minimum Objective function value	9575	8675	8840	8990	9920	10010	8785	8935	8675	9216,25
Cost of inter-material handling	0	210	150	210	0	0	150	210	210	116,25
Cost of manufacturing	8375	8465	8690	8780	8320	8410	8635	8725	8465	8550
Cost of training	1200	0	0	0	1600	1600	0	0	0	550
Number of training	2	0	0	0	2	2	0	0	0	0,75
Number of cell	2	2	2	2	2	2	2	2	2	2
Number of inter-material handling	0	5	4	5	0	0	4	5	5	2,875

The part-machine-cell matrix of the 1st route is shown below as *PMC_i*.

$$PMC_i = \begin{bmatrix} 2 & 4 & 6 & 1 & 3 & 5 & 7 \\ 0 & 1 & 2 & 3 & 4 & 5 & 0 \end{bmatrix}$$

$$A = \begin{bmatrix} 2 & 1 & 2 & 1 & 2 & 1 & 2 \\ 1 & 1 & 2 & 2 & 2 & 1 & 2 \end{bmatrix}$$

Table 4.3 Components of the 1st route

minimum objective function value	complexity level of cell 1	complexity level of cell 2
9575	3	3

8375	0	1200
manufacturing cost	material handling cost	training cost

It can be seen that manufacturing area has two cells in Table 4.2. It can be also seen in matrix A (cell number matrix). This matrix shows the cell numbers that parts and machines belong to. These two matrices show that the parts 2-4-6 and machine 1-2 are in cell 1, parts 1-3-5-7 and machines 3-4-5 are in cell 2.

The details of 2nd route are shown in table 4.4. This is the minimum costed route. We can see that material handling cost is *210* with *5* times inter-cell movement. And 0 unit training cost occurs. Manufacturing cost is different from first route, because different parts use different machines (alternative machines-routes). Complexity level is 2-3 for cells.

Table 4.4 Components of the 2nd route

minimum objective function value	complexity level of cell 1	complexity level of cell 2
8675	3	2

8465	210	0
manufacturing cost	material handling cost	training cost

The part-machine-cell matrix of the 2nd route is shown below as PMC_i;

$$PMC_i = \begin{bmatrix} 2 & 4 & 6 & 1 & 3 & 5 & 7 \\ 3 & 0 & 0 & 1 & 2 & 4 & 5 \end{bmatrix}$$

$$A = \begin{bmatrix} 2 & 1 & 2 & 1 & 2 & 1 & 2 \\ 2 & 2 & 1 & 2 & 2 & 1 & 1 \end{bmatrix}$$

We used Kaiser's Rule in our algorithm because we wanted to see how suitable our system was for a cell formation problem. When we run our algorithm without the Kaiser's Rule and with the manually specified cell number (as 3 cells), our algorithm found the minimum value as *9310* cost unit in overall iterations. We can see the difference from results in Table 4.2. This also means that more inter-material handling occurs. This is a worse result than our result with Kaiser's Rule. This shows the Kaiser's Rule's improvement effect on our problem.

4.1.1 Analysis Of Inter-Material Handling Cost

We can analyze one more example to check the sensitivity of our system with 5 machines and 7 parts. We change some of inputs to see how our system reacts. Inter-material handling cost is multiplied by 4 in this example.

The input datas for Table 4.5 are same as datas for Table 4.2. The only difference is;

G=4x[10 15 20 10 15 15 10];

Table 4.5 Computational results of 5x7 part-machine matrix

Alternative matrix	***1***	***2***	***3***	***4***	***5***	***6***	***7***	***8***	***min***	***average***
Minimum Objective function value	9575	9305	9290	9380	9920	10010	9235	9565	9235	9535
Cost of inter-material handling	0	840	600	0	0	0	600	840	600	360
Cost of manufacturing	8375	8465	8690	8780	8320	8410	8635	8725	8635	8550
Cost of training	1200	0	0	600	1600	1600	0	0	0	625
Number of training	2	0	0	1	2	2	0	0	0	0,875
Number of cell	2	2	2	2	2	2	2	2	2	2
Number of inter-material handling	0	5	4	0	0	0	4	5	4	2,25

It can be seen in the Table 4.5 that, algorithm gave no material handling in the system at the end of 200 iterations for the first route. But with same cell number, in the second route, algorithm gave 5 times inter-material handlings in the system at the end of 200 iterations which costed *840*. These two combinations are same as in Table 4.2. When we redound the cost of inter-material handling, algorithm enforces the alternative routes of the system to make less inter-material handling and more training according to that decreasement. We can see that the fourth route changed its combination by the decision of having training instead of carriage. Because the one time training could do the job of 5 times carriage which would cost 4 times higher than the value in Table 4.2. The minimum value of whole system is came up with seventh route, which was second route in Table 4.2. We can see that the algorithm changed its decision by change of carriage costs.

We can analyze one more example to check the sensitivity of our system with 5 machines and 7 parts. Inter-material handling cost is multiplied by 8 in the next example.

The input datas for Table 4.6 are same as datas for Table 4.2. The only difference is;

G=8x[10 15 20 10 15 15 10];

Table 4.6 Computational results of 5x7 part-machine matrix

Alternative matrix	***1***	***2***	***3***	***4***	***5***	***6***	***7***	***8***	***min***	***average***
Minimum Objective function value	9575	9665	9690	9380	9920	10010	9635	9725	9380	9700
Cost of inter-material handling	0	0	0	0	0	0	0	0	0	0
Cost of manufacturing	8375	8465	8690	8780	8320	8410	8635	8725	8780	8550
Cost of training	1200	1200	1000	600	1600	1600	1000	1000	600	1150
Number of training	2	2	1	1	2	2	1	1	1	1,5
Number of cell	2	2	2	2	2	2	2	2	2	2
Number of inter-material handling	0	0	0	0	0	0	0	0	0	0

We can see in Table 4.6, that the cost combination of 2nd,3th,7th and 8th routes are changed when they are compared with values in Table 4.5. When we redound the cost of inter-material handling more, algorithm enforces the system to make less inter-material handling once again. We can see that the number of inter-material handling of all routes in Table 4.6 is 0 (zero). This decreasement caused an increasement on number of training for these routes in the whole system, because at the end of comparison of training and increased inter-material handling costs, the algorithm chose the least costed trainings. Because system should use either inter-material handlings or worker trainings to compose cells. For example in eighth route, algorithm chose the 1000 unit cost of one time training instead of 840x2 unit cost of 5 times carriage. If inter-material handlings are decreased, then training has more importance and responsibility. Also the minimum value came up with fourth route with 0 (zero) inter-material handling, in this example. We can say that the system is non-sensitive to inter-material handling cost increasement from now on, because the system eliminated all inter-material handlings in the system.

4.1.2 Analysis 0f Training Cost

We can analyze one more example to check the sensitivity of our system with 5 machines and 7 parts. It is said that we changed some of inputs to see how our system reacts. In this example Inter-material handling cost is same as first example, training cost is multiplied by 2.

The input datas for Table 4.7 are same as datas for Table 4.2. The only difference is;

W=2x[1000 600];

We can see in Table 4.7 that the cost combination of first route is changed when it is compared with values in Table 4.2. Algorithm chose the 1250 unit cost of 9 times carriage instead of 2000 unit cost of one time training for this route. In Table 4.2, 6 times training is done in the whole system, but it is four in this example. When we redound the cost of training, algorithm enforces the routes of the system to make less training. Because of that algorithm makes the number of inter-material handling higher in some alternative routes. But some routes (5th and 6th) accept to endure the training cost instead of having carriage. Training cost and inter-material handling cost are inversely proportional cost componenets. It is seen that the second route has minimum value in Table 4.2, it is again second route that has minimum value in Table 4.7. Because system already made training cost of some alternative routes 0 (zero). Because of small size of the system, some changes will not show any effects on system.

Table 4.7 Computational results of 5x7 part-machine matrix

Alternative matrix	***1***	***2***	***3***	***4***	***5***	***6***	***7***	***8***	***min***	***average***
Minimum Objective function value	9625	8675	8840	8990	11520	11610	8785	8935	8675	9747,5
Cost of inter-material handling	1250	210	150	210	0	0	150	210	210	272,5
Cost of manufacturing	8375	8465	8690	8780	8320	8410	8635	8725	8465	8550
Cost of training	0	0	0	0	3200	3200	0	0	0	800
Number of training	0	0	0	0	2	2	0	0	0	0,5
Number of cell	2	2	2	2	2	2	2	2	2	2
Number of inter-material handling	9	5	4	5	0	0	4	5	5	4

We can analyze one more example to check the sensitivity of our system with 5 machines and 7 parts. In this example Inter-material handling cost is same as first example, training cost is multiplied by 3.

The input datas for Table 4.8 are same as datas for Table 4.2. The only difference is;

W=3x[1000 600];

It can be seen in Table 4.8 that, that the cost combination of 5th route is changed when it is compared with values in Table 4.7. In Table 4.7, 4 trainings are done in the whole system. But it is turned to 3 in Table 4.8. Because the algorithm chose the 1050 unit cost of 8 times carriage instead of 1800 unit cost of one time training for the 5th route. We can say when we redound the cost of training, algorithm enforces the system to make less training. As in Table 4.7, algorithm makes the number of inter-material handling higher again in some routes(5th). But the alternative route, which has the minimum value, is not changed. The changement of unit costs, could not change the decision but changed the whole system's structure. We can say that the system is sensitive to cost changes of training but the second route is dominant and makes the system to act as nonsensitive. The effect of training cost changements in the manufacturing system are not clearly seen, because of the system's small size. Also the reason is, system could make the training costs 0 (zero) at the beginning and without any changes on costs. The system goes on with same choises for higher training costs than 3xW. That means system gets nonsensitive for higher costs than 3xW.

Table 4.8 Computational results of 5x7 part-machine matrix

Alternative matrix	***1***	***2***	***3***	***4***	***5***	***6***	***7***	***8***	***min***	***average***
Minimum Objective function value	9625	8675	8840	8990	12370	13210	8785	8935	8675	9966,25
Cost of inter-material handling	1250	210	150	210	1050	0	150	210	210	441,25
Cost of manufacturing	8375	8465	8690	8780	8320	8410	8635	8725	8465	8550
Cost of training	0	0	0	0	3000	4800	0	0	0	975
Number of training	0	0	0	0	1	2	0	0	0	0,375
Number of cell	2	2	2	2	2	2	2	2	2	2
Number of inter-material handling	9	5	4	5	8	0	4	5	5	5,25

The part-machine-cell matrix of Table 4.8, 8th route is shown in table 4.9. Cell 1 has 3 machines, cell 2 has 2 machines. Machines 1-3-4 are in cell 1, machines 2-5 are in cell 2. This configuration has 5 intercell movements between cell 1 and cell 2 as shown in table 4.8.

Our problem that is based on Simulated Annealing Meta-heuristic has a 5x7 sized part machine matrix. Two cells have two workers (one for each), who are trained as hard as the most complex machine in that cells. An example with higher capacity should be analyzed to see the effects. It is analyzed a problem with 10x15 sized part machine matrix in the next part.

Table 4.9 Part-machine-cell matrix

	cell 1			cell 2			
machines	1	3	4	2	5	0	0

4.2 Computational Results For 10x15 Sized Part-Machine Matrix

In this part, a manufacturing area is studied with 10 machines and 15 parts. Several examples are shown below for different cost component level combinations. A sensitivity analyse is done again by these combinations as previous part.

First, we can analyze the table below, which shows results for 8 alternative part-machine matrix of our algorithm with 10 machines and 15 parts.

Numerical results are shown in Table 4.10. To be able to analyze the system we can check these results. For example, the minimum objective value that is achieved within 200 iterations is *40580* by the 3rd route. Cell number is *3*, which is found by the Kaiser's Rule with the inputs given below. Algorithm gave 14 times material handling in the system at the end of 200 iterations which costed *675* for the 3rd route. However with same cell number, in the 1st route, algorithm gave *28* material handlings in the system at the end of 200 iterations which costed *1365*. This means, algorithm formed a system with 3 cells for all alternative routes but cells had different combinationsof machines and partsbecause of the different route existance.

The input datas for this problem are explained below.

Alternative machine matrix, which forms the alternative routes, is shown as X_a. Rows are parts and coloumns are machines.

X_a= [0 0;0 0;2 6;0 0;0 0;2 5;0 0;0 0;1 8;0 0;0 0;0 0;0 0;0 0;0 0]

The part-machine incidence matrix is (rows are machines);

X_p=

$$\begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

Production times of parts at each machine are shown in matrix Z. Rows are machines and coloumns are parts.

Z=

$$\begin{bmatrix} 1 & 3 & 2 & 3 & 1 & 4 & 3 & 7 & 2 & 2 & 4 & 1 & 1 & 3 & 2 \\ 3 & 5 & 5 & 1 & 4 & 3 & 2 & 3 & 6 & 3 & 4 & 1 & 3 & 2 & 3 \\ 7 & 2 & 2 & 4 & 1 & 1 & 3 & 5 & 1 & 1 & 6 & 3 & 4 & 7 & 2 \\ 1 & 4 & 2 & 3 & 6 & 3 & 4 & 5 & 1 & 4 & 3 & 2 & 3 & 6 & 3 \\ 8 & 5 & 1 & 1 & 4 & 6 & 3 & 2 & 2 & 4 & 1 & 9 & 3 & 5 & 1 \\ 3 & 4 & 1 & 3 & 2 & 3 & 4 & 1 & 1 & 3 & 5 & 1 & 1 & 6 & 4 \\ 3 & 2 & 3 & 6 & 3 & 4 & 1 & 3 & 2 & 3 & 3 & 7 & 1 & 9 & 7 \\ 5 & 1 & 1 & 4 & 6 & 8 & 5 & 1 & 4 & 1 & 9 & 3 & 5 & 5 & 4 \\ 4 & 2 & 3 & 6 & 3 & 4 & 5 & 1 & 4 & 3 & 2 & 2 & 4 & 1 & 1 \\ 3 & 5 & 5 & 8 & 5 & 1 & 4 & 1 & 6 & 3 & 4 & 5 & 1 & 2 & 2 \end{bmatrix}$$

The temperature of Simulated Annaeling Algorithm probability equation is shown as;

$Te=10^8$;

The demands of parts are shown as;

D_p=[10 15 5 20 35 15 30 20 15 10 30 20 15 10 15] ;

The operating costs of machines per time unit are;

B= [8; 5; 9; 7; 6; 7; 9; 10; 4; 11] ;

The batch size of a part is

Bs=5;

The inter-cell material handling costs for each part are shown as;

G=[10 15 20 10 15 15 10 10 15 20 15 20 10 25 5];

Costs of worker training skips berween levels 1-2 and 3 are;

W=[1000 600];

Workers' talent levels are;

T= [1 2 3 1 2 2 1 3 3 2]

Part-machine-cell matrix as an initial solution in the problem;

$$PMC=\begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Machines' complexity levels are;

M=[2 3 3 3 2 2 3 2 3 3]

As an example, the details of sixth route are shown in table 4.11. We can see that complexity levels are 2-2-3 for cells. *1000* unit cost spent to make workers suitable for cells. This is the training cost. It can be seen that system needs to *15* inter-material handlings. And manufacturing has the main portion which costed *39860*.

The part-machine-cell matrix of the result that is shown in Table 4.11 is shown after the table. The matrix *A*, shows the cell numbers that parts and machines belong to. Matrix *A* and PMC_i show that the parts 3-6-9-12-15 and machine 5-8 are in cell 1, parts 2-5-8-11-14 and machines 1-2-3-6-9 are in cell 2, the parts 1-4-7-10-13 and machine 4-7 are in cell 3.

Table 4.10 Computational results of 10x15 part-machine matrix

Alternative matrix	***1***	***2***	***3***	***4***	***5***	***6***	***7***	***8***	***min***	***average***
Minimum Objective function value	40955	41675	40580	42290	41440	41585	41880	40840	40580	41405,625
Cost of inter-material handling	1365	725	675	1025	940	725	1065	665	675	898,125
Cost of manufacturing	39590	39950	39905	40265	39500	39860	39815	40175	39905	39882,5
Cost of training	0	1000	0	1000	1000	1000	1000	0	0	625
Number of training	0	1	0	1	1	1	1	0	0	0,625
Number of cell	3	3	3	3	3	3	3	3	3	3
Number of inter-material handling	28	15	14	21	20	15	22	14	14	18,625

Table 4.11 Components of the 1st route

minimum objective function value	complexity level of cell 1	complexity level of cell 2	complexity level of cell 3
41585	2	2	3

39860	725	1000
manufacturing cost	material handling cost	training cost

$$PMC_i = \begin{bmatrix} 3 & 6 & 9 & 12 & 15 & 2 & 5 & 8 & 11 & 14 & 1 & 4 & 7 & 10 & 13 \\ 5 & 8 & 0 & 0 & 0 & 0 & 1 & 2 & 3 & 6 & 9 & 10 & 4 & 7 & 0 \end{bmatrix}$$

$$A = \begin{bmatrix} 3 & 2 & 1 & 3 & 2 & 1 & 3 & 2 & 1 & 3 & 2 & 1 & 3 & 2 & 1 \\ 2 & 2 & 2 & 3 & 1 & 2 & 3 & 1 & 2 & 2 & 3 & 1 & 1 & 1 & 1 \end{bmatrix}$$

The details of third route are shown in table 4.12. This is the minimum costed route. We can see that material handling cost is *675* with 14 times inter-cell movement. And 0 unit training cost occurs. Manufacturing cost is different from first route, because different parts use different machines (alternative machines). Complexity level is 3-2-2 for cells.

Table 4.12 Components of the 3rd route

minimum objective function value	complexity level of cell 1	complexity level of cell 2	complexity level of cell 3
40580	3	2	2

39905	675	0
manufacturing cost	material handling cost	training cost

The part-machine-cell matrix of the result that is shown in Table 4.12 is shown below;

$$PMC_i = \begin{bmatrix} 3 & 6 & 9 & 12 & 15 & 2 & 5 & 8 & 11 & 14 & 1 & 4 & 7 & 10 & 13 \\ 0 & 0 & 0 & 10 & 1 & 2 & 4 & 5 & 8 & 9 & 0 & 3 & 6 & 7 & 0 \end{bmatrix}$$

$$A = \begin{bmatrix} 3 & 2 & 1 & 3 & 2 & 1 & 3 & 2 & 1 & 3 & 2 & 1 & 3 & 2 & 1 \\ 2 & 2 & 3 & 2 & 2 & 3 & 3 & 2 & 2 & 1 & 3 & 1 & 1 & 2 & 1 \end{bmatrix}$$

These two matrices show that the parts 3-6-9-12-15 and machine 10 are in cell 1, parts 2-5-8-11-14 and machines 1-2-4-5-8-9 are in cell 2, the parts 1-4-7-10-13 and machine 3-6-7 are in cell 3.

We used Kaiser’s Rule in our algorithm because we wanted to see how beneficial our system was for a cell formation problem. When we run our algorithm not with the Kaiser’s Rule but with the manually specified cell number as 2 and 4 cells, our algorithm found the minimum value as *41225* cost unit for the system with 2 cells, and *41040* cost unit for the system with 4 cells in overall iterations. We can see the difference from results in Table 4.10.

4.2.1 Analysis Of Inter-Material Handling Cost

We can analyze one more example to check the sensitivity of our system with 10 machines and 15 parts. We change some of inputs to see how our system reacts. Inter-material handling cost is multiplied by 4 in this example.

The input datas for Table 4.13 are same as datas for Table 4.10. The only difference is;

G=4x[10 15 20 10 15 15 10 10 15 20 15 20 10 25 5];

It can be seen in the Table 4.13 that, 1st, 2nd, 4th, 5th, 6th, 7th routes changed their combinations of training and inter-material handling by the decision of having training instead of carriage. Because, for example in first route the 3 times training could provide the job cheaper with 14 times carriage instead of just 28 times carriage which would cost 4 times higher than the value in Table 4.10. When we redound the cost of inter-material handling, algorithm enforces the system to make less inter-material handling and more training according to that decreasement. However the third and eighth routes show the same combinations as Table 4.10. And algorithm didn’t change the selection of alternative route. The minimum value of whole system is came up with third route again.

Table 4.13 Computational results of 10x15 part-machine matrix

Alternative matrix	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*min*	*average*
Minimum Objective function value	44370	42850	42605	44685	43920	42760	44915	42835	42605	43617,5
Cost of inter-material handling	2580	700	2700	2220	2220	700	2900	2660	2700	2085
Cost of manufacturing	39590	39950	39905	40265	39500	39860	39815	40175	39905	39882,5
Cost of training	2200	2200	0	2200	2200	2200	2200	0	0	1650
Number of training	3	3	0	3	3	3	3	0	0	2,25
Number of cell	3	3	3	3	3	3	3	3	3	3
Number of inter-material handling	14	7	14	12	12	7	15	14	14	11,875

Inter-material handling cost is multiplied by 8 in the next example, to see the effect better.

The input datas for Table 4.14 are same as datas for Table 4.10. The only difference is;

G=8x[10 15 20 10 15 15 10 10 15 20 15 20 10 25 5];

Table 4.14 Computational results of 10x15 part-machine matrix

Alternative matrix	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*min*	*average*
Minimum Objective function value	46310	43550	44715	46905	46140	43460	47815	45275	43460	45521,25
Cost of inter-material handling	3920	1400	4700	4440	4440	800	5800	3500	800	35625
Cost of manufacturing	39590	39950	39905	40265	39500	39860	39815	40175	39860	39882,5
Cost of training	2800	2200	600	2200	2200	2800	2200	1600	2800	2075
Number of training	4	3	1	3	3	4	3	2	3	2,875
Number of cell	3	3	3	3	3	3	3	3	3	3
Number of inter-material handling	12	7	10	12	12	5	15	11	5	10,5

We can see in Table 4.14, some routes changed their combinations of training and inter-material handling again by the decision of having training instead of carriage. When we redound the cost of inter-material handling more, algorithm enforces the system to make less inter-material handling once again. For example in first route the 1 more time training could provide the job cheaper with 12 times carriage instead of 3 times training and 14 times carriage which would cost 4 times higher carriage than the value in Table 4.10. But in the 6th route, 7 times carriage and 3 times training costs are equal to 5 times carriage and 4 times training cost, and algorithm chose to add one more training to the system.

Also the minimum value came up with sixth route, in this example. We can say that the system is sensitive to inter-material handling cost increasement, but we should analyse one more increasement to see the nonsensitive point.

It is analyzed if the inter-material handling cost is multiplied by 12 in the next example.

The input datas for Table 4.15 are same as datas for Table 4.10. The only difference is;

G=12x[10 15 20 10 15 15 10 10 15 20 15 20 10 25 5];

We can see in Table 4.15, that the cost combination of 7th route is changed when all values are compared with values in Table 4.14. When we redound the cost of inter-material handling more, algorithm enforces the system to make less inter-material handling once again. This decreasement caused an increasement on number of training for these routes in the whole system, because at the end of comparison of training and increased inter-material handling costs, the algorithm chose the least costed trainings. Because system should use either inter-material handlings or worker trainings to compose cells. For example in 7th route, algorithm chose the 1000 unit cost of one more time training and 13 times carriage instead of 5800x2 unit cost of 15 times carriage and 3 times training. If inter-material handlings are decreased, then training has more importance and responsibility. Also the minimum value came up with 6th route again, in this example. We can say that the system is sensitive to inter-material handling cost increasement till now but nonsensitive from now on, because the algorithm gives same results and makes same choise with higher values than *12xG*.

Table 4.15 Computational results of 10x15 part-machine matrix

Alternative matrix	***1***	***2***	***3***	***4***	***5***	***6***	***7***	***8***	***min***	***average***
Minimum Objective function value	48270	44250	47555	49125	48360	43860	50275	47025	43860	47340
Cost of inter-material handling	5880	2100	7050	6660	6660	1200	7260	5250	1200	5257,5
Cost of manufacturing	39590	39950	39905	40265	39500	39860	39815	40175	39860	39882,5
Cost of training	2800	2200	600	2200	2200	2800	3200	1600	2800	2200
Number of training	4	3	1	3	3	4	4	2	4	3
Number of cell	3	3	3	3	3	3	3	3	3	3
Number of inter-material handling	12	7	10	12	12	5	13	11	5	10,25

4.2.2 Analysis Of Training Cost

We can analyze one more example to check the sensitivity of our system with 10 machines and 15 parts. In this example Inter-material handling cost is same as first example, training cost is multiplied by 2.

The input datas for Table 4.16 are same as datas for Table 4.10. The only difference is;

W=2x[1000 600];

Table 4.16 Computational results of 10x15 part-machine matrix

Alternative matrix	***1***	***2***	***3***	***4***	***5***	***6***	***7***	***8***	***min***	***average***
Minimum Objective function value	40955	42675	40580	43290	42440	42585	42880	40840	40580	42030,6
Cost of inter-material handling	1365	725	675	1025	940	725	1065	665	675	898,125
Cost of manufacturing	39590	39950	39905	40265	39500	39860	39815	40175	39905	39882,5
Cost of training	0	2000	0	2000	2000	2000	2000	0	0	1250
Number of training	0	1	0	1	1	1	1	0	0	0,625
Number of cell	3	3	3	3	3	3	3	3	3	3
Number of inter-material handling	28	15	14	21	20	15	22	14	14	18,625

We can see in Table 4.16 that the cost combinations have no changes according to Table 4.10. Algorithm gives the same results as in Table 4.10. Also the minimum value came up with 3th route again, in this example. Because system already gives solution with no training in the first example.

Table 4.17 Computational results of 10x15 part-machine matrix

Alternative matrix	***1***	***2***	***3***	***4***	***5***	***6***	***7***	***8***	***min***	***Average***
Minimum Objective function value	40955	42850	40580	44290	43440	43585	43880	40840	40580	421552,5
Cost of inter-material handling	1365	2900	675	1025	940	725	1065	665	675	1170
Cost of manufacturing	39590	39950	39905	40265	39500	39860	39815	40175	39905	39882,5
Cost of training	0	0	0	3000	3000	3000	3000	0	0	1500
Number of training	0	0	0	1	1	1	1	0	0	0,5
Number of cell	3	3	3	3	3	3	3	3	3	3
Number of inter-material handling	28	37	14	21	20	15	22	14	14	21,375

The results for;

W=3x[1000 600] is shown in Table 4.17.

Algorithm gives same minimum objective function value (3th route) also for the example which training cost is multiplied by 3, which is shown in Table 4.17. Just for the 2nd route, the number of inter-material handlings are increased and the combination of system is changed. Instead of one time training with 3000 unit cost and 15 times inter-material handling, algorithm chose not to make training and to increase the inter-materail handling number to 37 with 2900 unit cost. The rest of routes have same combinations. We can say that the system is sensitive to cost changes of training but the 3th route is dominant and makes the system to act as nonsensitive. The same outcome is received by higher unit training costs. That means system gets nonsensitive for higher costs than 3xW.

The part-machine-cell matrix of minimum objective value in Table 4.16 and 4.17, the 3th route, is shown in table 4.18. Cell 1 has 2 machines, cell 2 has 5 machines and cell 3 has 3 machines. This configuration has 14 intercell movements between cell 1, cell 2 and cell 3 as shown in table 4.16 and 4.17.

Table 4.18 Part-machine-cell matrix

	cell 1		**cell 2**					**cell 3**		
machines	9	10	1	2	5	6	8	3	4	7

An example with higher capacity should be analyzed to see the effects. It is analyzed a problem with 18x30 sized part machine matrix in the next part.

4.3 Computational Results For 18x30 Sized Part-Machine Matrix

Now, we can analyze the table below, which shows results for 8 alternative part-machine matrix of our algorithm with 18 machines and 30 parts.

Numerical results are shown in Table 4.19. It can be seen that the optimal cell number for the inputs given, is 4. Algorithm gave 58 material handlings in the system at the end of 200 iterations which costed *4629* in the sixth route (the minimum value within all routes). The overall objective function cost of sixth route is *66609* unit cost.

The input datas for Table 4.19 are explained below.

The part-machine incidence matrix is (rows are machines);

X_p=

$$\begin{bmatrix}
0&1&1&0&1&0&1&1&0&1&1&1&1&0&1&1&1&0&0&0&0&1&0&1&1&0&1&1&1&0\\
1&0&1&1&1&1&0&0&0&1&1&0&0&0&1&1&0&1&1&1&1&0&1&1&1&0&0&0&1&1\\
1&1&1&0&0&0&0&1&0&1&1&0&1&1&0&0&0&0&1&0&1&0&1&1&0&1&0&1&0&1\\
0&0&1&1&0&0&0&1&1&0&0&0&0&1&1&1&1&0&0&0&0&1&0&1&1&0&1&1&1&0\\
0&0&0&0&1&0&1&0&1&1&0&1&0&1&0&0&0&1&1&0&0&0&0&1&1&1&1&0&0&0\\
1&0&1&1&1&1&1&0&1&1&0&1&1&0&0&0&1&1&0&0&0&0&1&1&1&0&0&0&1&1\\
1&0&1&1&0&1&1&0&0&0&0&1&0&1&1&0&0&1&0&0&0&0&0&0&0&1&0&1&1&0\\
1&0&1&1&1&1&0&0&1&0&1&1&0&1&0&1&0&0&0&0&1&1&0&0&0&0&1&0&1&1\\
0&0&1&1&0&1&1&1&0&0&1&0&1&1&0&1&1&0&0&0&0&0&1&1&0&0&0&1&0&1\\
0&0&0&1&0&0&0&1&1&0&1&1&1&1&1&0&1&0&1&1&1&1&0&1&1&0&1&1&0&0\\
0&1&1&0&1&1&0&0&0&1&0&1&1&1&1&0&0&1&0&1&1&0&0&0&1&1&0&1&0&0\\
1&1&0&0&1&1&0&0&0&1&1&0&1&1&0&0&0&1&1&0&0&0&1&0&1&0&1&1&0&1\\
1&1&0&1&1&0&1&1&0&0&0&0&1&1&1&1&0&0&0&0&1&0&1&1&0&0&0&0&1&1\\
0&1&0&1&0&1&1&0&1&0&1&0&0&1&1&0&1&0&1&0&0&0&0&1&0&1&1&1&1&0\\
0&0&0&1&0&1&0&1&1&0&1&0&1&1&1&0&1&0&1&1&1&1&0&0&1&0&1&1&0&0\\
0&0&1&1&1&1&1&0&0&1&1&1&0&1&0&1&0&0&1&1&1&0&1&1&0&1&0&0&0&1\\
1&0&1&1&0&0&0&0&1&0&1&1&1&1&1&0&1&0&1&0&0&1&0&1&0&0&0&1&1&0\\
0&0&1&0&1&1&0&1&1&1&1&1&0&1&0&1&1&0&1&0&1&0&1&1&0&0&1&0&0&0
\end{bmatrix}$$

Alternative machine matrix, which forms the alternative routes, is shown as X_a. Rows are parts and coloumns are machines.

$$X_a=\begin{bmatrix}
0 & 0\\
0 & 0\\
2 & 10\\
0 & 0\\
0 & 0\\
2 & 13\\
1 & 8
\end{bmatrix}$$

The demands of parts are shown as;

D_p=[10 15 5 20 10 15 30 20 10 5 25 15 15 35 25 20 30 10 15 10 5 40 30 40 25 25 5 15 35 15]

Production times of parts at each machine are shown in matrix Z. Rows are machines and coloumns are parts.

Z=

$$\begin{bmatrix}
1 & 3 & 2 & 3 & 1 & 4 & 3 & 7 & 2 & 2 & 4 & 1 & 1 & 3 & 3 & 6 & 3 & 4 & 1 & 4 & 3 & 2 & 3 & 5 & 5 & 1 & 2 & 3 & 6 & 3 \\
3 & 5 & 5 & 1 & 4 & 3 & 2 & 4 & 2 & 3 & 6 & 3 & 4 & 4 & 1 & 1 & 3 & 3 & 6 & 3 & 4 & 5 & 5 & 1 & 2 & 2 & 4 & 4 & 2 & 1 \\
7 & 2 & 2 & 4 & 1 & 1 & 3 & 5 & 1 & 1 & 4 & 2 & 3 & 5 & 5 & 1 & 4 & 3 & 2 & 4 & 2 & 3 & 2 & 4 & 1 & 1 & 3 & 3 & 4 & 4 \\
1 & 4 & 2 & 3 & 6 & 3 & 4 & 3 & 5 & 5 & 1 & 4 & 3 & 3 & 4 & 3 & 8 & 5 & 1 & 1 & 7 & 9 & 2 & 4 & 1 & 6 & 4 & 2 & 5 & 7 \\
8 & 5 & 1 & 1 & 4 & 2 & 3 & 4 & 2 & 7 & 5 & 1 & 1 & 8 & 3 & 6 & 1 & 1 & 5 & 6 & 6 & 4 & 6 & 8 & 2 & 6 & 4 & 3 & 8 & 9 \\
3 & 5 & 1 & 1 & 1 & 1 & 8 & 3 & 4 & 3 & 3 & 4 & 3 & 4 & 1 & 1 & 3 & 5 & 1 & 1 & 4 & 2 & 5 & 1 & 1 & 7 & 9 & 2 & 5 & 1 \\
3 & 5 & 1 & 1 & 3 & 5 & 5 & 1 & 4 & 3 & 2 & 4 & 2 & 3 & 2 & 4 & 1 & 6 & 3 & 4 & 3 & 5 & 5 & 1 & 4 & 5 & 6 & 6 & 4 & 6 \\
4 & 1 & 1 & 3 & 5 & 1 & 3 & 6 & 3 & 4 & 3 & 5 & 5 & 1 & 4 & 3 & 3 & 4 & 3 & 8 & 3 & 4 & 2 & 7 & 5 & 1 & 1 & 8 & 3 & 5 \\
3 & 2 & 4 & 2 & 1 & 1 & 4 & 2 & 3 & 5 & 5 & 1 & 4 & 1 & 1 & 5 & 6 & 6 & 4 & 6 & 5 & 5 & 1 & 4 & 5 & 6 & 6 & 3 & 4 & 3 \\
5 & 5 & 1 & 4 & 3 & 2 & 4 & 2 & 1 & 4 & 3 & 3 & 4 & 3 & 8 & 1 & 1 & 4 & 2 & 3 & 4 & 2 & 7 & 5 & 1 & 1 & 8 & 3 & 6 & 1 \\
7 & 2 & 2 & 4 & 1 & 1 & 3 & 2 & 7 & 5 & 1 & 1 & 8 & 3 & 6 & 1 & 5 & 1 & 4 & 3 & 2 & 4 & 2 & 3 & 2 & 4 & 1 & 1 & 4 & 2 \\
1 & 1 & 4 & 2 & 3 & 5 & 4 & 6 & 5 & 5 & 1 & 4 & 5 & 6 & 6 & 3 & 4 & 3 & 5 & 1 & 3 & 6 & 3 & 4 & 3 & 5 & 5 & 4 & 3 & 5 \\
7 & 2 & 2 & 4 & 1 & 1 & 3 & 4 & 3 & 4 & 1 & 1 & 3 & 5 & 4 & 2 & 7 & 5 & 1 & 2 & 1 & 4 & 3 & 3 & 1 & 1 & 5 & 6 & 6 & 4 \\
3 & 2 & 4 & 2 & 3 & 2 & 1 & 4 & 2 & 3 & 4 & 2 & 7 & 5 & 1 & 1 & 6 & 5 & 5 & 8 & 3 & 4 & 3 & 3 & 4 & 3 & 4 & 1 & 1 & 3 \\
9 & 2 & 4 & 4 & 2 & 3 & 4 & 2 & 7 & 5 & 5 & 5 & 1 & 4 & 5 & 6 & 6 & 4 & 6 & 6 & 3 & 4 & 3 & 5 & 5 & 1 & 4 & 3 & 5 & 1 \\
6 & 3 & 2 & 3 & 2 & 4 & 1 & 6 & 3 & 3 & 5 & 5 & 1 & 4 & 1 & 1 & 5 & 6 & 6 & 1 & 1 & 5 & 6 & 6 & 4 & 6 & 8 & 3 & 6 & 1 \\
6 & 3 & 4 & 1 & 4 & 3 & 2 & 5 & 1 & 4 & 1 & 4 & 5 & 6 & 6 & 3 & 4 & 3 & 6 & 6 & 3 & 4 & 3 & 5 & 1 & 3 & 6 & 1 & 4 & 1 \\
4 & 3 & 7 & 2 & 2 & 4 & 1 & 4 & 3 & 8 & 1 & 1 & 4 & 2 & 3 & 4 & 2 & 4 & 3 & 3 & 4 & 3 & 8 & 1 & 1 & 4 & 2 & 4 & 3 & 3
\end{bmatrix}$$

The temperature of Simulated Annaeling Algorithm probability equation is shown as;

Te=10^8

The operating costs of machines per time unit are;

B=[1; 2; 5; 5; 2; 4; 2; 5; 3; 4; 2; 3; 2; 5; 4; 2; 3; 2]

The batch size of a part is;

Bs=5

The inter-cell material handling costs for each part are shown as;

G=[10 15 30 20 30 15 25 15 30 15 14 23 16 15 17 29 22 12 23 14 18 23 25 30 10 35 27 32 13 24]

Costs of worker training skips between levels 1-2 and 3 are;

W=[1000 600]

Workers’ talent levels are;

T=[1 3 2 1 2 2 1 3 3]

Part-machine-cell matrix as an initial solution in the problem;

PMC=

$$\begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 20 & 21 & 22 & 23 & 24 & 25 & 26 & 27 & 28 & 29 & 30 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Machines' complexity levels are;

M=[2 3 1 3 2 3 2 2 3 3 2 3 1 3 2 3 3 2]

Table 4.19 Computational results of 18x30 part-machine matrix

Alternative matrix	***1***	***2***	***3***	***4***	***5***	***6***	***7***	***8***	***min***	***average***
Minimum Objective function value	67311	68638	67575	66978	68248	66609	67358	67990	66609	67588,4
Cost of inter-material handling	5661	4828	5985	4828	4828	4629	5798	6070	4629	5328,38
Cost of manufacturing	59650	60010	59590	59950	59620	59980	59560	59920	59980	59785
Cost of training	2000	3800	2000	2200	3800	2000	2000	2000	2000	2475
Number of training	2	5	2	3	5	2	2	2	2	2,875
Number of cell	4	4	4	4	4	4	4	4	4	4
Number of inter-material handling	74	60	76	72	60	58	72	75	58	68,375

We can analyze some of runs to be able to analyze the whole system.

The details of sixth route are shown in table 4.20. This is the minimum costed route. We can see that algorithm divided the system to 4 cells and complexity levels are 1-2-2-3.

Table 4.20 Components of the 6th route

minimum objective function value	complexity level of cell 1	complexity level of cell 2	complexity level of cell 3	complexity level of cell 4
66609	1	2	2	3

59980	4629	2000
manufacturing cost	material handling cost	training cost

The part-machine-cell matrix of the result that is shown in Table 4.20 is shown below;

A=

$$\left[\begin{matrix} 4 & 3 & 2 & 1 & 4 & 3 & 2 & 1 & 4 & 3 & 2 & 1 & 4 & 3 & 2 & 1 & 4 & 3 & 2 & 1 \\ 3 & 2 & 1 & 4 & 4 & 2 & 2 & 4 & 4 & 3 & 4 & 3 & 4 & 2 & 3 & 4 & 4 & 3 & 1 & 4 \end{matrix}\right.$$

$$\left.\begin{matrix} 4 & 3 & 2 & 1 & 4 & 3 & 2 & 1 & 4 & 3 \\ 2 & 3 & 1 & 3 & 3 & 4 & 3 & 3 & 3 & 2 \end{matrix}\right]$$

PMC_i=

$$\left[\begin{matrix} 4 & 8 & 12 & 16 & 20 & 24 & 28 & 3 & 7 & 11 & 15 & 19 & 23 & 27 & 2 & 6 & 10 & 14 & 18 & 22 \\ 0 & 0 & 2 & 6 & 7 & 14 & 0 & 0 & 1 & 10 & 12 & 15 & 18 & 0 & 0 & 0 & 0 & 0 & 0 & 3 \end{matrix}\right.$$

$$\left.\begin{matrix} 26 & 30 & 1 & 5 & 9 & 13 & 17 & 21 & 25 & 29 \\ 4 & 5 & 8 & 9 & 11 & 13 & 16 & 17 & 0 & 0 \end{matrix}\right]$$

When we ran our algorithm without Kaiser's Rule and specified the cell number manual as 3, the minimum objective value that our algorithm found is *70214*. That is worse than the result shown in Table 4.19. When the cell number is set to 5, the minimum objective value is *68119*. That is also worse than the result shown in Table 4.19. It was seen that the inter-material handling cost has decreased but the training cost has increased with 3 cells. Lower cell number causes less inter-material handling but more training to have workers which can operate more machines. The difference shows the Kaiser's Rule's improvement effect on our problem.

4.3.1 Analysis Of Inter-Material Handling Cost

We can analyze one more example to check the sensitivity of our system with 18 machines and 30 parts. We change some of inputs to see how our system reacts. Inter-material handling cost is multiplied by 4 in this example.

The input datas for Table 4.21 are same as datas for Table 4.19. The only difference is;

G=4x[10 15 30 20 30 15 25 15 30 15 14 23 16 15 17 29 22 12 23 14 18 23 25 30 10 35 27 32 13 24];

It can be seen in Table 4.21 that, algorithm gave 58 material handlings in the system at the end of 200 iterations which costed *18516* for the 6th route, which gave the minimum objective function value. When we multiply the cost of inter-material handling by four, algorithm enforces the system to make less inter-material handling. For example in the 1st route, algorithm chose to make 3 times more training and 56 times inter-material handling instead of 74 times inter-material handling. Because the second desicion costed lower. So the algorithm changed combinations of 1st, 3rd and 8th routes according to cost comparisons. However the route is not changed. The minimum value came up with 6th route again, in this example. The 6th route has *58* times carriage in Table 4.21 as in Table 4.19.

Table 4.21 Computational results of 18x30 part-machine matrix

Alternative matrix	***1***	***2***	***3***	***4***	***5***	***6***	***7***	***8***	***min***	***average***
Minimum Objective function value	82402	83122	82102	81462	82732	80496	84752	81432	80496	82366,5
Cost of inter-material handling	18952	19312	19312	19312	19312	18516	23192	19312	18516	19481,5
Cost of manufacturing	59650	60010	59590	59950	59620	59980	59560	59920	59980	59785
Cost of training	3800	3800	3200	2200	3800	2000	2000	2200	2000	2875
Number of training	5	5	4	3	5	2	2	3	2	3,625
Number of cell	4	4	4	4	4	4	4	4	4	4
Number of inter-material handling	56	60	60	72	60	58	72	60	58	62,25

We can analyze one more example to check the sensitivity of our system with 18 machines and 30 parts. Inter-material handling cost is multiplied by 8 in this example.

The input datas for Table 4.22 are same as datas for Table 4.19. The only difference is;

G=8x[10 15 30 20 30 15 25 15 30 15 14 23 16 15 17 29 22 12 23 14 18 23 25 30 10 35 27 32 13 24];

Table 4.22 Computational results of 18x30 part-machine matrix

Alternative matrix	***1***	***2***	***3***	***4***	***5***	***6***	***7***	***8***	***min***	***average***
Minimum Objective function value	101354	98418	101414	100374	98196	98892	96608	101152	96608	99551
Cost of inter-material handling	37904	34008	38624	37224	34176	35112	33848	38432	12693	36166
Cost of manufacturing	59650	60010	59590	59950	59620	59980	59560	59920	59560	59785
Cost of training	3800	4400	3200	3200	4400	3800	3200	2800	3200	3600
Number of training	5	6	4	4	6	5	4	4	4	4,75
Number of cell	4	4	4	4	4	4	4	4	4	4
Number of inter-material handling	56	53	60	59	53	53	54	59	54	55,875

In table 4.22, it can be seen that, the optimal cell number for the inputs given above, is not changed. Because the part-machine incidence matrix is same. We should check the other datas. We can see that the algorithm changed the combinations of 2nd, 4th, 5th,7th and 8th routes. And the minimum value in overall routes and iterations come up with 7th route with *96608* unit cost. Algorithm decided to increase training number according to the unit cost increasement in inter-material handling. For example, algorithm gave *54* material handlings in the system at the end of 200 iterations which costed *33848* for the minimum costed route (7th route). But it was *72* material handlings in the system at the end of 200 iterations in Table 4.21. The total cost increased by the increasement of unit cost for inter-material handling. But it can be seen that the number of inter-material handling is decreased. This is the main point that we want. Algorithm enforces the system to make less inter-material handling. So, the problem is sensitive to inter-material handling cost.

It is analyzed if the inter-material handling cost is multiplied by 12 in the next example.

The input datas for Table 4.23 are same as datas for Table 4.19. The only difference is;

G=12x[10 15 30 20 30 15 25 15 30 15 14 23 16 15 17 29 22 12 23 14 18 23 25 30 10 35 27 32 13 24];

We can see in Table 4.23, that the cost combination of 3rd route is changed when all values are compared with values in Table 4.22. When we redound the cost of inter-material handling more, algorithm enforces the system to make less inter-material handling once again. Because system should use either inter-material handlings or worker trainings to compose cells. For example in 3rd route, algorithm chose the 600 unit cost of one more time training and 54 times carriage instead of *57936* unit cost of 60 times carriage and 4 times training. If inter-material handlings are decreased, then training has more importance and responsibility. However the minimum value came up with 2nd route, in this example. It is seen that the differences got smaller when we multiply inter-material handling cost by 12. Different unit costs cause different route choises and changes the numbers of training and inter-material handling traffics. But we can say that the system is nonsensitive from now on, because the algorithm gives same results and makes same choise with higher values than 12xG.

Table 4.23 Computational results of 18x30 part-machine matrix

Alternative matrix	Minimum Objective function value	Cost of inter-material handling	Cost of manufacturing	Cost of training	Number of training	Number of cell	Number of inter-material handling
1	120306	56856	59650	3800	5	4	56
2	113334	48924	60010	4400	6	4	53
3	114618	51228	59590	3800	5	4	54
4	120546	57396	59950	3200	4	4	59
5	115284	51264	59620	4400	6	4	53
6	116448	52668	59980	3800	5	4	53
7	113532	50772	59560	3200	4	4	54
8	120368	57648	59920	2800	4	4	59
min	113334	48924	60010	4400	6	4	50
average	116805	53344,5	59785	3675	4,875	4	55,125

4.3.2 Analysis Of Training Cost

We can analyze one more example to check the sensitivity of our system for training cost with 18 machines and 30 parts. It is said that we changed some of inputs to see how our system reacts. In this example Inter-material handling cost is same as first example (Table 4.19), training cost is multiplied by 2.

The input datas for Table 4.24 are same as datas for Table 4.19. The only difference is;

W=2x[1000 600];

Table 4.24 Computational results of 18x30 part-machine matrix

Alternative matrix	***1***	***2***	***3***	***4***	***5***	***6***	***7***	***8***	***min***	***average***
Minimum Objective function value	69311	71411	68688	69178	70573	68517	68274	69990	68274	69492,8
Cost of inter-material handling	5661	7401	7898	4828	6953	6537	7514	6070	7514	6607,75
Cost of manufacturing	59650	60010	59590	59950	59620	59980	59560	59920	59560	59785
Cost of training	4000	4000	1200	4400	4000	2000	1200	4000	1200	3100
Number of training	2	2	1	3	2	1	1	2	1	1,75
Number of cell	4	4	4	4	4	4	4	4	4	4
Number of inter-material handling	74	81	87	72	73	86	85	75	85	79,125

We can see in Table 4.24 that the cost combination of 2nd, 3th, 5th, 6th and 7th routes are changed when it is compared with values in Table 4.19. Algorithm chose the 7th route as minimum value with *7514* unit cost of 85 times carriage and 1 time training instead of 72 times carriage and 2 times of training for this route. This is the minimum objective function value in overall routes and iterations for this example (it was the 6th route in the first example). When we redound the cost of training, algorithm enforces the routes of the system to make less training. Because of that algorithm makes the number of inter-material handling higher in some alternative routes. But some routes (1st, 4th and 8th) accept to endure the training cost instead of having carriage. Training cost and inter-material handling cost are inversely proportional cost components.

We can analyze one more example to check the sensitivity of our system with 18 machines and 30 parts. Training cost is multiplied by 3 in this example.

The input datas for Table 4.25 are same as datas for Table 4.19. The only difference is;

W=3x[1000 600];

In this example, algorithm changed the combination of the 3th, 4th and 8th routes. It is seen that the algorithm choose third route as minimum objective function solution. It was the seventh route in Table 4.24. In the 3th route, algorithm decided to make 95 times inter-material handling and not to make any training instead of 87 times inter-material handling and 1 time training.

According to these explanations, it is seen that our algorithm with 18x30 sized matrix is sensitive to training unit cost changes. But for further trials, the algorithm has same solutions and same choise. This shows that the algorithm is nonsensitive on increasements, which are more than 3 times, for training.

Table 4.25 Computational results of 18x30 part-machine matrix

Alternative matrix	Minimum Objective function value	Cost of inter-material handling	Cost of manufacturing	Cost of training	Number of training	Number of cell	Number of inter-material handling
1	71311	5661	59650	6000	2	4	74
2	73411	7401	60010	6000	2	4	81
3	68695	9105	59590	0	0	4	95
4	72835	6885	59950	6000	2	4	87
5	72573	6953	59620	6000	2	4	73
6	69517	6537	59980	3000	1	4	86
7	68874	7514	59560	1800	1	4	85
8	71205	8285	59920	3000	1	4	86
min	68695	9105	59590	0	0	4	95
Average	71052,6	7292,63	59785	3975	1,375	4	83,375

The part-machine-cell matrix of Table 4.25, 3rd route is shown in table 4.26. Cell 1 has 2 machines, cell 2 has 3 machines, cell 3 has 4 machines, cell 4 has 9 machines. This configuration has *95* intercell movements between these four cells as shown in table 4.25.

Table 4.26 Part-machine-cell matrix

	cell 1		**cell2**			**cell3**				**cell4**								
machines	12	18	3	9	10	6	14	16	17	1	2	4	5	7	8	11	13	15

17 numerical examples are processed by the proposed SA algorithm. 5 of them are constructed in small scaled size, 6 of them are constructed in medium scaled size, 6 of them are constructed in large scaled size. When values are getting higher, the difference will be much more higher. The running time by the proposed simulated annealing algorithm is longer for the large scaled sizes.

CHAPTER FIVE
CONCLUSION

In the literature review, it was not seen that much studies about cell formation with human issue. In real life manufacturing systems, it can not be thought the system without workers and effects of them. A study was done in this paper about that topic.

In this study, a cell formation was done for single period and with alternative routes of parts. It was tried to determine cells by the cost calculation of demand for parts, training for workers and inter-cellular material handling for parts again. Also the basis is similarty coefficients. Jaccard's similarity coefficient formulation was used.

The objective function was about minimizing the cost of the system overall, using Matlab R2008a program. Kaiser's Rule was used to find the optimal cell number and to form cells according to that number. To check the efficiency of Kaiser's Rule in this problem, a pre-determined numbers as cell numbers were given to the system. It was seen that we received better results (lower cost values) when we used The Kaiser's Rule for giving the cell number as initial number. Numerical examples showed that Kaiser's Rule made the algorithm faster. Because the cell number that Kaiser's Rule gave, gave a lower cost value at the beginning. This means, Kaiser's Rule provided a better assignment of machines and parts.

Simulated Annealing Algorithm (SAA) was used to construct the system. To determine cells and their part families, SAA minimized the sum of system cost for the all parts. SAA was chosen, because it was seen that this heuristic could be simply performed. Also SA algorithm gives good results among other heuristics in general.

Cell formation was done with talented workers. We could train them if needed. Also system let to choose different routes for manufacturing. These routes were predetermined for the problem, like demand, manufacturing times, manufacturing costs, trainig costs, material handling costs.

The neighbourhood solution was found by mutation. This also allowed to the system, to change the number of machines in each cell and see the different combinations for number of cell members.

Different combinations were used for cost components in numerical examples to see the effects on results. The training cost and inter-material handling cost are used to see the effects. Because these two components affects the decision variables in the problem. It can be seen that the increased cost forced the system to decrease action scores of that cost. But until a cost level. This level is different for each problem. When the cost combinations are higher than that level, increasement does not affect the result mainly by scores, but increases the total cost.

Multi-period cell formation will be studied for future researches. Multi-period is very important for the real world systems. A new cell formation is not efficient for a plant for each demand period. So a cell formation that considers the whole demand periods should be planned.

Also the layout planning and change of place costs for machines should be added for future studies. Because the layout is the part of cell formation problem and have a big importance on real life systems. Changing the place of a machine means extra-cost for the manufacturing system. Configuration should be done according to that issue.

REFERENCES

Aryanezhad, M. B., Deljoo, V., Mirzapour Al-e-hashem, & S. M. J., (2009). Dynamic cell formation and the worker assignment problem: a new model. *The International Journal of Advanced Manufacturing Technology, 41*, 329–342.

Askin, R. G. & Huang, Y., (2001). Forming effective worker teams for cellular manufacturing. *International Journal of Production Research, 39(11),* 2431–2451.

Azizi, N., Zolfaghari, S., & Liang, M., (2010). Modeling job rotation in manufacturing systems: The study of employee’s boredom and skill variations. *Int. J. Production Economics, 123*, 69–85.

Balakrishnan, J. & Cheng, C. H., (2007). Multi-period planning and uncertainty issues in cellular manufacturing: A review and future directions. *European Journal of Operational Research, 177*, 281-309.

Balasubramanian, K. N. & Panneerselvam, R., (1993). Covering technique-based algorithm for machine grouping to form manufacturing cells. *International Journal of Production Research, 31,* 1479–1504.

Bashir, H. A. & Karaa, S., (2008). Assessment of clustering tendency for the design of cellular manufacturing systems. *Journal of Manufacturing Technology Management, 19 (8),* 1004 – 1014.

Bhadury, J. & Radovilsky, Z., (2006). Job rotation using the multi-period assignment model. *International Journal of Production Research, 44*, 4431–4444.

Bidanda, B., Ariyawongrat, P., LaScola Needy, K., Norman, B. A., & Tharmmaphornphilas, W., (2005). Human related issues in manufacturing cell design, implementation, and operation: a review and survey. *Computers and Industrial Engineering, 48,* 507–523.

Black, J. T.,(1991). *The design of the factory with a future*. New York: McGraw-Hill.

Cesani, V.I. & Steudel, H.J., (2005). A study of labor assignment flexibility in cellular manufacturing systems. *Computers and Industrial Engineering, 48,* 571–591.

Cheng, C. H., Gupta, Y.P., Lee, W.H., & Wong, K.F., (1998). A TSP-based heuristic for forming machine groups and part families. *International Journal of Production Research, 36(5),* 1325–37.

Choobineh, F., (1988). A framework for the design of cellular manufacturing systems. *International Journal of Production Research, 26,* 1161- 1172.

Corominas, A., Pastor, R., & Rodriguez, E., (2006). Rotational allocation of tasks to multifunctional workers in a service industry. *Int. J. Production Economics, 103,* 3–9.

Dawis, D.J. & Mabert, V.A., (2000). Order dispatching and labor assignment in cellular manufacturing systems. *Decision Sciences, 31(4; ABI/INFORM),* 745

Fitzpatrick, E. L. & Askin, R. G., (2005). Forming effective worker teams with multi-functional skill requirements. *Computers and Industrial Engineering, 48,* 593–608.

Defersha, F. M. & Chen, M., (2006). A comprehensive mathematical model for the design of cellular manufacturing systems. *Int. J. Production Economics, 103,* 767–783.

Guerrero, F., Lozano, S., Smith, K. A., Canca, D., & Kwok, T., (2002). Manufacturing cell formation using a new self-organizing neural network. *Computers and Industrial Engineering, 42,* 377–382.

Gupta, T., (1993). Design of manufacturing cells for flexible environment considering alternative routing. *International Journal of Production Research, 31,* 1259–1273.

Ham, I., Hitomi, K., & Yoshida, T., (1985). *Group technology application to production management.* Nijhoff, EN: Kluwer.

Hwang, H. & Ree, P., (1996). Routes selection for the cell formation problem with alternative part process plans. *Computers and Industrial Engineering, 30,* 423–431.

Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P., (1983). Optimization by simulated annealing. *Science, 220,* 671–680.

Mahdavi, I., Aalaei, A., Paydar, M. M., & Solimanpur, M., (2010). Designing a mathematical model for dynamic cellular manufacturing systems considering production planning and worker assignment. *Computers and Mathematics with Applications, 60,* 1014-1025.

Mosier, C. T. & Taube, L., (1985). The facets of group technology and their impact on implementation. *OMEGA, 13,* 381-391.

Norman, B. et al, (2002). Worker assignment in cellular manufacturing considering technical and human skills. *International Journal of Production Research, 40 (6),*1479–1492.

Pailla, A., Trindade, A. R., Parada, V., & Ochi, L. S., (2010). A numerical comparison between simulated annealing and evolutionary approaches to the cell formation problem. *Expert Systems with Applications, 37,* 5476–5483.

Pinedo, M., (2004). *Planning and scheduling in manufacturing and services.* Springer, New York, 424-431.

Plaquin, M. & Pierreval, H., (2000). Cell formation using evolutionary algorithms with certain constraints. *Int. J. Production Economics, 64(1-3),* 267–278.

Saeedi, S., Solimanpur, M., Mahdavi, I., & Javadian, N., (2010). Heuristic approaches for cell formation in cellular manufacturing. *J. Software Engineering and Applications, 3,* 674-682.

Safaei, N., Saidi-Mehrabad, M., & Jabal-Ameli, M. S., (2008). A hybrid simulated annealing for solving an extended model of dynamic cellular manufacturing system. *European Journal of Operational Research, 185,* 563–592.

Seifoddini, H., (1988). *Recent developments in production research.* Elsevier Science Publishers B.V.. The Netherlands, 562–570.

Selvam, R. P. & Balasubramanian, K. N., (1985). Algorithmic grouping of operation sequences. *Engineering Cost and Production Economics, 9,* 125–134.

Shunk, D., (1985). Group technology provides organized approaches to realizing benefits of CIMS. *Industrial Engineering, 17,* 74-81.

Slomp, J., Bokhorst, J. A. C., & Molleman, E., (2005). Cross-training in a cellular manufacturing environment. *Computers and Industrial Engineering, 48,* 609–624.

Stevens, M. J. & Campion, M. A., (1994). The knowledge, skill, and ability requirements for teamwork: implications for human resource management. *Journal of Management, 20(2),* 503-530.

Suer, G. & Bera, I., (1998). Optimal operator assignment and cell loading when lot-splitting is allowed. *Com. and Ind. Eng., 35(3– 4),*431–434.

Sütçü, A., Tanrıtanır, E., Durmuşoğlu, B., & Koruca, H. I., (2011). An integrated methodology for layout design and work organisation in a furniture manufacturing plant. *South African Journal of Industrial Engineering, 22(1),* 183-197.

Tam, K. Y., (1990). An operation sequence based similarity coefficient for part families formations. *Journal of Manufacturing Systems, 9,* 55–68.

Tavakkoli-Moghaddam, R., Rahimi-Vahed, A. R., Ghodratnama, A., & Siadat, A., (2009). A simulated annealing method for solving a new mathematical model of a multi-criteria cell formation problem with capital constraints. *Advances in Engineering Software, 40,* 268–273.

Tavakkoli-Moghaddam, R., Javadian, N., Javadi, B., & Safaei, N., (2007). Design of a facility layout problem in cellular manufacturing systems with stochastic demands. *Applied Mathematics and Computation, 184,* 721–728.

Vakharia, A.J. & Wemmerlöv, U., (1990). Designing a cellular manufacturing system: A materials flow approach based on operation sequences. *IIE Transactions, 22,* 84–97.

Won, Y. K., (2000). New p-median approach to cell formation with alternative process plans. *International Journal of Production Research, 38,* 229–240.

Won, Y. K. & Kim, S. H., (1997). Multiple criteria clustering algorithm for solving the group technology problem with multiple process routings. *Computers and Industrial Engineering, 32*, 207–220.

Wu, X., Chu, C. H., Wang, Y., & Yan, W., (2007). A genetic algorithm for cellular manufacturing design and layout. *European Journal of Operational Research, 181*, 156-167.

Wu, T. H., Chang, C. C., & Chung, S. H., (2008). A simulated annealing algorithm for manufacturing cell formation problems. *Expert Syst. Appl., 34(3),* 1609–1617.

Yin, Y. & Yasuda, K., (2006). Similarity coefficient methods applied to the cell formation problem: a taxonomy and review. *Int. J. Production Economics, 101,* 329-352.

APPENDIX

A1.Cms Matlab Codes

```
clear;
nn=2;
np=3;
```

$$X_p=\begin{bmatrix} 0&1&1&0&0&1&1\\ 1&0&1&0&1&1&0\\ 1&1&0&1&0&1&0\\ 0&0&0&1&1&0&0\\ 1&0&1&0&1&0&1 \end{bmatrix};$$

```
nnn=0;
Xa=[0 0;0 0;2 4;0 0;0 0;2 5;1 3];
a=find(Xa(:,1)>0);
aa=length(a);

for x1=1:nn
    for x2=1:nn
        for x3=1:nn
            Xp(Xa(a(1),:),a(1))=0;
            Xp(Xa(a(2),:),a(2))=0;
            Xp(Xa(a(3),:),a(3))=0;
            Xp(Xa(a(1),x1),a(1))=1;                 %alternative machine matrix
            Xp(Xa(a(2),x2),a(2))=1;               % finds parts which have alternative routes
            Xp(Xa(a(3),x3),a(3))=1;


nnn=nnn+1;
BN=10000000000;                                   %a big number
ii=1;
jj=7;
kk=5;
LL=2;
iterationmax=200;
temperature=100000000;

obj=BN*(ones(1,iterationmax));
TT=zeros(1,iterationmax);
```

D_p=[10 15 5 20 35 15 30];   %demands of parts%

$$Z=\begin{bmatrix} 1&3&2&3&1&4&3\\ 3&5&5&1&4&3&2\\ 7&2&2&4&1&1&3\\ 1&4&2&3&6&3&4\\ 8&5&1&1&4&6&3 \end{bmatrix};$$

%production times of parts, rows are machines%

```
B=[ 8; 5; 9; 7; 6];                 %operating cost of machines per time
                                    unit coloumns are machines%
```

```
bs=5;                                         %batch size%

G=[10 15 20 10 15 15 10];

W=[1000 600];                                 %training costs by levels%

T=fix(1+(3*rand(1,18)));                      %talent levels of workers%

PMC=[1 2 3 4 5 6 7;1 2 3 4 5 0 0];            %Initial part-machine-cell matrix

M=[2 3 1 3 2 3 2 2 3 3 2 3 1 3 2 3 3 2];      %complexity level of machines

WC=[1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18];   %worker matrix%

obj=0;

Am=Xp;                                        %Kaiser's Rule optimal cell
                                              number
                                              calculations%

[u,v]=size(Am);
Xm=zeros (u);
Ym=zeros (u);
Zm=zeros (u);
  for i=1:u
    for j=1:u
      for am=1:v
        if Am(i,am)==Am(j,am) && Am(i,am)==1;
          Xm(i,j)=Xm(i,j)+1;
          Ym(i,j)=Ym(i,j)+Am(i,am);
          Zm(i,j)=Zm(i,j)+Am(j,am);
        else
          Ym(i,j)=Ym(i,j)+Am(i,am);
          Zm(i,j)=Zm(i,j)+Am(j,am);
        end
      S(i,j)=Xm(i,j)/(Ym(i,j)+Zm(i,j)-Xm(i,j));
      end
    end
  end
C=eig(S);
th(nnn)=hucresayisi;                          %optimal cell number%
hucresayisi=0;



  if jj>kk;
    PMCi=zeros(2,jj);
    A=zeros(2,jj);
    hhh=jj;
  else
    PMCi=zeros(2,kk);
    A=zeros(2,kk);
    jj=kk;
    hhh=kk;
  end

  for j=1:2;
    huc=(th(nnn));
    for i=1:jj;
```

```
            if huc>=1;
            A(j,i)=huc;
            huc=huc-1;
            else
            huc=(th(nnn));
            A(j,i)=huc;
            huc=huc-1;
            end
        end
    end

Ayedek=A;

for iteration=1:iterationmax
    egitimadett=0;
    a1=1;
    b1=1;
    d=0;
    a0=zeros(1,max(A));
    b0=zeros(1,max(A));
    c0=zeros(1,max(A));
    alt=Xp;
    for p=1:(th(nnn));
        A=A';
        A=reshape(A,1,hhh+hhh);
        h=find(A==p);
        a0(p)=length(find(h<(hhh+1)));
        b0(p)=length(find(h>hhh));
        c0(p)=length(h);
        i1=1;
        for i=a1:(a0(p)+a1-1)
            for k=i1:i1
                if h(k)<=hhh
                PMCi(1,i)=PMC(1,h(k));
                end
            end
            i1=i1+1;
        end
        a1=a1+a0(p);
        b1=b1+b0(p);
        for j=(a0(p)+1):c0(p)
            d=d+1;
            if d<=hhh;
            PMCi(2,d)=PMC(2,(h(j)-hhh));
            end
        end
    end


ZX=(Z.*Xp);

DZX=Dp*ZX';                                             %Total time on machines in the period%

DZXB=DZX*B;

%Inter-Cellular Mat. Handling calculations%
```

```
l=zeros(1,jj);
li=zeros(1,jj);
for j=1:jj;
   U=find(Xp(:,j)==1);
   U=U';
   m=length(U);
   for p=1:(th(nnn));

         tas=zeros(1,p);
         h=find(A==p);
         Xpt=PMC(2,((h(find(h>hhh)))-hhh));
      for i=1:m;
         if length(find(U(i)==Xpt))>0;
            l(j)=1+l(j);
         else
            l(j)=0+l(j);
         end
         tas(p)=tas(p)+l(j);
      end
      if l(j)>0;
        li(j)=li(j)+1;
      end
   end
   tasima(j)=li(j)-1;
end


%Training calculations%
Wtotal=0;
b=1;
egitimadett=0;

   for p=1:(th(nnn));
      h=find(A==p);
      a0=zeros(1,(max(A)+1));
      a0(p+1)=length(find(h<(hhh+1)));
      x=0;
      b=a0(p)+b;

      for i=b:(a0(p+1)+b-1);
         if PMCi(2,i)~=0;
            if M(PMCi(2,i))>x;
                  x=M(PMCi(2,i));
            end
         end
      end

      if (T(WC(p)))<x;                          %if a machine has higher complexity than
                                                %worker's talent level, then training is done%
         for i=(T(WC(p))):(x-1);

            Wtotal=Wtotal+W(i);
            egitimadett=egitimadett+1;
         end
      end
   end

%Objective function%
```

```
obj(iteration)=(DZXB)+(sum(tasima.*G.*(Dp/bs)))+(Wtotal);
A=reshape(A,hhh,2);
A=A';

    if obj(iteration)<BN;
       Ayedek=A;
       BN=obj(iteration);
       if iteration>1;
         TT(iteration)=TT(iteration-1);
       end
       if obj(iteration)<=min(obj);
          egitimadet(nnn)=egitimadett;
          alt=Xp;
          Wmin=Wtotal;
          Tasmin(nnn)=sum(tasima);
          Ctasima=(sum(tasima.*G.*(Dp/bs)));
          Curetim=DZXB;
          Cegitim=Wtotal;
          Egitimsayisi=sum(egitimadet);

       end

       for j=1:hhh;
          A(2,j)=fix((th(nnn)*rand)+1);
       end
       sonuc=BN;

     else
           temperature=temperature*(0.95);
           TT(iteration)=temperature;
           if exp((BN-obj(iteration))/temperature)>(2*rand(1));
              BN=obj(iteration);
              Ayedek=A;
              for j=1:hhh;
                 A(2,j)=fix((th(nnn)*rand)+1);
              end

           else
              for j=1:hhh;
                 A=Ayedek;
                 A(2,j)=fix((th(nnn)*rand)+1);

              end
           end
    end

end
objmin(nnn)=min(obj);
Ctasimamin(nnn)=Ctasima;
Curetimmin(nnn)=Curetim;
Cegitimmin(nnn)=Cegitim;
      end
   end
end
```