DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

DESIGN OF MANUFACTURING CELLS FOR UNCERTAIN PRODUCTION REQUIREMENTS

by
Özgür ESKİ

December, 2007
İZMİR

DESIGN OF MANUFACTURING CELLS FOR UNCERTAIN PRODUCTION REQUIREMENTS

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 Doctor of Philosophy in Industrial Engineering, Industrial Engineering Program

by
Özgür ESKİ

December, 2007
İZMİR

Ph.D. THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"DESIGN OF MANUFACTURING CELLS FOR UNCERTAIN PRODUCTION REQUIREMENTS"** completed by **ÖZGÜR ESKİ** under supervision of **Asst. Prof. Dr. ŞEYDA TOPALOĞLU** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Doctor of Philosophy.

Asst. Prof. Dr. Seyda TOPALOĞLU

Supervisor

Prof. Dr. Semra TUNALI

Thesis Committee Member

Prof. Dr. Tatyana YAKHNO

Thesis Committee Member

Prof. Dr. İrem ÖZKARAHAN

Second Supervisor

Examining Committee Member

Examining Committee Member

Examining Committee Member

Prof.Dr. Cahit HELVACI
Director
Graduate School of Natural and Applied Sciences

ACKNOWLEDGEMENTS

I would like to express my thanks and appreciation to my advisors Dr. Irem Ozkarahan and Dr. Seyda Topaloglu for their invaluable support and advice that helped me to complete this dissertation. I would like to acknowledge other committee members Dr. Semra Tunalı and Dr. Tatyana Yakhno for their helpful suggestions. I also would like to thank Dr. Gökalp Yıldız and Dr Ceyhun Araz for their encouragement and support.

Last, I would like to thank my family and fiancé for their love and support during my graduate study.

Özgür ESKİ

DESIGN OF MANUFACTURING CELLS FOR UNCERTAIN PRODUCTION REQUIREMENTS

ABSTRACT

Cellular manufacturing has been seen as an effective strategy to the changing worldwide competition. Most of the existing cell design methods ignore the existence of stochastic production requirements and routing flexibility. In this study, a simulation based Fuzzy Goal Programming model is proposed for solving cell formation problems considering stochastic production requirements and routing flexibility. The model covers the objectives of minimizing the number of exceptional elements, maximizing system utilization, minimizing mean tardiness and minimizing the percentage of tardy jobs. The simple additive method and max-min method are used to handle fuzzy goals. A tabu search based solution methodology is used for solution of the proposed models and the results are presented.

Keywords: Cellular manufacturing, fuzzy goal programming, tabu search, simulation

ÜRETİM İHTİYAÇLARININ BELİRSİZ OLDUĞU DURUMDA İMALAT HÜCRELERİNİN TASARIMI

ÖZ

Hücresel imalat firmaların rekabet gücünü artıracak etkin bir strateji olarak değerlendirilmektedir. Hücresel imalat çalışmalarında genellikle göz ardı edilen ancak tasarım kararları üzerinde etkisi bulunan iki önemli etken üretim gereksinimlerindeki belirsizlik ve alternatif süreç planlarının varlığıdır. Bu çalışmada, belirsizlik ve alternatif proses planlarının varlığını dikkate alarak hücre tasarımını gerçekleştiren, benzetim modelleri ile entegre edilmiş hibrid bir bulanık hedef programlama modeli geliştirilmiştir. Modelin çözümü, tabu arama algoritması kullanılarak gerçekleştirilmiş, elde edilen sonuçlar çalışma kapsamında sunulmuştur.

Anahtar sözcükler: Hücresel imalat, bulanık hedef programlama**,** tabu arama algoritması**,** benzetim

CONTENTS

CHAPTER ONE

INTRODUCTION

1.1 Cellular Manufacturing

Shorter life-cycles, unpredictable demand and customized products have forced manufacturers to improve the efficiency and productivity of their production activities. Manufacturing systems must be able to produce items with low production costs and high quality as possible in order to meet the customers' demand on time. Moreover manufacturing systems have gone through major changes during recent years mainly due to advances in technology and new strategies to deal with the technology. Informational vagueness in parameter estimates is being recognized as a reality in most of the problems in manufacturing system design. Manufacturing systems, today, should be able to respond quickly to changes in product design, product demand, technology etc. Traditional manufacturing systems such as job shops and flow lines are not capable of satisfying such requirements. The concept of cellular manufacturing (CM) is one of the most effective strategies to the changing worldwide competitive environment.

Job shops are the most common manufacturing systems in the world. Job shops are designed to achieve maximum flexibility in order to produce a wide variety of products with small lot sizes. In this type of production system, parts require different processing operations and sequences. Parts are released to the job as batches and general purposed machines are utilized. In general, machines are grouped according to their functions. Figure 1.1 illustrates a typical job shop layout. This type of machine layout is also known as functional layout.

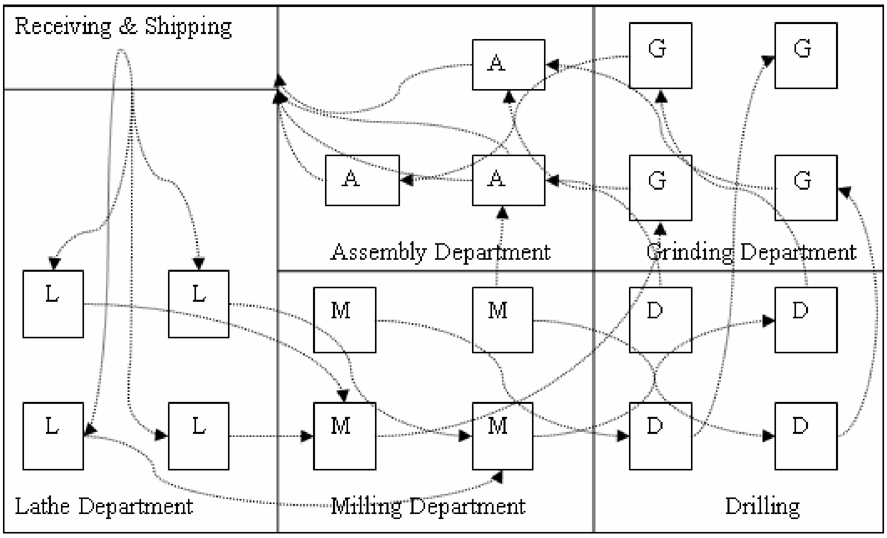


Figure 1.1 Job shop manufacturing

In job shop layout, products must flow from one department to another through its various processing steps. This results in long waiting times and difficulties in production scheduling and control. Therefore parts are moved in batches to make processing more economical. Each part in a batch has to wait for remaining parts of its batch before it is moved to next processing step. The end result is higher inventory costs, larger scraps and less customer satisfaction due to long delivery times. In job shops, jobs spend 95% of their time in non-productive activities such as waiting in queue, waiting for machine setups etc.(Aksin & Standridge, 1993)

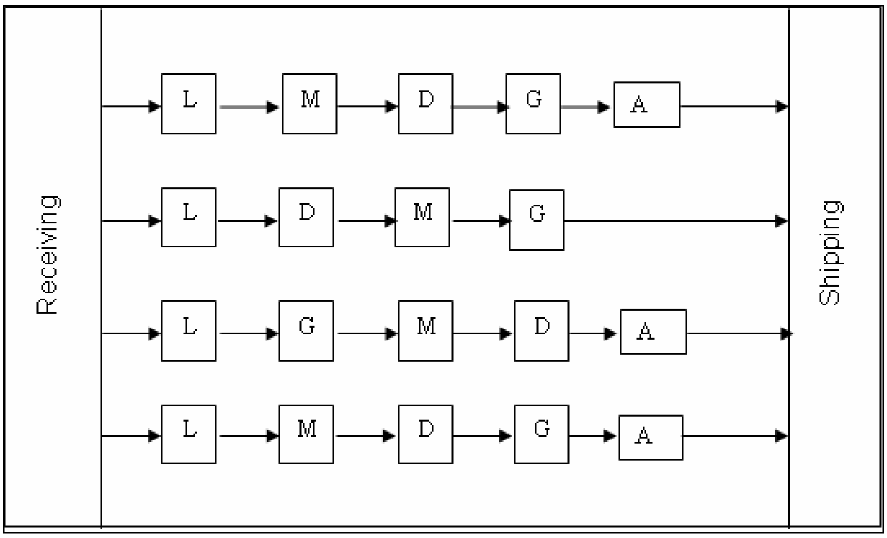


Figure 1.2. Flow line Manufacturing

In contrast to job shops, flow lines are designed for high volume industries and require high capital commitments while retaining little production flexibility. A flow line is organized according to the processing sequence of a product. Specialized machines dedicated to the manufacture of utilized to achieve high production rates. Figure 1.2 shows an example of a flow line.

Job shops and flow lines are not able to meet today's production requirements where manufacturing systems are often required to reconfigured to respond to changes in product design and demand. Cellular manufacturing (CM), an application of group technology (GT) offers a middle-ground alternative to the traditional job shops or flow lines. GT is defined as a manufacturing philosophy identifying and grouping similar parts in part families in order to take advantage of similarities in both design and manufacturing (Selim, Askin and Vakharia, 1998). The driving force behind CM is the need in a wide variety of industries to simplify production requirements while still ensuring production flexibility. The job shop in Figure 1.1 can be converted into Cellular manufacturing system (CMS) as shown in Figure 1.3. The benefits gained by such a conversion are improved competitiveness by reduction of lead times (primarily reduction of times associated with movement of parts), less work in process (WIP), less transport distance for materials, improved plant capacity and flexibility by reducing the setup time.

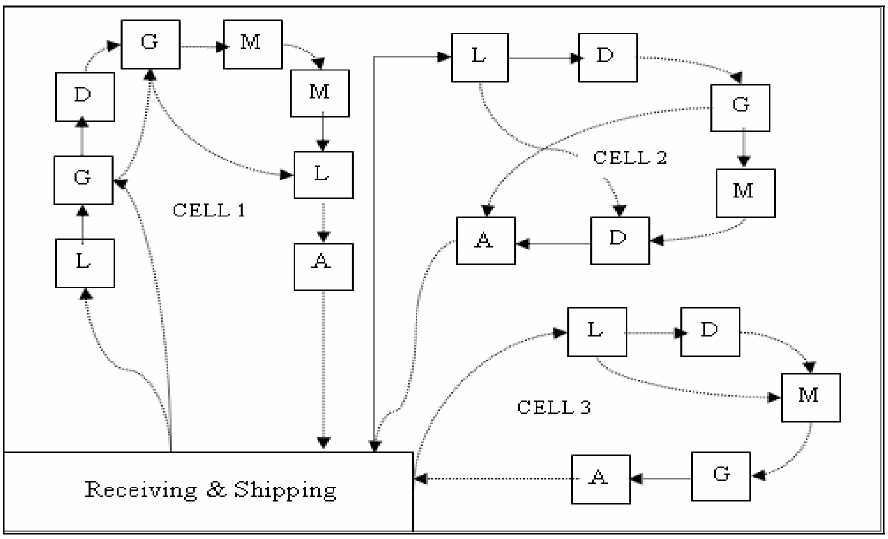


Figure 1.3. Cellular manufacturing system

CM is a hybrid system linking the advantages of job shops and flow lines. As seen from Figure 1.3, in CM, machines are dedicated to a part family and located into close proximity. This provides the efficient flow and high production rate similar to flow lines. On the other hand, similar to job shops, the use of general purpose machines and equipments provide flexibility in producing a variety of products. Generally, CM production environments are less complex to manage than job shops, but usually less flexible than job shops. Conversely, cell based production is more flexible than flow lines, but requires additional organization and management compared with dedicated transfer lines that manufacture single product types.

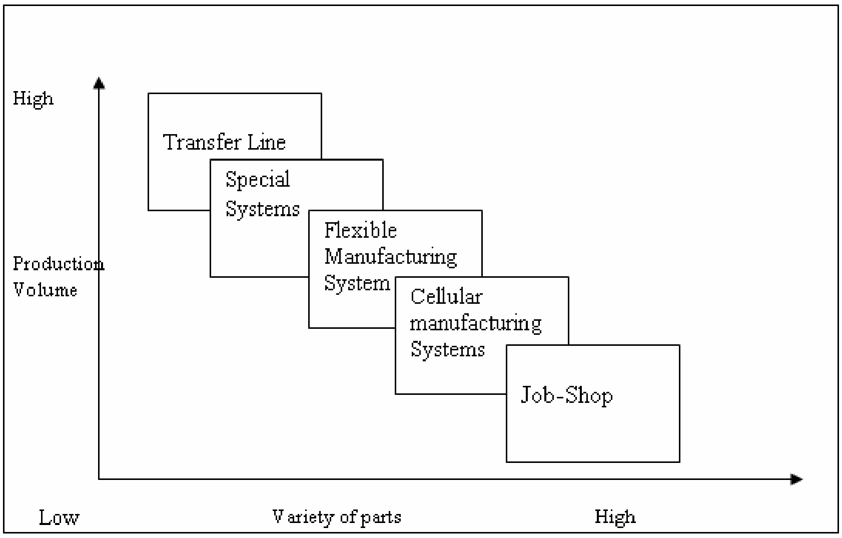


Figure 1.4 Applicability of Cellular Manufacturing

In conclusion, CM is a manufacturing system that can produce medium volume/medium variety part types more economically than other manufacturing systems (Black, 1983). Figure 1.4 shows the applicability of CM approach in terms of volume and variety of products. CM is a manufacturing strategy to global competition by reducing manufacturing costs, improving quality and by reducing the delivery lead times of products in a high variety, low demand environment. Hence CM has become popular among manufacturers in the last several decades.

1.2 Benefits of Cellular Manufacturing

The benefits of CM in comparison with traditional manufacturing systems in terms of system performance can be summarized as follows:

1. Material Handling: In Cellular Manufacturing a part is completely processed within a cell. Since the machines in a cell are located in a close proximity, part travel times and distance between machines are minimal.
2. Setup time: Since a manufacturing cell is dedicated to parts having similar design and manufacturing attributes, it is possible to use the same fixtures and tools. Generic fixtures for part family can be also developed and the time required for changing the fixtures can be reduced. The parts should also require similar tooling, which further reduces the setup time. In the press shop at Toyota, for example, workers routinely change dies in 3-5 minutes. The same job at GM may take 4-5 h. (Black, 1983).
3. Batch Size: In CM, since the setup times are greatly reduced, small lot production becomes economical. Small lots also smooth the production flow.
4. Work in process: In job shop, the economic order quantity for different parts varies due to the differences in setup and inventory costs. A level of stock up to 50% of annual sales is not unusual for batch production (Askin & Standridge, 1993). In CM, the WIP levels can be reduced with smaller lot sizes and reduced setup times.
5. Throughput time: In a traditional job shop, a part moves between different machines in a batch through its processing steps. However in CM, each part is immediately transferred to the next station after it has finished operation. Hence the waiting times are reduced.
6. Machine utilization: Since the setup times are decreased, the effective capacity of the machines is increased. This leads lower machine utilization. The general level of utilization of cells (except for key machines) is of the order of 60-70% (Nanua & Rajamani, 1996) . However this is not a disadvantage as is often stated. This is working smart and short. In job shops, the primary objective is to use the machines at full capacity. However undue

emphasis on high machine utilization results in excessive WIP and long throughput times.

7. Labor: Due to lower utilization in CM, it is possible to assign more than one machine to a worker. This leads job enrichment and also forms the basis for total quality management.
8. Quality: Since the parts are processed as single units (or small batches) and completed in a small region, the feedback is immediate and the corrective actions can be easily taken.
9. Space: Due to the decrease in WIP and finished goods, there will be floor space available for adding machines and for expansion.
10. Production control and scheduling: In a job shop, parts have to travel from one department to another through its processing steps. This results in complicated material control and scheduling. In CM, parts travel in a cell instead of the whole manufacturing plant. This results in easier scheduling and production control.

The benefits gained from implementing CM also have been reported. Northern Telecom, the leading supplier of digital communications systems applied CM to the DMS-100 Switching Division and gains more than $2 million in annual cost savings from the reduction of WIP inventory (by 82 %), as well as improvement in throughput (by more than 50%).

In an Indian engineering Company, the number of machines employed has been reduced from 120 to 94 and the shop floor space requirement is reduced by 21%.

Howard and Newman (1993), reported the results of implementing CM at PMI Food equipment Group. Some of the benefits included doubling of capacity for part families due to cell configuration, $25.000 in labor savings from setup reductions, over $ 2 million savings in finished goods inventory, improved customer service, and an improved quality of employee work life.

Table 1.1. Reported performance improvements from Cellular manufacturing (Wemmerlov &Johnson, 1997)

Performance Measure	Average % Improvement	Minimum % Improvement	Maximum % Improvement
1. Reduction of move distance/time	61.3	15.0	99.0
2. Reduction in throughput time	61.2	12.5	99.5
3. Reduction of response time to orders	50.1	0.0	93.2
4. Reduction in WIP inventory	48.2	10	99.7
5. Reduction in setup times	44.2	0.0	96.6
6. Reduction in finished goods inventory	39.3	0.0	100.0
7. Improvement in part/product quality	28.4	0.0	62.5
8. Reduction in unit costs	16.0	0.0	60.0

Wemmerlov and John (1997) conducted a survey in performance improvements of CM at 46 firms from different industries (electronic products and components, machinery and machine tools, heating and cooling products, tools, engines and bearings) Table 1.1 Shows the reported performance improvements. As seen form the results of this survey, the major improvements are achieved in part movement distance (times), throughput time, response time, WIP and set-up times.

CM is also considered a basis for Just-in-Time (JIT) manufacturing philosophy. CM is well suited for the JIT requirements such as little or zero setup time, working with small lot sizes and low WIP etc. Black (1983) emphasized that CM is the first critical step to achieve JIT manufacturing.

1.3 Design of Cellular Manufacturing Systems

The design of CMS has been called as cell formation (CF), part family/Machine cell formation and manufacturing cell design. The design of CMS is a complex, multi-criteria problem. This problem is NP-complete even under fairly restrictive conditions (Ballakur, 1985). The CF problem consists of two main tasks:

1. Part family formation: Parts are grouped into part families according to their processing requirements.

2. Machine cell formation: Machines are grouped into manufacturing cells.

These two tasks are not necessarily performed in the above order or sequentially. Part families and machine cells can be also formed simultaneously. After the above steps are completed, manufacturing cell configuration is obtained. Manufacturing cell configuration is constituted of machines which are dedicated to part families. The arrangement of machines within a cell is a layout design problem and is not considered in this study.

In the design of CMSs the design objectives must be specified. The design objectives can be classified as *cost oriented* and *performance oriented* (Mansouri, Moattar, Husseini and Newman, 2000). Cost oriented objectives are in the form of minimization. Common objectives are minimizing inter-cell movement costs, minimizing exceptional parts (parts that require processing from more than one cell). An example of seven machines and seven part types are used to describe the terminology. A part-machine matrix represents the processing requirements of parts on machines as shown in Figure 1.5. A 1 entry on row i and column j indicates that part type j has an operation on machine type *i.* For example part type 1 has operations on machines M2 and M5.

	P1	P2	P3	P4	P5	P6	P7
M1		1			1		
M2	1						1
M3			1	1		1	
M4			1	1		1	
M5	1						1
M6			1	1		1	
M7		1			1		

(a) Part / machine matrix

	P1	P7	P3	P4	P6	P2	P5
M2	1	1	1*				
M5	1	1					
M3			1	1	1		
M4			1	1	1		
M6			1	1	1		
M1						1	1
M7						1	0

(b) Optiml clustering

Figure 1.5 Part /machine matrix and optimal clustering

Three cells are formed according to part-machine matrix given in Figure 1.3. The first cell is composed of machines M2 and M5 and produces parts P1, P7. The second cell consists of machines M3, M4, M6 and produces P3, P4 and P6. Machines M1 and M7 constitute Cell3 and produces parts P2 and P5. However, part P3 has an operation on machine M2 which is assigned to cell 1. Therefore an inter-cell movement (from cell 2 to cell 1) is required for manufacturing part P3. The symbol (*) represents an inter-cell movement and part P3 called as "exceptional part" so these two machine cells are called "partially separable". The machine M2 is one that is required for more than one cell is called as bottleneck machine. The 0 s in the diagonal blocks are referred to as "voids". A void indicates that a machine assigned to a cell is not required for the processing of a part in the cell. For example in Table 1.3, machine M7 is not required for processing part P5 in cell 2. The presence of voids leads to inefficient large cells which leads additional intra-cell material handling costs and complex control requirements.

In addition to inter-cell material handling cost, other cost oriented objectives such as minimizing equipment costs, minimizing inventory costs, minimizing operating costs, minimizing machine relocation costs, minimizing machine duplication costs have been widely used in cell formation literature.

Performance oriented objectives can be in the form of maximization or minimization. Maximizing cell utilization, maximizing system throughput, minimizing cell workload unbalance, maximizing flexibility are the common performance oriented objectives in cell literature.

In the last two decades lots of research papers and practical reports have been published in the field of the design of CMSs. Reviews existing CM literature can be found in (Greene & Sadowski, 1984; Kamrani, Parsaie, and Chaudhry, 1993; Offodile, Mehrez, and Grznar, 1994; Joines, King and Culbreth, 1996; Agarwal &Sarkis, 1998; Shanker & Vrat, 1999; Mansouri et al., 2000; Pierreval, Caux & Viguier, 2003; Car & Mikac, 2006; Balakrishnan & Cheng, 2007) According to those reviews , the existing CM methods can be classified into following categories:

1. Array based techniques: Array based clustering is the most commonly used techniques in CF. These techniques operate on a 0-1 part machine matrix. A part /machine incidence matrix consists of elements a_{ij}=1 if part j requires

processing on machine *i*, otherwise a_{ij}=0. These techniques try to create small clustered blocks by performing a series of column and row manipulations on 0-1 part machine matrix (see Figure 1.5). In these techniques, part families and machine groups are formed simultaneously. Any tightly clustered blocks represent the candidate part families and machine groups. The most common array based techniques are Bond Energy algorithm (Mc Cormick, Schweitzer and White, 1972), Rank Order Clustering (King, 1980), Direct clustering (Chan & Milner, 1982)

2. Hierarchical Clustering techniques: These techniques operate on an input data set described in terms of similarity or distance function and produce a hierarchy of clusters or partition (Joines, King and Culbreth, 1996). Unlike the array based techniques, part families and machine cells do not form simultaneously. Since the similarity measures can incorporate manufacturing data other than 0-1 binary part machine matrix, lots pf similarity measures have been defined. These similarity measures are used to form part families and machine cells based on the methods such as single linkage cluster (Mc Auley, 1972), average linkage method (Seifoddini, 1986) complete linkage clustering (Mosier, 1989) etc.

3. Graph Theoretic approaches: These approaches structure the cell formation problem in the form of networks, bipartite graphs etc. In these methods, machines and/ or parts are represented by nodes, whereas processing of parts or similarity among machines are indicated by arcs. These approaches aim at obtaining disconnected sub-graphs from a machine-machine or machine-part graph in order to form machine cells and allocate part families to machines.
4. Mathematical programming approaches: These approaches have been widely used in the design of CMSs since they incorporate ordered sequence of operations, alternative process plans, setup and processing times etc. They can be further classified into four categories as linear programming (LP), linear and quadratic programming (LQP), dynamic programming (DP) and goal programming (GP).

5. Artificial Intelligence based methods: Researchers have applied AI techniques to design of CMS. Artificial neural networks, simulated annealing, tabu search, genetic algorithms, fuzzy logic are common AI based techniques applied to cell formation problems. These approaches are generally used as alternatives to mathematical programming approaches when computational time is prohibitive.

1.4 Important Issues in Designing CMS

Current CM design methods have some drawbacks. First is the lack of consideration of uncertainty in design parameters. Second is the lack of accounting for the presence of routing flexibility. Third is the lack of design methods that considers performance oriented objectives such as mean tardiness, average time in the system, percentage of tardy jobs etc. Fourth is the lack of usage of solution approaches that considers subset of non-dominated solutions from which the designer could select.

1.4.1 Uncertainty in Design Parameters

There are three basic types of models available for the analysis of manufacturing cells. (Kamrani et. al, 1998)

1. Static or deterministic models
2. Queuing models
3. Simulation models

Deterministic or static models use a set of deterministic equations in order to obtain gross estimates of system parameters such as utilization, capacity, throughput etc. Relaxations in modeling assumptions such as infinite production rates, certainty of cost factors, deterministic demand and deterministic processing time situations etc. affects the implementation of cellular manufacturing systems designs. Most of the current cell formation methods assume a static, deterministic production environment. However, real manufacturing systems tend to have uncertainty or vagueness in system parameters. Deterministic models are not able to provide good estimates of more probabilistic operating characteristics such as queue waiting time, machine breakdowns, demand fluctuations etc.

Queuing models are based on mathematical theory of queues. These models permit the involvement of some dynamic system characteristics such as fluctuations in queue levels. These characteristics are complex and probabilistic in nature however the queuing models predict these aspects of system performance only in a general way and for simple situations

There is a growing need to address some practical manufacturing considerations associated with uncertainty and vagueness in system parameters. Unfortunately a few works in the design of CMSs have addressed the uncertainty in design parameters. Seifoddini (1990) proposed a probabilistic modeling of CMS design by incorporating the probability of product mix. CMS design under uncertainty has been modeled through chance constrained programming by Shanker and Vrat (1996) for choosing the best strategy to deal with exceptional elements and bottleneck machines.

One of the most important tasks for complex organizations is to manage uncertainty. Many system parameters are difficult to capture by determinism, as traditionally considered in the mathematical programming approaches. Simulation is an useful tool for analyzing such systems. Dynamic and stochastic system characteristics can be incorporated into models easily. So a high degree of realism can be achieved. Since simulation is not an optimization tool, simulation studies performed in CF literature are generally focused on analyzing the performance of manufacturing cells (Gupta & Tompkins, 1982; Morris & Tersine, 1990; Shafer & Charnes, 1993; Kannan & Gosh, 1996; Lagendran & Talkington, 1997, Kamrani, Hubbard, Parsaei and Leep, 1998; De Los, Irrizary, Wilson and Trevino, 2001; Djasssemi, 2005). With the use of hybrid simulation-analytical optimization approaches, the stochastic nature of some system parameters (such as stochastic demand rate, processing times, material transport times etc) can be implied and more realistic CMS designs can be obtained.

Application of fuzzy sets to multi-objective optimization problems allows for handling linguistic vagueness in the estimates of system parameters. Fuzzy clustering techniques has been applied in the area of cell formation (Xu & Wang, 1989; Chu &Hayya,1991; Zhang &Wang, 1992; Gindy, Ratchev and Case, 1995; Gill &Bector, 1997; Susanto, Kennedy and Price,1999; Josien &Liao, 2000; Lozano, Dobado, Larrrenta and Onieva, 2002; Yang, Hungand Chen, 2006; Li, Chu, Wang and Yan, 2007). However a few works in the design of CMS have used fuzzy modeling in a mathematical programming framework. Fuzzy clustering problem is different from the fuzzy programming problem. In fuzzy clustering problems, fuzzy membership functions of a machine (and/or part) with respect to a cell (and/or part family) are defined and hierarchical clustering is performed to design CMSs. In fuzzy mathematical programming, linguistic vagueness in many other design parameters may be modeled and the solution is obtained by applying mathematical programming tools such as LP, GP etc. GP is one of the most powerful, multi-objective decision making approaches in practical decision making. This method requires the decision maker (DM) to set goals for each objective that he/she wishes to attain. In a standard GP formulation, goals and constraints are defined precisely. However, one of the major drawbacks for a DM in using GP is to determine precisely the goal value of each objective function. Applying fuzzy set theory (FST) into GP has the advantage of allowing for the vague aspirations of a DM. Goal Programming, as a mathematical programming tool, has been applied most to multi-criteria cell design problems (Gongaware & Ham, 1984; Sankaran, 1990; Shafer & Meredith, 1991, 1993; Min & Shin, 1993; Baykasoglu & Gindy, 1998). However, applying fuzzy goal programming to CF is a relatively new attempt (Tsai, 1996, 1997).

1.4.2 Routing Flexibility

Most of the current CF methods assume that each operation of a part can be processed only on a one specific machine type. This is not valid when machines are capable of performing different processes. The use of such machines results in alternate machine routings for each operation. When a part type is processed on a multiple routings, it is referred to as "routing flexibility" (Sethi & Sethi, 1990). In the presence of routing flexibility each part will have more than one process plan. In such situations the problem of "searching for the best routing" arises. The existence of alternative process plans for parts can improve the groupability of parts and increase the utilization of machines (Mungwattana, 2000). On the other hand the presence of routing flexibility also increases the number of ways to form manufacturing cells. Ignorance of routing flexibility may result in an increased operation cost and additional investment in machines (Defersha, 2006).

1.4.3 Performance Oriented Objectives

There have been many efforts towards the design of manufacturing cells considering only a single criterion such as minimizing inter-cell movements of parts. However, there has been a growing pressure on the today's manufacturing firms to improve their performances with regard to such measures such as shorter delivery lead times, wider range of products, shorter set-up times, lower prices etc. This leads to a number of conflicting criteria on which performance is evaluated. Thus the design of CMSs is critical to the efficient performance of the business. As stated in the previous section, the objectives used in cell design can be classified into two groups as cost oriented and performance oriented. Minimizing inter-cell and intra-cell costs, minimizing operating costs, minimizing setup costs are the most common cost oriented objectives. Maximizing utilization, minimizing cell load unbalance, maximizing flexibility are the most common performance oriented objectives used in CF literature. The performance oriented objectives such as minimizing mean tardiness, minimizing the percentage of tardy jobs, minimizing average time spent in the system etc. are also important for the manufacturing systems which operate under just-in time manufacturing philosophy. However, such objectives are not considered by the most current CF approaches probably due to the complexity of the general CF problems. Analytic representation of such objectives is difficult and also leads to computationally complex models which are not practical for real applications. The hybrid analytic-simulation models in which some of the objectives are obtained by simulation model can be employed to overcome such difficulties.

1.4.4 Solution Approaches That Consider a Subset of Non-Dominated Solutions

The most common solution approaches in the multi-criteria problems are weighting method and goal programming. These approaches are able to find a single non-dominated solution. If the solution is not good enough to satisfy the requirements of the system designer, the model should be resolved with different set of parameters. The application of other solution approaches that works with a rich subset of true non-dominated solutions allows for dealing with many alternative solutions from which system designer can select. Employing meta-heuristic approaches like genetic algorithms, tabu search, simulated annealing etc. may be an efficient way to find non-dominated solutions. Tabu search (TS) (Gloover, 1989) is a global optimization heuristic and can handle any type of objective function and any type of constraints. The solution process of TS involves working with more than one solution (neighborhood solutions) at a time. Baykasoglu (1999, 2002), noted that this feature of TS gives a great opportunity to deal with multiple objectives or goals simultaneously.

1.5 Problem statement

Although the benefits of CM are substantial current cell formation methods have some shortcomings as mentioned in previous section. The motivation of this research is to develop a novel cell design method considering the stochastic production requirements and the existence of routing flexibility.

Research question: How can we design cellular manufacturing systems in stochastic production environments, exploiting the routing flexibility?

1.6 Research objectives

A new design methodology that addresses the problems discussed in section 1.4 is needed. The objectives of this thesis can be summarized as follows:

1. Develop a design methodology for cellular manufacturing systems in stochastic production environment which employs routing flexibility.

2. Consider multiple performance oriented objective combinations of minimizing inter-cell movements, minimizing mean tardiness, minimizing percentage of tardy jobs, maximizing utilization etc. which have not been considered by current CF approaches.

1.7 Research Approach

In this study, to achieve the development of the new CM design methodology that addresses the problems discussed in section 1.4, a hybrid simulation-analytic fuzzy goal programming model (FGP) is developed. In this model, the achievement levels of goals which are difficult to represent analytically are obtained by simulation model whereas the achievement levels of other goals are calculated analytically. The stochastic nature of the manufacturing system is also reflected by simulation model. Part demand rates, part processing and transfer times are all stochastic. Proposed hybrid simulation-analytic FGP model is solved by using simple tabu search algorithm.

The research approach consists of the following steps:

1. Mathematical representation of the CF problem.

2. Implementing tabu search procedure for solving fuzzy goal programming models.

3. Development of computer program of tabu search procedure for solving fuzzy goal programming models.

4. Application of tabu search procedure for solving fuzzy goal programming CF models (with analytic objectives and deterministic design parameters)

5. Comparison of TS solutions with LINGO solutions for validation of the proposed TS procedure for CF.

6. Integration with simulation model.

7. Application of tabu search procedure for solving hybrid analytic-simulation fuzzy goal programming models.

8. Solution of different sized example problems for examining the performance of the proposed method.

9. Solution of problems from literature

10. Drawing conclusions and discussion of the future work.

1.8 Outline of Document

The remainder of this dissertation is organized as follows. Chapter 2 reviews the existing cell formation literature. Chapter 3 divided into two main sections. The first section gives a brief explanation about fuzzy mathematical programming, fuzzy linear programming and fuzzy goal programming. In the second section, a hybrid analytic-simulation fuzzy goal programming model is proposed for cell formation.

Chapter 4 presents a tabu search based solution approach. The applicability of the solution approach is tested on several deterministic test problems. In Chapter 5, the tabu search based solution approach is extended to solve hybrid analytic-simulation fuzzy goal programming models for cell formation. Finally Chapter 6 presents the conclusions, contributions and future research.

CHAPTER TWO

LITERATURE REVIEW

The survey of literature is divided into areas that seem to major impact in defining and solving the problem. These areas are:

1) Design approaches to multi-criteria cell formation
2) Uncertainty issues in cell formation
3) Tabu Search in cell formation

2.1. Design Approaches to Multi-Criteria Cell Design

As stated in the previous chapter, the identification of part families and machine groups in the design of cellular manufacturing is referred to as "cell formation" or "manufacturing cell design". Lots of models and solution approaches have been developed to deal with the problem of cell formation since 1980s. There have been many efforts to the design of manufacturing cells considering only one criterion such as minimizing inter-cell movements or maximizing parts (or machines) similarities etc. However, under the pressure of worldwide competition, today's manufacturing industries should improve their performances with regard to performance measures such as shorter delivery times, lower costs, lower setup times, wider range of products, shorter lead times etc. The pressure forces the manufacturing firms to evaluate a number of conflicting criteria in cell formation decisions. Hence the cell formation decisions depend on several criteria.

From a system designer's point of view, it is important to reach an optimal solution with respect to the all criteria considered. However it may be impossible since some of these criteria are contradictory.

For example, we can increase the number of machines to reduce the part traffic between cells and to create independent cells. But, increasing the number of machines may increase operating costs and may lead to lower utilization. Thus, the new solution will be good for the part flow criterion; however, this solution will be worse for the other criteria such as cost. The difficulty in the multi-criteria problem is to find a consistent cost function representing a single measure of quality for a solution. A value for each criterion to be optimized can be computed and the difficulty is then to choose a solution, which ‘‘is good for each criterion’’. Moreover, each criterion may have a particular importance, expressed by weight. These types of problems are known as multi-criteria decision problems. The design of manufacturing cells considering multiple criteria has been an attractive research area.

Lots of studies have been performed in the research area of cellular manufacturing systems since 1970s. There have also been some comprehensive review papers for cell formation. Wemmerlov and Hyer (1986) reviewed 70 papers and categorized them into two main groups based on the main data for grouping as either part attributes or part routings. Chu (1989) provided a comprehensive literature survey for cell formation and partitions the literature into design oriented and production oriented approaches. The production oriented approaches are further partitioned into array based, hierarchical, non hierarchical, mathematical, graph theoretic and heuristic approaches. Offodile et al (1994) divided all the methods for identifying part-machine families into three groups as visual methods, parts coding and analysis and production flow analysis. The models in the latter class are further divided into sub-groups considering their solution approach, decision variables, objectives and constraints. Joines et al. (1996) provided a comprehensive review and classification of techniques to manipulate part routing sequences for manufacturing cell formation. The cell formation approaches are aggregated into methodological groups including array-based methods, hierarchical clustering, non-hierarchical clustering, graph theoretic approaches, artificial intelligence based approaches, mathematical programming, and other heuristic approaches. Selim et al. (1998) provided a review based on solution methods. They classified cell formation studies into five main groups as descriptive procedures, cluster analysis, graph partitioning, artificial intelligence, mathematical programming. Mansouri et al. (2000) provided a review and comparison of the approaches to multi-criteria decision making in the design of manufacturing cells. The authors reviewed selected papers and a structured scheme is outlined which allows comparison of inputs, criteria, solution approaches and outputs across selected models.

2.1.1 Review of the Papers

In this section, 32 selected papers which consider the cell formation problem as a multi criteria decision-making problem will be reviewed briefly. The main criterion for the selection of papers is the consideration of at least two criteria simultaneously in the solution approach of the model. Hence the studies based on single criterion are not included in this review. The classification schema used in this section is partially influenced by the works of Mansouri et al (2000) and Joines et al. (1996). The comparison of the multi-criteria cell design models is carried out their inputs, criteria and solution approaches as in the work of Mansouri et. al (2000). The sub-groups of solution methods are structured as in the work of Joines et al (1996). The brief review of the selected papers is given below:

Wei and Gaither (1990) developed a four objective cell formation model to minimize the bottleneck cost, maximize average cell utilization, minimize intra-cell load imbalances and minimize inter-cell load imbalance. The authors developed a 0-1 programming model to solve small problems and a heuristic to solve larger problems.

Sankran (1990) developed a cell formation procedure considering multiple goals. The set of goals considered by the author includes minimum similarity of parts based on their needed machines and tools (two goals), available machining capacity, minimum and maximum number of total parts movements (two goals), the optimal capital investment on machines and the optimal operating cost.

Shafer and Rogers(1991) applied goal programming in three unique situations: setting up an entirely new system and purchasing all new equipment, reorganizing the system using only existing equipment, and reorganizing the system using existing and some new equipment. The criteria considered in this study are minimizing set-up times, minimizing intercellular movements, minimizing investment in new equipment, and maintaining an acceptable level of machine utilization level. The proposed goal programming models combine the p-median (for identifying part families) and the traveling salesman problem (for determining the optimal sequence of parts).

Venegopal and Narendran (1992) proposed a bi-criteria mathematical model for cell formation. The authors consider the objectives of minimizing inter-cell moves and cell load variations. They used a genetic algorithm based solution approach in order to find a compromise solution.

Logendran (1993) developed a model to minimize total inter-cell and intra-cell movements of parts and to maximize cell utilization. These objectives are unified in a single objective by using weighting approach. The original model is formulated as a quadratic binary programming model and then converted into a linear binary programming model.

Min & Shin (1993) developed an integer goal programming model to form machine cells and human cells simultaneously. The goals are concerned with the level of parts similarity, available machine processing time, machine capabilities / operator skills matching and the difference between the wages of cells operators and the rest of the operators.

Gupta, Gupta, Kumar & Sundram (1995) developed a model that considers the minimization of inter-cell and intra-cell movements. The authors performed the part assignment procedure considering minimum acceptable level of machine utilization. They used a genetic algorithm in order to solve the model.

Liang and Taboun (1995) developed a bi-criterion non-linear programming model. The objectives of the model are to maximize system flexibility, and to maximize system efficiency. They used the weighting approach to unify the objectives. Then a heuristic which composed of two phases is proposed for solution.

Suresh, Slomp & Kaparthi. (1995) employed a hierarchical approach which consists three phase. In phase I, a neural network clustering technique is used to identify part families and machine groups. In phase II a mixed integer goal programming model is used to assign individual machines to specified cells. Phase III aims to satisfy conflicting goals of maximizing cell independence, minimizing the purchase of new equipment and maximizing the routing flexibility.

Boctor (1996) developed a mixed integer model for designing manufacturing cells. The objective function is composed of two cost terms i.e. machine duplication and inter-cell movement costs. The author unified these two conflicting goals through the weighting approach. Then simulated annealing is used for solution of the model.

Rajamani, Singh & Aneja (1996) developed a mixed integer programming model with the assumption of flexible process plans for parts. The objective of the weighted sum of three cost functions as sum of investment, process and material handling costs. Authors used a column generation scheme and a branch and bound technique for solution of the relaxed linear model.

Hoo & Moodie (1996) developed a two stage solution approach considering flexible part routings. In stage 1, part families are formed. In stage II, machines are allocated to the part families using a mixed integer programming model. The objective function of stage II is composed of three cost functions: operation costs, machine duplication costs and inter-cell movement cost. They used the weighting approach to unify these three cost functions.

Lee & Chen (1997) used a three stage solution methodology which determines machine cells and part families and allows for machine duplication when necessary. They employed a weighting approach to combine two criteria i.e. minimizing inter-cellular movements and maximizing workload balance among duplicated machines.

Su & Hsu (1998) proposed a mathematical model considering three objectives. These objectives are minimizing total cost of inter-cell part transportation, intra-cell part transportation and machine investment; Minimizing intra-cell load imbalance; and minimizing inter-cell load imbalance. The authors unified these three objectives through weighting method and solve the model by means of parallel simulated annealing.

Vakharia & Chang (1999) developed a multi objective model considering the objectives of total cost of the machines and material handling cost. They used tabu search and simulated annealing for solution of the model and compared the results.

Shanker & Vrat (1998) presented fuzzy goal programming models for the design of cellular manufacturing systems. A multi-objective formulation is also presented to handle informational vagueness. The objective function minimizes the total costs associated with exceptional elements and bottleneck machines, such as subcontracting cost, inter-cell transfer cost and discounted cost of machines acquired.

Lozano, Gerrero, Eguia & Onieva (1999) proposed a two-phased approach for cell design and loading in the presence of alternative routing. In the first phase, machine cells are created using two different alternative clustering approaches. In the second phase, cell loading problem is modeled as a multi-period linear programming model. They used objective function that minimizes the total inter-cellular transportation cost and inventory holding cost.

Zhao & Wu (2000) presented a genetic algorithm approach for cell formation with the multiple objectives: minimizing costs due to inter-cell and intra-cell part movements; minimizing cell load variations; and minimizing exceptional elements. Manufacturing cells are formed based on production data, e.g. part routing sequence, production volume and workload.

Baykasoglu & Gindy (2000) proposed a preemptive goal programming model for cell formation problem considering the objectives: minimizing dissimilarity of parts, maximizing capacity, minimizing cell load imbalance and maximizing flexibility. They solved the model specially developed tabu search algorithm.

Suresh & Slomp (2001) proposed a hierarchical methodology for the design of manufacturing cells which includes labor-grouping considerations in addition to part/machine grouping. The procedure includes three phases. In Phase I, part families and associated machine types are identified through neural network methods. Phase II involves a prioritization of part families identified, along with adjustments to certain load-related parameters. Phase III involves interactive goal programming for regrouping machines and labor into cells. In machine grouping, factors such as capacity constraints, cell size restrictions, minimization of load imbalances, minimization of inter-cell movements of parts, minimization of new machines to be purchased, provision of flexibility, etc. are considered. In labor grouping, the functionally specialized labor pools are partitioned and regrouped into cells. Factors such as minimization of hiring and cross-training costs, ensuring balanced loads for workers, minimization of inter-cell movements of workers, providing adequate levels of labor flexibility, etc. are considered in a pragmatic manner.

Saad, Baykasoglu & Gindy (2002) developed an integrated framework for reconfiguring manufacturing cells. The cell creation module of the framework is integrated with the simulation model of the manufacturing system. Authors used a hybrid analytic-simulation preemptive goal programming model for cell formation. In this model some objectives are calculated analytically whereas other objectives are obtained by the simulation model. The goals of the model are: acceptable level of inter-cell movement, acceptable level for tardiness, desired level of overall system utilization, desired level of system throughput.

Khoo, Lee & Yin (2003) developed a genetic algorithm based solution approach for solving cell formation problems subject to objective functions such as gross part movement (inter-cell and intra-cell), cell load variations and machine set-up costs.

Jayaswal & Adil (2004) developed a mathematical model that incorporates operation sequences, alternative routings, cell size, production volume and allocating units of identical machines into different cells. The objectives of the proposed model are minimizing the sum of costs of inter-cell moves, machine investments and machine operating costs. They used simulated annealing for solution of the proposed model.

Solimanpur, Vrat & Shankar (2004) proposed a multi-objective integer programming model for cell formation problem considering different process plans for parts. The objectives of the models are to maximizing total similarity between parts, minimizing total processing cost, minimizing total processing time, and minimizing total investment needed for acquisition of machines. They used weighting approach to unify these objectives and solved the models using genetic algorithms.

Kim, Baek and Baek (2004) proposed a two phase heuristic to deal with multi-objective cell formation problem. The objective is to minimizing inter-cell part movements and machine workload imbalance. Together with the objective function, alternative part routes and machine sequences of part routes are considered. Part routes are determined in phase 1 using the look-ahead method. Machine groups are constructed in phase 2 using the maximum density rule.

Mehrabad & Safeai (2005) proposed a nonlinear integer model of cell formation under dynamic conditions. Authors applied a neural approach based on mean filed theory for solving the proposed model. In this approach, the network weights are updated by an interaction procedure. The objective function is a cost function which is the sum of machine amortization, inter-cell material handling cost, and machine relocation cost.

Vin, De Lit & Delchambre (2005) proposed an integrated approach based on a multiple objective genetic algorithm for solving cell formation problems with the presence of alternative routes and machine capacity constraints. The main objective is to minimize the inter-cellular traffic while respecting machine capacity constraints.

Lei and Wu (2006) presented a Pareto optimality based multi-objective tabu search algorithm to the cell formation problems with multiple objectives: minimizing the weighted sum of inter-cell and intra-cell moves, and minimizing total cell load variation.

Tsai, Chu, & Wu (2006) developed a multi objective fuzzy mathematical programming model for cell formation problems. The first objective of the model is a cost function that is the sum of cost of duplicating machines, inter-cell part transfer and subcontracting. The second objective is to maximize similarity between a pair of parts and machines. They also proposed a heuristic genetic algorithm for solving large scale fuzzy multi-objective cell formation problems.

Hu & Yasuda (2006) proposed a genetic algorithm approach in order to solve the cell formation problems with alternative routes. The objective function is composed of the weighted sum the amount of inter-cell and intra-cell movements. The proposed genetic algorithm approach is also capable of solving cell formation problems without predetermination of the number of cells.

Mahesh & Srinivasan (2006) consider multiple objectives (minimization of cycle time for an equivalent part, minimization of cell workload imbalance, and minimization of total work content for an equivalent part) for incremental cell formation and develop a lexicographic based simulated annealing algorithm. The performance of the algorithm is tested over several data sets by taking different initial feasible solutions generated using different heuristics.

Defersha & Chen (2006) proposed a comprehensive mixed integer programming model for design of CMS based on tooling requirements of the parts and tooling available on the machine. The model incorporates dynamic cell configuration, alternative routings, lot splitting, sequence of operations, multiple units of identical machines, cell size limits. The objective function is a cost function which is the sum of machine maintenance and overhead costs, machine procurement cost, inter-cell material handling cost, machine operating cost, tool consumption cost, setup cost, machine relocation cost and part subcontracting cost.

2.1.2. Comparison of the Models and Research Direction

The comparison of the above models is carried out based on their inputs, criteria and solution approaches.

2.1.2.1 Comparison based on the inputs

The input data of the previously discussed models are divided into four categories as in the work of Mansouri et al. (2000): part related data, machine related data, constraints and general features. The data related to parts and machines are further categorized based on the type of data as: quantitative data and cost data. Figure 2.1shows the classification of these categories. According to this classification scheme, the input data of the previously discussed models are illustrated in Table 2.1. As seen from Table 2.1 the most common input data used in the models are required machines, production capacity of machines, maximum and minimum cell size and demand. Other relatively common inputs are the number of available machines, predetermined number of cells, fixed process flow of the parts, the inter-cell transportation cost and the acquisition cost of machines.

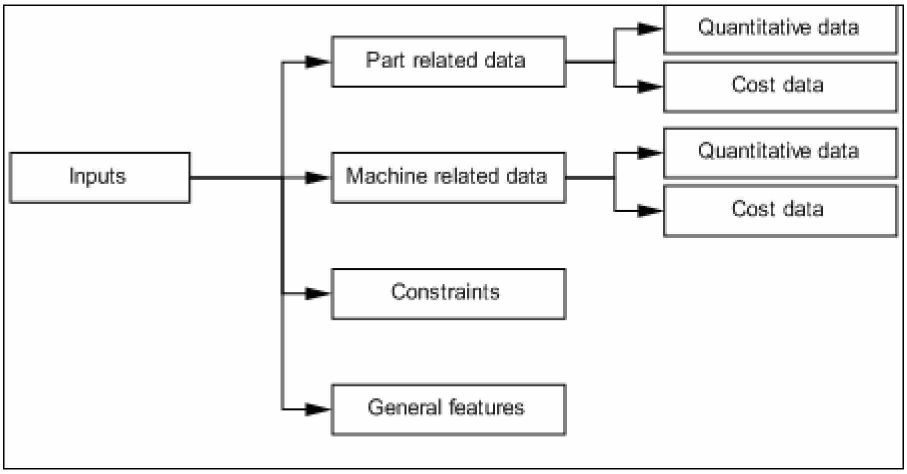


Figure 2.1 Classification of the input data (Mansouri et al, 2000)

As it could be traced in Table 2.1, fixed process flow has been becoming less important especially in recent studies mainly due to the increasing importance of flexibility that allows system designers to consider more alternatives in process route of the parts. Uncertainty issues in design parameters in and dynamic reconfiguration of cells are other features that which take attention from researchers in recent years. The studies that cover the uncertainty issues in cell formation will be discussed in Section 2.2.

2.1.2.1.1 General Considerations and Future direction: Input Data. Most of the reviewed studies depends on deterministic or static models in which a set of deterministic equations and inputs in order to obtain gross estimates of system parameters such as utilization, capacity, throughput etc. As stated in Chapter 1, relaxations in modeling assumptions such as infinite production rates, certainty of cost factors, deterministic demand and deterministic processing time situations etc. affects the implementation of cellular manufacturing systems designs. Majority of the methods assume a static, deterministic production environment. However, real manufacturing systems tend to have uncertainty or vagueness in system parameters. Deterministic models are not able to provide good estimates of more probabilistic operating characteristics such as queue waiting time, machine breakdowns, demand fluctuations etc.

	Inputs	Wei and Gaither (1990)	Sankran (1990)	Shafer and Rogers(1991)	Venegopal & Narendran (1992)	Logendran (1993)	Min & Shin (1993)	Gupta et al. (1995)	Liang & Taboun (1995)	Suresh et al. (1995)	Boctor (1996)	Rajamani et. al (1996)	Hoo and Modie (1996)	Lee and Chen (1997)	Su and Hsu (1998)	Vakharia and Chang (1999)	Shanker & Vrat (1999)	Lozano et al. (1999)	Zhau & Wu (2000)	Baykasoglu & Gindy (2000)	Suresh & Slomp (2001)	Saadet al. (2002)	Khoo, Lee and Yin (2003)	Jayaswal & Adil (2004)	Solimanpur et al. (2004)	Kim, Baek and Baek (2004)	Mehrabad &Safeai (2005)	Vin et al. (2005)	Lei and Wu (2006)	Tsai et al. (2006)	Hu & Yasuda (2006)	Mahesh & Srinivasan (2006)	Defersha &Chen (2006)
Part related data	Demand	*	*	*	*					*	*	*	*	*		*	*	*	*	*		*		*	*		*	*	*	*	*	*	*
	Required machines	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*		*	*	*	*	*	*	*	*	*	*	*	*
	Required tools		*																														
	Fixed process flow			*		*		*		*					*	*	*				*		*		*		*		*				*
	Alternative process flows											*	*					*	*			*		*		*		*			*		
	Alternative machine for oper.				*		*			*																						*	
	Process times	*	*	*		*			*	*	*	*	*	*	*	*	*	*	*			*	*	*	*	*	*	*	*		*	*	*
	Set-up times								*					*	*																		
	Batch size			*					*					*													*	*					*
	Direct product cost		*				*					*	*											*									
	Tool usage cost		*																														*
	Operator wage						*																										
	Subcontracting cost				*												*												*				*
	Inter-cell transport cost		*		*						*	*	*		*	*	*	*									*		*		*		*
	Intra-cell transport cost														*		*														*		
	Relocation/installation cost																										*						*
	Set-up cost																						*										

Table 2.1. Comparison of the models based on input data

	Inputs	Wei and Gaither (1990)	Sankran (1990)	Shafer and Rogers(1991)	Venegopal & Narendran (1992)	Logendran (1993)	Min & Shin (1993)	Gupta et al. (1995)	Liang & Taboun (1995)	Suresh et al. (1995)	Boctor (1996)	Rajamani et. al (1996)	Hoo and Modie (1996)	Lee and Chen (1997)	Su and Hsu (1998)	Vakharia and Chang (1999)	Shanker & Vrat (1999)	Lozano et al. (1999)	Zhau & Wu (2000)	Baykasoglu & Gindy (2000)	Suresh & Slomp (2001)	Saadet al. (2002)	Khoo, Lee and Yin (2003)	Jayaswal & Adil (2004)	Solimanpur et al. (2004)	Kim, Baek and Baek (2004)	Mehrabad &Safeai (2005)	Vin et al. (2005)	Lei and Wu (2006)	Tsai et al. (2006)	Hu & Yasuda (2006)	Mahesh & Srinivasan (2006)	Defersha &Chen (2006)
Machine data	Number of machines			*		*		*	*	*	*	*				*	*	*	*	*	*	*	*		*	*	*	*	*	*	*		*
	Production capacity	*	*	*	*	*	*	*	*		*	*	*	*	*	*	*	*	*	*	*		*		*	*	*		*				*
	Available tools		*																														
	Acquisition cost		*	*	*						*	*	*				*				*				*		*		*				*
	Operation cost		*										*			*									*								
Constraints	Fixed number of cells	*	*	*		*					*							*	*		*		*						*		*		
	Interval of cell numbers							*																									
	Maximum utilization															*		*										*					
	Minimum utilization							*		*																		*					
	Maximum cell size	*	*	*		*	*		*	*	*			*	*	*				*	*	*		*	*		*						
	Minimum cell size			*						*	*									*	*	*											
	Operator limitations						*																										
	Maintenance cost																																*
	Funds available			*													*																
General features	Dynamic cell reconfiguration																	*				*					*						*
	Identical machines																												*				
	Operator skills						*																										
	Sequence of operations																		*														
	Arrangement of mac. in cells														*																		
	Arrangement of cells in plant												*		*																		
	Uncertainty in design parameters																*	*				*					*			*			*

Table 2.1. (Cont.) Comparison of the models based on input data

As can be seen from Table 2.1, the demand fluctuations have becoming more important in the studies performed after 90s. However the stochastic nature of other input data such as processing times, part transfer times etc. is not addressed in most of the studies. There is a growing need to address some practical considerations associated with the stochastic nature of production environment which directly affects the implementation of cellular manufacturing systems.

Flexibility in process flows is significant feature of modern manufacturing systems. Most of the current CF methods assume that each operation of a part can be processed only on a one specific machine type. However, this is not valid when machines are capable of performing different processes. The flexible machines which are capable of different operation are generally employed in today's manufacturing systems. Hence the use of such machines results in alternate machine routings for each operation. When a part type is processed on a multiple routings, it is referred to as "routing flexibility" (Sethi & Sethi, 1990). In the presence of routing flexibility each part will have more than one process plan. In such situations the problem of "searching for the best routing" arises. As stated in previous chapter, the existence of alternative process plans for parts can improve the groupability of parts and increases the number of ways to form manufacturing cells. Ignorance of routing flexibility may result in an increased operation cost and additional investment in machines. Hence it is important to consider such an important factor by quantifying alternative routes and flexible machining processes in the process of cell formation. As can be traced from Table 2.1, the existence of alternative process plans and machines takes much attention in the recent cell formation literature.

2.1.2.2. Comparison Based on the Criteria

The criteria used in reviewed models have been classified by the authors into "objectives" and "goals". The objectives can be classified as "cost oriented" and "performance oriented". Cost oriented objectives are in the form of minimization. Performance oriented objectives can be in the form of minimization or maximization.

The goal programming models aim to minimize deviations from predetermined goals. The classification scheme of objective and goals are given in Figure 2.2.

Table 2.2 gives the comparison of the models based on objectives / goals. Minimizing the machine duplication cost, minimizing the inter-cell transport cost, minimizing the number of inter-cell movements, minimizing the cell load imbalance are the most common objectives. Objectives such as minimizing the machine duplication cost, minimizing intra-cell transportation cost and maximizing flexibility take attention from researchers.

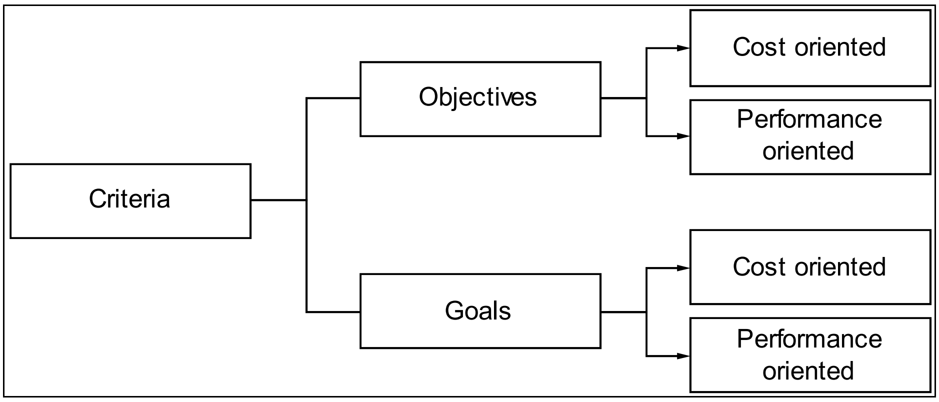


Figure 2.2. Classification of criteria

2.1.2.2.1 General Considerations: Criteria-Objectives/Goals. Majority of the research on cell design aims to develop models with a different combination of objectives that have not been considered previously. The performance oriented objectives such as minimizing mean tardiness, minimizing the percentage of tardy jobs, minimizing average time spent in the system etc. are important especially for today's manufacturing systems which operate under just- in time manufacturing philosophy. However, such objectives are not considered by the most current CF approaches probably due to the complexity of the general CF problems. Analytic representation of such objectives is difficult and also may lead to computationally complex models which are not practical for real applications. Developing more efficient solution tools enabling system designers to consider such objectives and to achieve good solutions in a reasonable time is a possible trend for the research in cell formation.

2.1.2.3 Comparison Based on Solution Approaches

The solution approaches of the reviewed models are classified based on their solution approaches as in Figure 2.3.

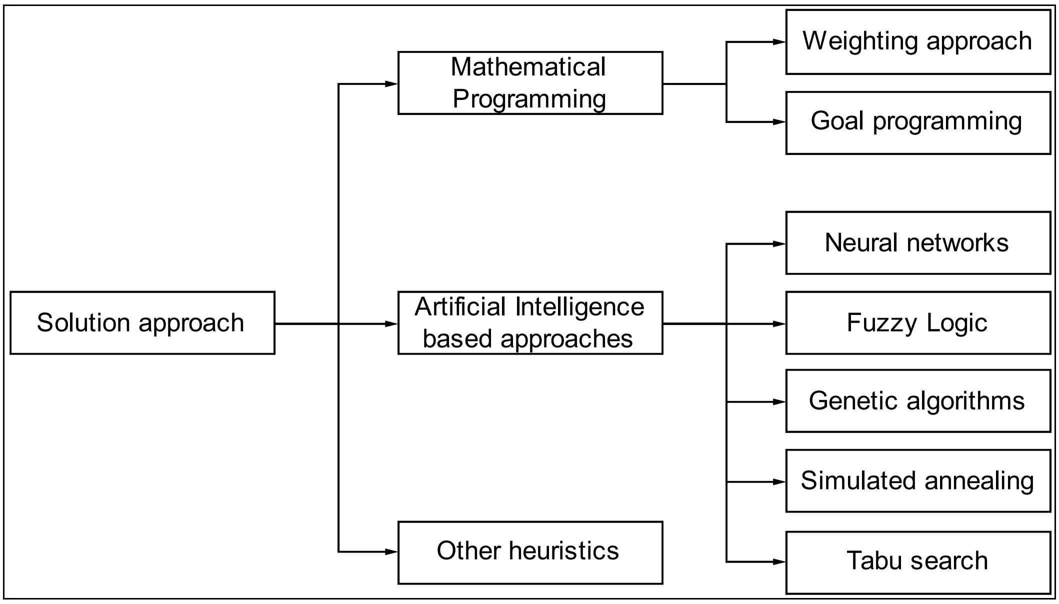


Figure 2.3 Classification of the solution approaches

The solution approaches used in the models are compared in Table 2.3. As seen from Table 2.3, Mathematical programming approaches are the most common solution approach. The mathematical programming methods weighting method and goal programming have been applied most to cell formation problems. Due to the complex nature of the cell formation problem, other search techniques such as genetic algorithms, simulated annealing, neural networks, tabu search etc. have been employed to solve large scale real world problems.

Objectives		Criteria	Wei and Gaither (1990)	Sankran (1990)	Shafer and Rogers(1991)	Venegopal & Narendran (1992)	Logendran (1993)	Min & Shin (1993)	Gupta et al. (1995)	Liang & Taboun (1995)	Suresh et al. (1995)	Boctor (1996)	Rajamani et. al (1996)	Hoo and Modie (1996)	Lee and Chen (1997)	Su and Hsu (1998)	Vakharia and Chang (1999)	Shanker & Vrat (1999)	Lozano et al. (1999)	Zhau & Wu (2000)	Baykasoglu & Gindy (2000)	Suresh & Slomp (2001)	Saadet al. (2002)	Khoo, Lee and Yin (2003)	Jayaswal & Adil (2004)	Solimanpur et al. (2004)	Kim, Baek and Baek (2004)	Mehrabad &Safeai (2005)	Vin et al. (2005)	Lei and Wu (2006)	Tsai et al. (2006)	Hu & Yasuda (2006)	Mahesh & Srinivasan (2006)	Defersha &Chen (2006)
Objectives	Min cost of	Machine duplication										*	*	*		*	*	*							*	*		*			*			*
		Operation cost											*	*											*	*								
		Part subcontracting	*															*										*			*			
		Inter-cell transportation	*									*	*	*		*	*	*	*	*				*	*						*			
		Intra-cell transportation														*	*			*				*										
		Machine relocation cost																										*						*
		Inventory holding cost																	*															
		Machine setup cost																						*										*
		Machine maintenance cost																																*
		Tool consumption cost																																*
	Min. Amount of	Duplicated machines																				*												
		Inter-cell movements				*	*		*		*				*							*					*		*	*		*		
		Intra-cell transportation					*		*		*																		*	*		*		
		Exceptional elements																		*														
		In cell load imbalance	*			*										*				*	*	*					*			*			*	
		In plant load imbalance	*												*	*																		
		Part dissimilarity																			*													
		Part processing time																							*									
		Part cycle time																															*	
		Total work content																															*	

Table 2.2 Comparison of the models based on criteria

Goals		Criteria	Wei and Gaither (1990)	Sankran (1990)	Shafer and Rogers(1991)	Venegopal & Narendran (1992)	Logendran (1993)	Min & Shin (1993)	Gupta et al. (1995)	Liang & Taboun (1995)	Suresh et al. (1995)	Boctor (1996)	Rajamani et. al (1996)	Hoo and Modie (1996)	Lee and Chen (1997)	Su and Hsu (1998)	Vakharia and Chang (1999)	Shanker & Vrat (1999)	Lozano et al. (1999)	Zhau & Wu (2000)	Baykasoglu & Gindy (2000)	Suresh & Slomp (2001)	Saadet al. (2002)	Khoo, Lee and Yin (2003)	Jayaswal & Adil (2004)	Solimanpur et al. (2004)	Kim, Baek and Baek (2004)	Mehrabad &Safaei (2005)	Vin et al. (2005)	Lei and Wu (2006)	Tsai et al. (2006)	Hu & Yasuda (2006)	Mahesh & Srinivasan (2006)	Defersha &Chen (2006)
Goals	Max. Amount of	Flexibility								*	*										*	*												
		Efficiency								*																								
		Utilization	*				*																											
		Cell independence									*																							
		Machine capacity																			*													
		Part similarity																							*						*			
	Min. deviation from	Part similarity		*				*																										
		Mac. available process time		*				*																										
		Mac. operator skill matching						*																										
		Cell operator wages						*																										
		Maximum part movements		*																														
		Minimum part movements		*																														
		Machine investment cost		*																			*											
		Operating cost		*																														
		Set-up time			*																													
		Utilization			*																		*											
		Available funds			*																													
		Inter-cell moves			*																													
		System throughput																					*											
		Total tardiness																					*											

Table 2.2 (cont.) Comparison of the models based on criteria

Other than the mathematical programming techniques, most cell formation methods are heuristics. However, most of those methods have been placed into the aggregate category of AI based approaches. Other than these heuristics are grouped into the class of "other heuristics".

2.1.2.3.1 General Considerations: Solution Approaches. The most common solution approach in the mathematical programming is the weighting method. These approaches are able to find a single non-dominated solution. If the solution is not good enough to satisfy system designer needs, the model should be resolved with different system parameters. However, in multi-objective optimization problems, it is essential to work with multiple non-dominated solutions where system designer could select the most suitable solution among a subset of non-dominated alternative solutions. The use of solution techniques such as genetic algorithms, tabu search etc. which work with more than one solution at a time is a new trend for solving multi-objective cell formation problems. The use of fuzzy set theory in cell formation is another potential research area in cell formation. The detailed literature review of tabu search and fuzzy set theory in cell formation will be given in the following sections.

2.2 Uncertainty Issues in Cell Formation

Most of the current cell formation methods assume a static, deterministic production environment. However, real manufacturing systems tend to have uncertainty or vagueness in system parameters. Deterministic models are not able to provide good estimates of more probabilistic operating characteristics such as queue waiting time, machine breakdowns, demand fluctuations etc. Deterministic or static models use a set of deterministic equations and inputs in order to estimate system parameters such as utilization, capacity, throughput etc. Relaxations in modeling assumptions such as infinite production rates, certainty of cost factors, deterministic demand and deterministic processing time situations etc. affects the implementation of cellular manufacturing systems designs.

Solution Approaches		Wei and Gaither (1990)	Sankran (1990)	Shafer and Rogers(1991)	Venegopal and Narendran (1992 a)	Logendran (1993)	Min & Shin (1993)	Gupta et al. (1995)	Liang and Taboun (1995)	Suresh et al. (1995)	Boctor (1996)	Rajamani et. al (1996)	Hoo and Modie (1996)	Lee and Chen (1997)	Su and Hsu (1998)	Vakharia and Chang (1999)	Shanker & Vrat (1999)	Lozano et al. (1999)	Zhau & Wu (2000)	Baykasoglu and Gindy (2000)	Suresh & Slomp (2001)	Saad, Baykasoglu and Gindy (2002)	Khoo, Lee and Yin (2003)	Jayaswal & Adil (2004)	Solimanpur et al. (2004)	Kim, Baek and Baek (2004)	Mehrabad &Safaei (2005)	Yin, De Lit and Delchambre (2005)	Lei and Wu (2006)	Tsai et al. (2006)	Hu & Yasuda (2006)	Mahesh & Srinivasan (2006)	Defersha &Chen (2006)
Mathematical Programming Approaches	Weighting method	*		*		*			*		*	*	*		*			*			*	*		*	*								*
	Goal programming		*				*			*																							
Artificial Intelligence Based Approaches	Fuzzy Logic																*													*			
	Genetic algorithms				*			*											*				*		*			*		*	*		
	Tabu search															*				*		*							*				
	Simulated annealing										*				*	*								*								*	
	Neural networks									*											*						*						
Other Heuristics							*						*					*			*					*							

Table 2.3 Comparison of the models based on solution approaches

In recent years, uncertainty issues in cell formation process have been popular among the researchers. In this section, the studies that cover uncertainty issues in cell formation are reviewed.

There is a growing need to address some manufacturing considerations associated with uncertainty and vagueness in system parameters. Unfortunately a few works in the design of CMSs have addressed the uncertainty in design parameters. Seifoddini (1990) proposed a probabilistic modeling of CMS design by incorporating the probability of product mix. CMS design under uncertainty has been modeled through chance constrained programming by Shanker and Vrat (1996) for choosing the best strategy to deal with exceptional elements and bottleneck machines.

Most of the recent studies have focused on multi period cell creation and loading according to demand fluctuations. However, in these studies, the demand levels of each period are considered as deterministic. Hence the stochastic production requirements are again omitted.

The stochastic nature of the manufacturing systems is generally reflected by queuing models and simulation. Queuing models are based on mathematical theory of queues and it allows for analyzing dynamic characteristics of manufacturing systems such as fluctuations in queue levels which are complex and probabilistic in nature. However queuing models estimates these characteristics only in a general way and for simple situations.

Another tool for dealing with uncertainty in design parameters is fuzzy sets. Application of fuzzy set theory to multi-objective optimization problems allows for handling linguistic vagueness of system parameters. Linguistic vagueness of fuzzy type has been modeled in many areas of production research. The cell formation studies using simulation and fuzzy sets will be reviewed in following sections.

2.2.1. Simulation Studies in Cell Formation

Simulation is a useful tool for analyzing the characteristics of manufacturing systems considering the stochastic nature of design parameters. Some researchers have focused on the effect of conversion from a job-shop into a cellular manufacturing system by using simulation. Gupta and Tompkins (1982) used computer simulation to study the effects of routing flexibility in response to product mix variations. Mahmoodi et al. (1990) examined the different order releasing policies in a cellular manufacturing environment to reduce workload imbalance. Morris and Tersine (1990) examined the effects of the ratio of setup to process time, the time to transfer materials between work centers, the variability of demand, and the flow of work within cells. Suresh (1992) and Durmusoglu (1993) addressed set-up time reduction as a key element for conversion. Kannan and Gosh (1996) compared cellular manufacturing to process layout under a wide range of different conditions. Logendran and Talkington (1997) perform a comprehensive study to compare the performance of cellular and functional layouts by considering two significant environmental factors: machine breakdowns and batch size. Abino and Garavelli proposed a simulation model to analyze costs and benefits of routing flexibility referring to the concept of limited flexibility. Shambu and Suresh (1998) compare the performance of hybrid cellular manufacturing systems (the manufacturing system that contains the characteristics of both functional and cellular manufacturing systems) and functional layout considering different scheduling rules under a variety of shop operating conditions. Djassemi (2005) examined the performance of cellular manufacturing systems in a variable demand and flexible work force environment using simulation modeling. They conclude that the practice of flexible cross trained operators can improve the flexibility of CMS in dealing with an unstable demand.

The simulation studies, discussed so far, focused on the performance comparison of cellular manufacturing systems and analyzing the effects of factors in cell formation. Simulation is not performed as a part of cell formation process in these studies. A few researchers construct simulation studies as the part of cell formation process. Kamrani et. al (1998) presented a simulation based methodology which uses both design and manufacturing attributes. The methodology includes three phases. In phase I, parts are grouped into part families based on design and manufacturing dissimilarities. Phase 2 groups the machines into machine cells based on operational costs. Phase I and phase 2 depend on integer and mixed integer mathematical programming. Finally in Phase III, simulation model of the proposed system is built and run in order to evaluate results obtained from phase I and Phase II. Irrizary, Wilson and Trevino (2002) present a general manufacturing-cell simulation model for evaluating the effects of world class manufacturing practices on expected cell performance. They formulated a comprehensive annualized cost function for evaluation of alternative cell designs. The authors also presented a two phase approach to design of manufacturing cells based on simulation experimentation and response surface methodology using a general manufacturing-cell simulation model.

In both studies, simulation models have not been included in cell formation phase. Simulation has been used to evaluate the performance of cell formation alternatives which are obtained by mathematical programming or heuristic approaches. Hence the stochastic nature of the manufacturing system has not been reflected in the design process of cellular manufacturing systems.

Saad, et al. (2002) developed an integrated framework for reconfiguring manufacturing cells. The cell formation module of the framework is integrated with the simulation model of the manufacturing system. Authors used a hybrid analytic-simulation preemptive goal programming model for cell formation. In this model some objectives are calculated analytically whereas other objectives are obtained by the simulation model. Hence the simulation is used as a part of cell formation process in this study. However uncertainty issues in design parameters are not addressed in this study. For example, the part demands for planning periods are taken deterministic.

Simulation is a useful tool for analyzing such systems. Dynamic and stochastic system characteristics can be incorporated into simulation models easily. Hence a high degree of realism can be achieved though the use of simulation. However simulation is not an optimization tool. As stated above, the simulation studies performed in the CF literature are generally focused on analyzing the performance of manufacturing cells. With the use of hybrid simulation-analytical optimization approaches, the stochastic nature of some system parameters (such as stochastic demand rate, processing times, material transport times etc) can be implied in design process of manufacturing cells and more realistic CMS designs can be obtained. Hence, in this dissertation, a hybrid analytic-simulation model will be proposed for cell formation.

2.2.2 Fuzzy Sets in Cell Formation

Another tool for dealing with uncertainty is fuzzy sets which allows for handling linguistic vagueness of system parameters. Linguistic vagueness of fuzzy type has been modeled in many areas of production research including cell formation.

Fuzzy clustering techniques have been widely used for cell formation since 1980s. Xu and Wang (1989) incorporated uncertainty in the measurement of similarity of parts by using fuzzy mathematics. Chu and Hayya (1991) used the fuzzy c-mean clustering to identify part families. Zhang and Wang (1992) provided a fuzzy version of single linkage clustering and rank order clustering. Ben Arieh and Trianthaphyllou (1992) used fuzzy set theory for data quantification in group technology. Gindy, Ratchev & Case (1995) used a fuzzy c-mean algorithm and defined a validity measure for cell formation. Gill and Bector (1997) proposed an approach based on fuzzy linguistics to quantify part feature information for cell formation problems. Josien and Liao (2000) presented an approach which integrates two fuzzy clustering techniques: fuzzy c-means and fuzzy k-nearest neighbors. Lozano et al. (2002) propose a modified fuzzy c-means algorithm that groups components and machines in parallel. Josien & Liao (2002) proposed an integrated approach that is capable of simultaneous classification of parts and machines. Yang, Hung & Cheng (2006) applied a mixed variable fuzzy clustering algorithm called “mixed-variable fuzzy c-means algorithm” to cell formation problems. Chu, Wang and Yen (2007) proposed an improved fuzzy c-mean algorithm for solving large scale cell formation problems.

None of the above referred works in the design of manufacturing cells has used fuzzy modeling in a mathematical programming framework. The fuzzy clustering is different from fuzzy mathematical programming. In fuzzy clustering problems, the fuzzy membership functions of a machine (and or part) with respect to a cell (and / or part family) is defined and hierarchical clustering is performed for designing manufacturing cells. On the other hand, in fuzzy mathematical programming models, linguistic vagueness in many other design parameters can be modeled and the solution can be obtained by application of mathematical tools such as linear programming, goal programming etc. In a fuzzy environment, fuzzy constraints, vague goals, and ambiguous parameters can be taken into consideration in the mathematical programming model. Several membership functions can be used to incorporate fuzziness for fuzzy objective functions or parameters (Zimmerman, 1991).

Although the fuzzy clustering techniques have been widely used in cell formation problems, the application of fuzzy mathematical programming approaches is a relatively new research area in cell formation. There are relatively limited works that use fuzzy mathematical programming models in cell formation literature. The following studies use fuzzy mathematical models for cell formation problems:

Shanker and Vrat (1998) developed fuzzy goal programming models for handling linguistic vagueness in the design of cellular manufacturing systems. A multi-objective formulation is also presented to handle informational vagueness. The objective function minimizes the total costs associated with exceptional elements and bottleneck machines, such as subcontracting cost, inter-cell transfer cost and discounted cost of machines acquired.

Szwarc, Rajamani and Bector (1997) proposed fuzzy mathematical models to optimally determine machine groupings and part assignment under fuzzy demand

and machine capacity. They consider a single objective function which is the sum of the costs of material handling and processing.

Tsai et al. (2006) developed a multi objective fuzzy mathematical programming model for cell formation problems. The first objective of the model is a cost function that is the sum of cost of duplicating machines, inter-cell part transfer and subcontracting. The second objective is to maximize similarity between a pair of parts and machines. They also proposed a heuristic genetic algorithm for solving large scale fuzzy multi-objective cell formation problems.

As discussed above, the use of fuzzy mathematical programming models in cell formation is a relatively new research area. Hence the use of fuzzy mathematical programming models in cell formation is a potential research area in the field of cellular manufacturing. The use of fuzzy goal programming models may be a possible trend for the research in this field.

In all these studies in which fuzzy mathematical programming approaches are employed, the cost oriented objective functions have been used. As stated in earlier sections, the design of manufacturing systems considering performance oriented objectives also important for today's manufacturing systems. Hence, employing fuzzy mathematical programming models with performance based objectives could be another new research area in cell formation. Moreover comparisons of the effectiveness of models using different fuzzy operators (simple additive, max-min, preemptive etc.) should be also addressed.

2.3 Tabu Search in Cell Formation

Tabu search (TS) is a stochastic neighborhood solution approach which was first proposed by Glover (1989, 1990). TS is a global optimization heuristic and can handle any type of objective function and any type of constraints. The basic TS algorithm operates in the following way: It starts with a randomly chosen or a known solution vector. A set of neighborhood solution N is generated from starting solution by using different move strategies. The objective function is evaluated for each neighbor solution in N. The best one replaces the current solution although it may be worse than the current one. In this way, algorithm escapes from local minima (or maxima). To avoid recycling, certain attributes of the last replaced solutions are stored in a list called "tabu list". The neighbors of the current solution given by the tabu list are eliminated unless they meet an aspiration criterion. Hence the algorithm is forced to select a point not recently selected. TS is an adaptive search procedure that has been employed for solving combinatorial optimization problems and cell formation problem a well. The cell formation studies in which TS algorithm is used for solution are briefly reviewed below:

Vakharia and Chang (1999) developed a multi objective model considering the objectives of total cost of the machines and material handling cost. They used TS and simulated annealing for solution of the model and compared the results.

Baykasoglu and Gindy (2000) proposed a preemptive goal programming model for cell formation problem considering the objectives: minimizing dissimilarity of parts, maximizing capacity, minimizing cell load imbalance and maximizing flexibility. They solved the model specially developed TS algorithm.

Onwubolu and Songore (2000) presented a TS based cell formation procedure which allows for the simultaneous creation of part families and machine cells. They consider two models one for minimizing cell load variation objective and one for minimizing inter-cell movements.

Saad, Baykasoglu and Gindy (2002) developed an integrated framework for reconfiguring manufacturing cells. The cell creation module of the framework is integrated with the simulation model of the manufacturing system. Authors used a hybrid analytic-simulation preemptive goal programming model for cell formation. In this model some objectives are calculated analytically whereas other objectives are obtained by the simulation model. The goals of the model are: acceptable level of inter-cell movement, acceptable level for tardiness, desired level of overall system utilization, desired level of system throughput. They solved the cell formation problem using TS algorithm.

Cao and Chen (2004) proposed an integrated approach for manufacturing cell formation with fixed charge cost. A mixed integer non-linear programming model is formulated to solve the problem. The NP-hardness of the problem makes direct solution computationally prohibitive for real-life applications. A TS algorithm was developed to solve the computationally complex cell formation problem efficiently.

Scheller (2005) proposed a two phased TS procedure for a cell formation integer programming model. Author used a cost function which composed of the cost of intra-cell and inter-cell moves and the cost of the equipment used in the cells.

Lei and Wu (2004) presented a Pareto optimality based multi-objective TS algorithm to cell formation problems with multiple objectives: minimizing the weighted sum of inter-cell and intra-cell moves and minimizing total cell load variation.

TS algorithm is generally used for the solving NP-Complete cell formation problems in most of the studies. However in recent studies (Baykasoglu et al. 2000, Saad et al 2002, Lei & Wu, 2004) TS algorithm is used for solving multiple objective cell formation problems. The idea of applying TS algorithm to multiple-objective optimization comes from its solution process. TS works with more than one solution at a time. Baykasoglu (2000, 2002) noted that this feature of TS gives a great opportunity to deal with multiple objectives or goals.

According to the reviewed literature, the primary reasons that TS has been used for the design of CMS are:

1) TS obtains efficient solutions in a reasonable time. According to the previous studies from different research areas, the solution quality of TS is also promising.

2) TS allows for handling any type of objectives and constraints.(Linear or nonlinear)

3) The feature of working with more than one solution at a time which is essential for dealing with multiple objective optimization problems.

These lead us to employ TS algorithm as a part of solution approach for the design of CMSs.

2.4 Limitations of the Existing Literature

In this section, limitations of the existing literature and how some of the limitations will be overcome by the proposed research will be discussed.

Cell formation literature is classified and reviewed in this chapter. Considering the coverage of the study, the survey of literature is performed into three categories: (1) Design approaches to multi-criteria cell formation. (2) Uncertainty issues in cell formation, and (3) Tabu search in cell formation. These comparisons and evaluations lead us to the following critical evaluation of the prior research:

1) Most of the current studies depend on deterministic models. Stochastic nature of the manufacturing systems is generally omitted. However, relaxations in modeling assumptions such as certainty of cost factors, deterministic demand, deterministic processing times etc. affects the implementation of cellular manufacturing systems. Such relaxations lead to cell designs that are far from meeting the requirements of real world applications. In this sense, simulation is a useful tool in representing stochastic nature of the manufacturing systems. However, in most of the studies simulation is not used in the design process of manufacturing cells. Instead, simulation is generally used for

evaluating and comparing the performance of cellular manufacturing systems under different production conditions and defining the effects of factors.

Fuzzy set theory is another tool for representing probabilistic and linguistic vagueness and uncertainty. The applications of fuzzy set theory in cell formation are common. However in most of these studies fuzzy clustering methods are used for identifying manufacturing cells and part families. Fuzzy mathematical programming is different form fuzzy clustering. In Fuzzy clustering approaches the fuzzy membership of a machine (or part) with respect to a cell (or part family) is defined and hierarchical clustering is done for designing CMS. On the other hand, by employing FMP models, linguistic vagueness in information pertaining to many other design parameters can be modeled and the solution may be obtained by using mathematical programming tools such as linear programming, goal programming etc. Application of fuzzy mathematical programming approaches to cell formation is a relatively new research area.

2) In most of the reviewed studies, a majority of the models used cost function unified through weighting approach in the form of single objective. Considering the requirements of today's manufacturing systems, performance measures such as mean tardiness, number of tardy jobs, system utilization etc. are also important for evaluating the performance of manufacturing cells. However, the combination of performance oriented objectives such as minimizing mean tardiness, maximizing utilization, minimizing the number of tardy jobs etc. have not been considered simultaneously in previous studies probably because the mathematical representation of such objectives may lead computationally complex models.

3) In some studies, it is assumed that each operation of a part can be processed only on a one specific machine type. However this assumption is not valid when machines are capable of performing different processes. The use of

flexible machines which can perform different operations results in alternate machine routings for each operation. In the presence of routing flexibility each part will have more than one process plan. In such situations the problem of "searching for the best routing" arises. According to the recent studies it is reported that the existence of alternative process plans for parts can improve the groupability of parts and increase the increases the number of ways to form manufacturing cells. It is also noted that the ignorance of routing flexibility may result in an increased operation cost and additional investment in machines. Hence it is important to consider such an important factor by quantifying alternative routes and flexible machining processes in the process of cell formation.

4) The most common solution approach in the mathematical programming models has been the weighting method and goal programming. These approaches are able to find a single non-dominated solution. If the solution is not good enough to satisfy the requirements of the system designer, the model should be resolved with different set of parameters. The use of meta-heuristics such as Tabu search, genetic algorithms etc. which works with more than one solution at a time and aim at searching individuals in the set of non-dominated solutions gives a great opportunity to deal with multiple objective cell formation problems. Using such methods, the selection strategy may favor non-dominated solutions while maintaining a sufficient diversity.

Considering the issues discussed above, in this thesis, a hybrid simulation-analytic fuzzy goal programming model (FGP) is developed in order to achieve the development of the new CM design methodology that addresses the uncertainty issues and routing flexibility,. In this model, the goals which are difficult to represent analytically are obtained by simulation model whereas other goals are calculated analytically. The stochastic nature of the manufacturing system is also reflected by simulation model. Part demand rates, part processing and transfer times are all stochastic. Proposed hybrid simulation-analytic FGP models are solved by using a tabu search algorithm.

CHAPTER THREE

APPROACH: SIMULATION BASED FGP MODEL FOR CELL FORMATION

The aim of this chapter is to develop a new cell formation methodology which addresses the issues of uncertain production requirements and routing flexibility. For this purpose, in this section, a hybrid analytic-simulation fuzzy goal programming model will be proposed. In this model, the goals which are difficult to represent analytically are obtained by simulation model whereas other goals are calculated analytically. The stochastic production requirements such as stochastic demand, processing times, transfer times etc. are also represented by simulation model. Since the target levels of goals are fuzzy, the proposed model also allows for the vague aspiration levels of decision makers. The issue of routing flexibility is also considered in the proposed model.

This chapter is divided into three sections. Section 3.1 gives a brief explanation about fuzzy mathematical programming, fuzzy linear programming and fuzzy goal programming. Section 3.2 presents the proposed Fuzzy Goal programming model for cell formation. The summary of Chapter 3 is given in Section 3.3

3.1 Fuzzy Mathematical Programming

Estimating the exact values of the coefficients, the right hand side values of constraints, the target values of goals are difficult tasks in modeling multi objective decision making problems. The uncertainty still exists in the problem even if all information is either given by a decision maker subjectively or by statistical inference from historical data. Therefore, reflecting this uncertainty one needs to construct a model with inexact coefficient, constraints and goals. Some researchers considered this problem as a Fuzzy Linear Programming with fuzzy coefficients of which a membership function was defined for each fuzzy coefficient. Thus, a fuzzy Solution can be obtained (Wang & Wang, 1997).

Inuiguchi & Ramik (2000) stated that two major different types of uncertainty, ambiguity and vagueness exist in the real life. Ambiguity is related with the situations in which the choice between two or more alternatives is left unspecified (e.g., "processing time of a job takes *about 8 min*" phrase shows that one value around 8 is true but not known exactly). On the other hand, vagueness is associated with the difficulty of making sharp or precise distinctions in the world; that is, some domain of interest is vague if it cannot be delimited by sharp boundaries (e.g., "decision maker wants to make profit substantially larger than $ 5100" phrase shows that values around 5100 and larger than 5100 are to some extent and completely satisfactory, respectively).

Inuiguchi & Ramik (2000) classified the fuzzy mathematical programming methods into three categories considering the types of uncertainty incorporated in the method:

- Fuzzy mathematical programming with vagueness: it treats decision making problem under fuzzy goals and constraints,
- Fuzzy mathematical programming with ambiguity: it treats ambiguous coefficients of objective functions and constraints but does not treat fuzzy goal and constraints,
- Fuzzy mathematical programming with vagueness and ambiguity: it treats ambiguous coefficients as well as vague decision maker's preference.

There are lots of fuzzy mathematical programming types. As discussed in the first chapter, in this dissertation, we use Fuzzy Goal Programming in order to form manufacturing cells. Thus, we will restrict ourselves to describe only the two types of fuzzy mathematical programming. These are Fuzzy Linear Programming (FLP) and Fuzzy Goal Programming (FGP). In the next section a brief overview of them is given.

3.1.1 Fuzzy Linear Programming

Consider a linear programming (LP) model,

$$\begin{aligned} &\min imize && z = cx \\ &subject\ to && Ax \le b \\ & && x \ge 0 \end{aligned} \tag{3.1}$$

where c= (c_1, c_2, ..., c_n) is the *n* dimensional row vector of coefficients of objective function, *x* is an *n*-dimensional column vector of the decision variables, *A* is an *m x n* matrix of constants, and *b* is an *m*-dimensional column vector of right-hand side constants. Considering the imprecision of the decision maker's judgment, Zimmerman adopted the fuzzy version of the model (3.1) as shown below:

$$\left.\begin{array}{l} cx \prec z_0 \\ Ax \prec b \\ x \ge 0 \end{array}\right\} \tag{3.2}$$

Where the symbols "$\prec$ and $\succ$" denote the fuzzified versions of "$\le$ and $\ge$" and can be read as "essentially less (greater) than or equal to", respectively.

In order to solve (3.2), Zimmermann (1978) suggested a linear membership function for the goal $\mu_1(cx)$, where

$$\mu_1(cx) = \begin{cases} 1 & if \quad cx \le z_0, \\ 1-(cx - z_0)/d_1 & if \quad z_0 \le cx \le z_0 + d_1, \\ 0 & if \quad cx \ge z_0 + d_1 \end{cases} \tag{3.3}$$

And another linear membership function $\mu_{2i}(a_i X)$ is suggested for the i^{th} constraint in the system constraints, where

$$\mu_{2i}((Ax)_i) = \begin{cases} 1 & if \quad a_i x \le b_i \\ 1-(a_i x - b_i)/d_{2i} & if \quad b_i \le a_i x \le b_i + d_{2i} \\ 0 & if \quad a_i x \ge b_i + d_{2i}, \end{cases} \tag{3.4}$$

Where d_1 and d_{2i} ($i=1, 2,..., m$) are chosen constants of admissible violations, and a_i is the i^{th} row of the matrix A (Mohamed, 1997). $\mu_1(cx)$ and $\mu_{2i}((Ax)_i)$ represent the degree of the membership of goals and constraints. It is assumed that the value of the i^{th} membership function should be 1 if the i^{th} constraint is very well satisfied, 0 if the i^{th} constraint is strongly violated its limit d_{2i}, and linear from 0 to1 (Sakawa, 1992). Figure 3.1 illustrates the "essentially less than or equal to" type linear membership function.

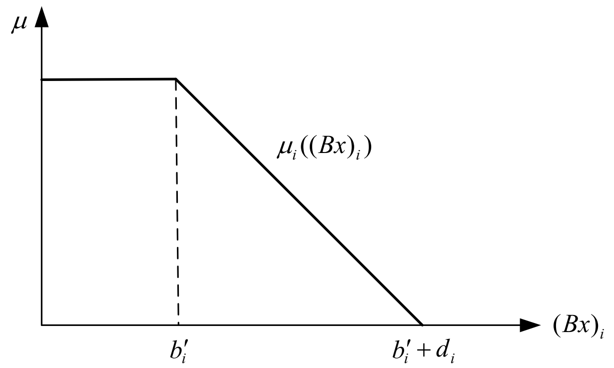


Figure 3.1 "essentially less than or equal to" type linear membership function

The degree of the membership of goals and constraints represents the satisfaction level of the decision maker. Hence the value of the membership functions should be maximized (Wang & Wang, 1997). Following the fuzzy decision theorem of Bellman and Zadeh (1970), the maximizing decision is then defined as:

$$\max_x \min(\mu_1(cx),..., \mu_{21}(a_1 x),..., \mu_{2m}(a_m x)) \tag{3.5}$$

Introducing a new variable λ, this problem can be equivalently transformed as:

$$
\begin{array}{ll}
\max & \lambda \\
subject\,to & \mu_1(cx) \geq \lambda \\
 & \mu_{2i}(a_i x) \geq \lambda \quad i = 1,2,\ldots,m \\
 & x \geq 0
\end{array}
\tag{3.6}
$$

According to above membership functions, FLP for (3.2) can be rewritten as (Mohamed, 1997):

$$
\begin{array}{ll}
\max & \lambda \\
subject\,to & \lambda \leq 1-(cx - z_0)/d_1 \\
 & \lambda \leq 1-(a_i x - b_i)/d_{2i} \quad i = 1,2,\ldots m \\
 & \lambda \geq 0, x \geq 0.
\end{array}
\tag{3.7}
$$

It is obvious that FLP model can be easily extended to fuzzy multi-objective linear programming (FMOLP) by defining a membership function for each objective functions. Assume that there are k linear objective functions to be minimized; the corresponding FMOLP model can be defined as

$$
\begin{array}{lll}
\max & \lambda & \\
subject\,to & \lambda \leq 1-(c_k x - z_{0k})/d_{1k} & k = 1,2,\ldots,k \\
 & \lambda \leq 1-(a_i x - b_i)/d_{2i} & i = 1,2,\ldots,m \\
 & \lambda \geq 0, x \geq 0. &
\end{array}
\tag{3.8}
$$

The construction of the linear membership functions is a difficult task. Zimmerman (1978) suggested the use of pay-off table in order to overcome this difficulty. The use of pay-off table operates in the following way: The first objective is set as objective and MOLP model is solved. Then the second, third and other objective functions are set as objective and MOLP models is solved for each of the objectives one by one. For each step, the value of the objectives and other objective function values are recorded. Then the payoff table is constructed as shown in Table 3.1.

Table 3.1 The payoff table

	The objective function			
Value	Z_1	Z_2	$\cdots$	Z_M
Z_1	Z_{11}	Z_{12}	$\cdots$	Z_{1M}
Z_2	Z_{21}	Z_{22}	$\cdots$	Z_{2M}
$\vdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
Z_M	Z_{M1}	Z_{M2}	$\cdots$	Z_{MM}

Examining the figures in Table 3.1, the best lower bound (l_k) and the worst upper bound (u_k) are determined. Then the membership functions of each objective can be defined as follows:

$$\mu_{z_k}(x) = \begin{cases} 1 & Z_k(x) \le l_k \\ \dfrac{u_k - Z_k(x)}{u_k - l_k} & l_k < Z_k(x) \le u_k \\ 0 & Z_k(x) > u_k. \end{cases} \quad (3.9)$$

Although the fuzzy description is hypothetical and membership values are subjective, various types of membership functions can be used to support the fuzzy analytical framework (Wang & Wang, 1997).

3.1.2 Fuzzy Goal Programming

Goal programming (GP) is one of the most powerful, multi-objective decision making approaches in practical decision making. GP requires the decision maker (DM) to set precise aspiration values of each objective that he/she wishes to achieve. GP solution technique aims at minimizing the deviations from each goal, subject to the goal constraints and system constraints. In a standard GP formulation, goals and constraints are defined precisely. However, the major difficulty for a DM in using

GP is to determine precisely the goal value of each objective function. (Arikan and Gungor, 2001).

Zadeh (1965) introduced FST that is a generalization of conventional set theory as a mathematical way to represent vagueness in everyday life. A fuzzy set A can be characterized by a membership function (MF), usually denoted by μ, which assigns to each object of a domain its grade of membership in A. The nearer the value of MF to unity, the higher the grade of membership of element or object in a fuzzy set A. Various types of membership functions can be used to represent the fuzzy set.

Applying fuzzy set theory into GP has the advantage of allowing for the vague aspirations of a DM, which can then be qualified by some natural language terms. In other words, FGP is specified in an imprecise manner. A fuzzy goal is considered as a goal with an imprecise aspiration level.

GP aims to minimize the distance between Z_k and an aspiration level (target value of the objective function) $\overline{Z}_k$, which is expressed by the deviational variables. In FGP, membership function values of the each objective replace by the deviational variables (Mohamed, 1997).

Narasimhan (1980) fist introduced fuzzy goal programming approach to specify imprecise aspiration levels of fuzzy goals. Yong, Inzigo and Kim (1991) formulated FGP with non-linear membership functions. FGP technique has been applied to various fields such as structural optimization (Rao, Sundaraju, Prakash and Balakrishna, 1992), agricultural planning (Sinha, Rao and Mangoraj, 1988), forestry (Pickens and Hof, 1991), cellular manufacturing (Shanker and Vrat, 1999), aggregate production planning (Baykasoglu, 2006).

A typical FGP problem formulation can be stated as follows:

Find x_i $\quad i = 1,2,...,n$

to satisfy

$$
\begin{aligned}
&Z_m(x_i) \prec \overline{Z}_m && m = 1,2,...,M, \\
&Z_k(x_i) \succ \overline{Z}_k && k = M+1, M+2,...,K, \\
&g_j(x_i) \leq b_j && j = 1,2,...,J, \\
&x_i \geq 0 && i = 1,2,...,n.
\end{aligned} \tag{3.10}
$$

Where

$Z_m(x_i)$ = the *m*th goal constraint,

$Z_k(x_i)$ = the *k*th goal constraint,

$\overline{Z}_m(x_i)$ = the target value of the *m*th goal,

$\overline{Z}_k(x_i)$ = the target value of the *k*th goal,

$g_j(x_i)$ = the *j*th inequality constraint,

b_j = the available resource of inequality constraint j.

In formulation (3.10), the symbols "$\prec$" and "$\succ$" denote the fuzzified versions of "$\leq$" and "$\geq$" and can be read as "approximately less (greater) than or equal to". For "approximately less than or equal to" situation, the goal *m* is allowed to be spread to the right-hand-side of $\overline{Z}_m$ ($\overline{Z}_m = l_m$ where l_m denote the lower bound for the m^{th} objective) with a certain range of r_m ($\overline{Z}_m + r_m = u_m$, where u_m denote the upper bound for the m^{th} objective). Similarly, with "approximately greater than or equal to", p_k is the allowed left side of $\overline{Z}_k$ ($\overline{Z}_k$ - $p_k = l_k$, and $\overline{Z}_k = u_k$) (Wang and Fu, 1997).

As can be seen from the explanations given so far, GP and FGP have some similarities. Both GP and FGP need an aspiration level for each objective. These aspiration levels are determined by the decision maker. Furthermore, FGP needs max-min limits (u_k, l_k) for each goal (Mohamed, 1997). After determining the fuzzified aspiration levels with respect to the linguistic terms of "approximately less than or equal to", and "approximately greater than or equal to", the fuzzy membership functions can be written for each goals as follows:

For "approximately less than or equal to";

$$\mu_{z_m}(x) = \begin{cases} 1 & if \quad Z_m(x) \le l_m, \\ 1 - \dfrac{Z_m(x) - l_m}{u_m - l_m} & if \quad l_m \le Z_m(x) \le u_m, \\ 0 & if \quad Z_m(x) \ge u_m. \end{cases} \tag{3.11}$$

For “approximately greater than or equal to”;

$$\mu_{z_k}(x) = \begin{cases} 1 & if \quad Z_k(x) \ge u_k, \\ 1 - \dfrac{u_k - Z_k(x)}{u_k - l_k} & if \quad l_k \le Z_k(x) \le u_k, \\ 0 & if \quad Z_k(x) \le l_k. \end{cases} \tag{3.12}$$

Figure 3.2 illustrates both types of membership functions.

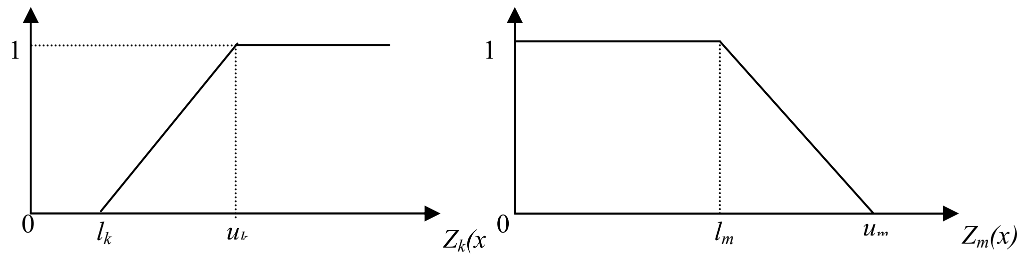


Figure 3.2 Membership functions of fuzzy goals

According to the approach suggested by Belman and Zadeh (1970), the feasible fuzzy solution set is obtained by the intersection of the all the membership functions representing the fuzzy goals. This feasible fuzzy solution set is then characterized by its membership $\mu_F(x)$ as following:

$$\mu_F(x) = \mu_{Z_1}(x) \cap \mu_{Z_2}(x) .... \cap \mu_{Z_k}(x) = \min[\mu_{Z_1}(x), \mu_{Z_2}(x), ..., \mu_{Z_k}(x)] \tag{3.13}$$

Then the optimum decision can be determined to be the maximum degree of membership for the fuzzy decision:

$$\max_{x \in F} \mu_F(x) = \max_{x \in F} \min[\mu_{Z_1}(x), \mu_{Z_2}(x), ..., \mu_{Z_k}(x)] \tag{3.14}$$

Zimmermann (1978) first used the maximin-operator of Bellman and Zadeh (1970). By introducing the auxiliary variable λ, which is the overall satisfactory level of compromise, By introducing the auxiliary variable λ, which is the overall satisfactory level of compromise, formulation (3.10) can be equivalently transformed as:

$$\left.\begin{array}{lll} \text{maximize} & \lambda & \\ \text{subject to} & & \\ & \lambda \le \mu_{Z_k} & k = 1,\ldots,K \\ & g_j(x_i) \le b_j & i = 1,\ldots, n, \quad j = 1,\ldots, J \\ & x_i \ge 0 & i = 1,\ldots, n \\ & \lambda \in [0,1]. & \end{array}\right\} \tag{3.15}$$

The model mentioned above commonly use the min- operator for aggregating goals to determine the decision set, and then to maximize the set. Tiwari, Dharmar and Rao (1986) presented a simple additive model to formulate an FGP problem. The simple additive model is formulated as:

$$\left.\begin{array}{lll} \text{Maximize} & \sum_{k=1}^{K} \mu_{Z_k} & \\ \text{subject to} & & \\ & g_j(x_i) \le b_j & i = 1,\ldots, n, \quad j = 1,\ldots, J \\ & \mu_{Z_k} \in [0,1] & k = 1,\ldots,K \\ & x_i \ge 0 & i = 1,\ldots, n. \end{array}\right\} \tag{3.16}$$

With the use of an additive model, the maximum sum of goals' achievement degrees can be obtained. The achievement degree of some goals will not be decrease because of a particular goal that is difficult to achieve. This advantage makes the additive model appealing(Chen & Tsai, 2001).

Since some goals may be more important than the others, consideration of different importance and priority levels of the goals is also important in FGP. For this purpose, the weighted average of deviations from aspiration levels is generally used in conventional GP. Hannan (1981) used this approach to formulate objective function of an FGP with fuzzy priority. Tiwari et al. (1987) proposed a weighted model that incorporates each goal's weight into the objective function in an additive fashion. Using Tiwari et al. (1987)'s approach, the model (3.10) can written as follows:

$$
\left.
\begin{array}{lll}
\text{Maximize} & \sum_{k=1}^{K} w_k \mu_{Z_k} & \\
\text{subject to} & & \\
& g_j(x_i) \le b_j & i = 1,\dots, n, \quad j = 1,\dots, J \\
& \mu_{Z_k} \in [0,1] & k = 1,\dots,K \\
& x_i \ge 0 & i = 1,\dots, n.
\end{array}
\right\} \qquad (3.17)
$$

Where w_k denotes the weight of the *k*th fuzzy goal, and $\sum w_k = 1$. Weights in the model show the relative importance of the fuzzy goals.

Chen and Tsai (2001) formulated the same problem with a preemptive structure using additive model. They incorporate the preemptive priority structure into this formulation to find a set of solutions that maximize the sum of each fuzzy goal's achievement degree.

To illustrate the formulation, an example with five goals is given below:

- Priority level 1: Goal 1 and 3;
- Priority level 2: Goal 2;
- Priority level 3: Goal 4 and 5;

According to the above preemptive priority structure, the following relationship exits for the respective achievement degrees for the goals:

$$\mu_1 \geq \mu_2$$
$$\mu_3 \geq \mu_2$$
$$\mu_2 \geq \mu_4 \quad \text{and}$$
$$\mu_2 \geq \mu_5$$

After addicting the above relationship to the model (3.10), the FGP can be formulated as;

$$\left.\begin{array}{lll} \text{Maximize} & \sum_{k=1}^{K} \mu_{Z_k} & \\ & & \\ \text{subject to} & & \\ & g_j(x_i) \leq b_j & i = 1,..., n, \quad j = 1,..., J \\ & \mu_{Z_1} \geq \mu_{Z_2} & \\ & \mu_{Z_3} \geq \mu_{Z_2} & \\ & \mu_{Z_2} \geq \mu_{Z_4} & \\ & \mu_{Z_2} \geq \mu_{Z_5} & \\ & \mu_{Z_k} \in [0,1] & k = 1,..., K \\ & x_i \geq 0 & i = 1,..., n. \end{array}\right\} \quad (3.18)$$

As described in Section 2.2.2, although the fuzzy clustering techniques have been widely used in cell formation, the application of FMP is a relatively new research area in cell formation. After brief definitions about FLP and FGP, a fuzzy goal programming model for cell formation will be presented in the following section.

3.2. Proposed Hybrid FGP Model for Cell Formation

As mentioned in Chapter 2, most of the current CF studies depend on deterministic models which are based on the assumption of certainty of cost factors, deterministic processing times, deterministic demand etc. Such relaxations affects the design and implementation of cellular manufacturing systems and may lead to cell designs that are far from meeting the requirements of real world applications. Real manufacturing systems have stochastic nature and tend to have uncertainty and vagueness in system parameters. Fuzzy set theory gives the opportunity to deal with uncertainty and vagueness. Simulation is another tool for representing the stochastic nature of the manufacturing systems. By simulation models, system parameters such as demand rate, processing times etc. can be modeled in stochastic manner.

Routing flexibility is another important issue in cell formation. In the presence of routing flexibility, each part can have different process plans. Exiting studies in CF literature shows that the ignorance of the existence of alternative process plans results in increased operation cost and additional investment in machine.

The aim of this study is to develop a cell design methodology that capture two important characteristics of manufacturing cells: (1) the existence of stochastic production requirements (2) the existence of routing flexibility. For this purpose, in this thesis, a hybrid analytic-simulation fuzzy goal programming (FGP) model is proposed in order to support the cell formation process. In the proposed hybrid analytic-simulation model, the stochastic nature of the production system is reflected by a simulation model. The part processing times, intercellular part movement times and the part arrivals are all stochastic. The objectives of maximizing system utilization, minimizing mean tardiness and minimizing the percentage of tardy jobs which are difficult to represent analytically are also obtained by simulation model. The other objective of minimum number of exceptional elements is obtained by an analytical equation. The proposed model also incorporates alternative process plans for parts. The fuzzy goals will be handled by using both simple additive method and max-min method. The mathematical model is developed under the following assumptions:

Assumptions:

- The product mix is known.
- Each part have equal number of operations (This assumption will be relaxed later)

- Parts moved between cells in batches.
- The upper and lower bounds on a quantity of machines in a cell need to be specified.
- Each machine type can perform one or more operations. Likewise, each operation can be performed on one or more machine types with equal (or different) times.
- Set-up times are not considered.
- Machine breakdowns are not considered.
- Batch size is constant for all productions.

3.2.1 Notation and Mathematical Formulation

The mathematical formulation for the design of CMS is developed such that part families and machine cells are formed simultaneously. According to existing CF literature, the use of simultaneous part-machine grouping strategy generally yields better results than sequential strategies. The mathematical formulation for the design of CMS is presented next. Note that, the stochastic production requirements such as stochastic demand, stochastic processing times etc. are not included in mathematical formulation. Such characteristics of manufacturing system will be incorporated in simulation model. The detailed explanations about simulation model and integration of the mathematical model with simulation will be given in Chapter 4.

Indices

$i = 1,2,\ldots,I$	Jobs
$o = 1,2\ldots O$	Operations
$c=1,2,\ldots C$	Cells
$m=1,2,\ldots M$	Machines

System and Input Parameters:

P_{iom} : Set of machines that can perform *o* th operation of part *i*.

$$P_{iom} = \begin{cases} 1 & \textit{if oth operation of job i can be performed on machine m} \\ 0 & \textit{Otherwise} \end{cases}$$

K: A big number

M_{min}: Min number of machines in order to form a cell

M_{max}: Max number of machines that can be included in a cell.

$goal_{int}$: Aspiration level for goal_1

$goal_{util}$: Aspiration level for goal_2

$goal_{tardiness}$: Aspiration level for goal_3

$goal_{tardyjobs}$: Aspiration level for goal_4

Product demand: Product demand is the quantity of each part type in the product mix to be produced. The product demand of each part is expected to be varied across the planning horizon. Since the proposed model is a hybrid analytic-simulation model, it is possible to define demand distribution of a part instead of deterministic demand. The distribution of part demand will be defined in simulation model.

Operating time: Operating time is the time required by a machine to perform an operation on a part type. Similar to part demand, it is possible to define the distribution of part processing times instead of deterministic part processing times. Hence operating times are not included in the mathematical models. They are reflected by the simulation model.

Transfer Time: Similar to processing times, the distribution of inter-cell part transfer times will be considered in simulation model.

Decision variables:

$$Q(c)=\begin{cases}1 & if\ cell\ is\ formed\\ 0 & Otherwise\end{cases} \tag{3.19}$$

$$Z_{ioc}=\begin{cases}1 & if\ oth\ \ op.\ of\ job\ i\ is\ performed\ in\ cell\ c\\ 0 & Otherwise\end{cases} \tag{3.20}$$

$$D1_{ioc}=\begin{cases}1 & if\ oth\ oper.\ of\ job\ i\ is\ performed\ in\ another\ cell\\ 0 & Otherwise\end{cases} \tag{3.21}$$

$$X_{iocm}=\begin{cases}1 & if\ oth\ op.\ of\ job\ i\ is\ assigned\ to\ machine\ m\ in\ cell\ c\\ 0 & Otherwise\end{cases} \tag{3.22}$$

$$Y_{cm}=\begin{cases}1 & if\ mahine\ m\ is\ assigned\ to\ cell\ c\\ 0 & Otherwise\end{cases} \tag{3.23}$$

Decision variable Q_c shows whether the cell c is formed or not. Z_{ioc} and D_{1ioc} used for calculating the number of exceptional elements. X_{iocm} represents the assignment of o^{th} operation of part *i* to a machine *m* which is allocated to cell *c*. Variable Y_{cm} represents the assignment of machine type *m* to cell *c*.

Fuzzy goals and other system constraints:

$$Goal\_1: \sum_i \sum_o \sum_c D_{1ioc} \prec goal_{excpt} \tag{3.24}$$

$$Goal\_2: system\ \ utilization\ \ \succ goal_{util} \tag{3.25}$$

$$Goal\_3: mean\ tardiness \prec goal_{tardiness} \tag{3.26}$$

$$Goal\_4: percentage\ of\ tardy\ jobs \prec goal_{tardyjobs} \tag{3.27}$$

In this study, combination of multiple performance oriented goals which is not considered in current CF literature will be considered. These objectives are as follows:

- Exceptional elements: In CMS, it is desired to complete al operations of a part in the same cell. However, in real applications, parts can visit different cells when it requires processing on a machine that is not available in the allocated cell of a part. Such parts are called as “exceptional elements” and require inter-cell movements. Inter-cell movements result in extra transportation costs and requires more coordinating efforts between cells as well. Thus, exceptional elements and inter-cell movements are undesirable and should be minimized in CMS.

- Utilization: Utilization is another important performance measure in determining manufacturing cell formation. Since the set-up times are decreased, the effective capacity of the machines is increased thus leading to lower utilization. Demand fluctuations can also lead to lower utilizations. The general level of utilization of cells is of the order of 60-70%. Hence the maximization of system utilization is an important objective for cellular manufacturing systems.

- Mean tardiness: mean tardiness objective is important when customers tolerate smaller tardiness but become rapidly and progressively more upset for larger ones.

- Percentage of tardy jobs: percentage of tardy jobs is important when customers simply refuse to accept tardy jobs, so that the order is lost.

Last two objectives are especially important for the manufacturing system that works with Just in Time manufacturing philosophy. However, these objectives have not been considered in most of the existing studies probably due to the complex nature of cell formation problems. As stated above, in this study, a hybrid analytic-simulation FGP model which allows us to represent the stochastic nature of manufacturing system and to consider objectives such as utilization, mean tardiness, number of tardy jobs etc. is proposed. In this proposed hybrid model, the objective of minimizing the number of exceptional elements is obtained by an analytic equation (Eq.3.24) whereas other three objectives which are difficult to obtain analytically are obtained by simulation model. The structure of the simulation model and how we integrated with mathematical model will be explained in the next chapter.

In formulation (3.24-3.27), the symbols "$\prec$" and "$\succ$" denote the fuzzified versions of "$\leq$" and "$\geq$" and can be read as "approximately less (greater) than or equal to". The objectives of the mathematical model are minimizing the number of exceptional elements (3.24), maximizing system utilization(3.25), minimizing mean tardiness(3.26) and minimizing the percentage of tardy jobs (3.27). The number of exceptional elements should be substantially smaller than $goal_{excpt}$, system utilization should be substantially greater than $goal_{util}$, the mean tardiness should be substantially smaller than $goal_{tardiness}$ and the percentage of tardy jobs should be substantially smaller than $goal_{tardyjobs}$. Other system constraints are as follows:

$$\sum_{c}\sum_{m} X_{iocm} P_{iom} = 1 \qquad (3.28)$$

$$\sum_{c} Y_{cm} = 1 \qquad \forall m \qquad (3.29)$$

$$X_{iocm} \leq K.Y_{cm} \qquad \forall i,o,c,m \qquad (3.30)$$

$$X_{iocm} \leq K.Z_{ioc} \qquad \forall i,o,c,m \qquad (3.31)$$

$$\sum_{c} Z_{ioc} = 1 \qquad \forall i,o \qquad (3.32)$$

$$Z_{ioc} - Z_{ioc-1} = D_{1ioc} - D_{ioc} \qquad \forall i,o,c \qquad (3.33)$$

$$\sum_{m} Y_{cm} \leq M_{\max} Q_c \qquad \forall c \tag{3.34}$$

$$\sum_{m} Y_{cm} \geq M_{\min} Q_c \qquad \forall c \tag{3.35}$$

$$X_{iocm}, Y_{cm}, Z_{ioc}, Q_c, D_{1ioc}, D_{ioc} = [0,1] \tag{3.36}$$

Equation (3.28) ensures that the operation of a job is assigned to a machine in a cell. Equation (3.29) ensures that each machine is assigned to only one manufacturing cell. Equation (3.30) indicates that if an operation in a cell c is assigned to a machine *m*, this machine is assigned to cell *c*. Equation(3.31) indicates that if the oth operation of part *i* is assigned to machine *m* in a cell c, operation *o* is assigned to cell *c*. Equation (3.32) ensures that an operation is assigned to only one cell. Equation (3.33) controls whether the consecutive two operations of a job is performed in the same cell. Instead of using predefined number of cells, in the proposed model, lower and upper bounds on the number of machines that can be included by a manufacturing cell is used. Equation (3.34-3.35) constraints the number of machines assigned to each cell if it is formed.

The fuzzy membership functions can be written for each goals as follows :

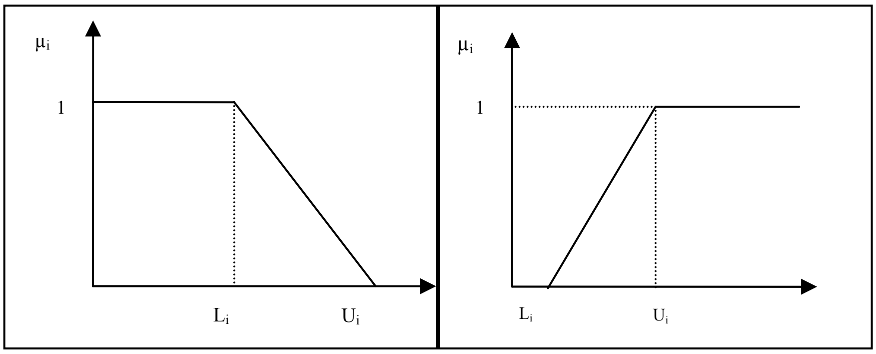
$$\mu_1 = \begin{cases} 1 & \text{if } f_1 \leq L_1 \\ \dfrac{U_1 - f_1}{U_1 - L_1} & \text{if } L_1 \leq f_1 \leq U_1 \\ 0 & \text{if } f_1 \geq U_1 \end{cases} \tag{3.37}$$

$$\mu_2 = \begin{cases} 1 & \text{if } f_2 \geq U_2 \\ \dfrac{f_2 - L_2}{U_2 - L_2} \ \text{if} & L_2 \leq f_2 \leq U_2 \\ 0 & \text{if } f_2 \leq L_2 \end{cases} \tag{3.38}$$

$$\mu_3 = \begin{cases} 1 & if\ f_3 \le L_3 \\ \dfrac{U_3 - f_3}{U_3 - L_3}\ if & L_3 \le f_3 \le U_3 \\ 0 & if\ f_3 \ge U_3 \end{cases} \tag{3.39}$$

$$\mu_4 = \begin{cases} 1 & if\ f_4 \le L_4 \\ \dfrac{U_4 - f_4}{U_4 - L_4}\ if & L_4 \le f_4 \le U_4 \\ 0 & if\ f_4 \ge U_4 \end{cases} \tag{3.40}$$

The shapes of the membership functions are given in Figure 3.3. For instance, in Figure3.3, the first goal is allowed to be spread to the right-hand side of L_1 with a certain range of r_1 ($r_1 = U_1 - L_1$).



(a) (b)

Figure 3.3. (a) The shape of membership function for objectives 1,3 and 4(minimization) (b) The shape of membership function for objective 2 (maximization)

After defining membership functions, using the max-min operator (3.15) λ, which is the overall satisfactory level of compromise, the standard goal programming formulation can be equivalently transformed as:

$$\left.\begin{aligned} & Max Z = \lambda \\ & \text{subject to} \\ & \qquad \mu_1, \mu_2, \mu_3, \mu_4 \geq \lambda \\ & \qquad \text{and other system constraints}(3.28-3.36) \end{aligned}\right\} \quad (3.41)$$

Using the additive method (3.16), standard goal programming formulation can be equivalently transformed as:

$$\left.\begin{aligned} & Max Z = \sum_{i=1}^{4} \mu_i \\ & \text{subject to} \\ & \qquad \mu_1, \mu_2, \mu_3, \mu_4 \geq 0 \\ & \qquad \text{and other system constraints}(3.28\text{-}3.36) \end{aligned}\right\} \quad (3.42)$$

The proposed models differ from the existing cell formation approaches in three ways:

First, application of FGP to cell formation problem allows for the vague aspirations of a DM. As mentioned in the previous chapter, GP approaches have been widely used in cell formation. However, defining precise aspiration levels is a difficult task especially for real applications. FGP has the advantage of allowing for the vague aspirations of a DM. Unlike GP, there is a limited number of works in which FGP approaches used for cell formation (See section 2.2.2).

Second, the proposed hybrid FGP models consider the stochastic production requirements such as stochastic demand, process times and part transfer times. The stochastic nature of the manufacturing system will be reflected by a simulation model.

Third, the combination of multiple performance oriented goals which have not been considered by the current CF literature is included in the proposed models. Instead of analytic representation, the values of some goals are obtained by simulation model. This also leads to decrease in complexity of the model.

Since the proposed model is a hybrid analytic-simulation model in which some objectives are obtained from simulation, classical solution approaches such as simplex based methods are not appropriate for solution. In the next chapter, a tabu search based solution methodology for solving hybrid FGP will be presented.

3.3 Chapter Summary

As stated in Chapter 2, the existence of stochastic production requirements and the existence of routing flexibility are two important characteristics of CMS design problem which are generally omitted by researchers. The most of the current CF approaches based on deterministic models. Relaxation in modeling assumptions such as deterministic product demand, deterministic processing times etc. affects the design and implementation process of manufacturing cells. Real manufacturing systems tend to have uncertainty and vagueness in system parameters. Hence, the issue of uncertainty should be considered in the design process of manufacturing cells in order to have more realistic cell designs. Fuzzy set theory gives the opportunity to deal with uncertainty and vagueness. Simulation is another tool for representing the stochastic nature of the manufacturing systems.

Routing flexibility is another important characteristic for manufacturing cells. In the presence of routing flexibility, parts can have more than one process plan. The existence of alternative process plans for the parts can improve the groupability of parts. The ignorance of routing flexibility may result in an increased operation cost and additional investment in machines. So, it is important to consider such an important factor by quantifying alternative routes and flexible machining processes in the process of cell formation.

The aim of the research proposed in this dissertation is to develop a cell design methodology considering the existence of uncertain production requirements and routing flexibility. For this purpose, a hybrid simulation-analytic FGP models using additive method and max-min method are presented in this chapter. In proposed hybrid analytic-simulation model, the stochastic nature of the production system is represented by a simulation model. The part processing times, intercellular part movement times and the part arrivals are all stochastic. In proposed models the combination of multiple performance oriented goals which have not been considered by the current CF literature are also included. Since the classical solution approaches are not appropriate for solving the proposed hybrid FGP model, a tabu search based solution methodology for solving FGP models will be presented in the following chapter.

CHAPTER FOUR

SOLUTION METHODOLOGY: A TABU SEARCH BASED SOLUTION METHODOLOGY FOR FGP MODELS

Since the FGP model developed in the previous chapter have some objectives determined by a simulation model, classical solution techniques such as simplex based methods are not appropriate for solution. As stated in Section 2.3, TS is an adaptive search procedure that has been employed for solving combinatorial optimization problems. TS can handle any type for variables (integer, binary, discrete etc) and constraints (linear or nonlinear). TS algorithm is also used for solving multiple objective optimization problems. TS works with more than one solution at a time (neighborhood solutions). Among these solutions, a DM can choose any solution that satisfies his/her requirements best. This feature of TS is important in dealing with multiple objectives or goals. Hence a tabu search based solution methodology is used for solution. In this chapter, the TS procedure used for solving FGP models will be presented.

This chapter is divided into five sections. Section 4.1 gives general explanation about basic TS algorithm. Section 4.2 presents a TS algorithm for solving FGP models. In Section 4.3 the effectiveness of TS algorithm is tested on FGP models from literature. The results are compared with the results of optimization software LINGO. In section 4.4 the deterministic form of FGP model for cell formation proposed in Chapter 3 is solved by TS and the results are compared with LINGO results. The summary and conclusions of Chapter 4 is given in Section 4.5.

4.1 Basic Tabu Search Procedure

Classical solution approaches to mathematical programming problems have some limitations. The effectiveness of these approaches are depended on the parameters such as the size of the solution space, number and type of constraints and variables. Since most of the classical solution techniques are local optimization techniques it is not possible to find global optimum or near optimum when the problem size gets larger. Model independent modern heuristic approaches such as genetic algorithm, TS, simulated annealing, neural networks etc. have been proposed to overcome these difficulties.

TS (Glover, 1986) is a stochastic neighborhood solution heuristic that can handle any type of objective function and any type of constraints. The basic TS algorithm is as given below:

```
k:=1 /* k is the number  of iteration */
      generate initial solution s0 ∈ S, where S is a discrete set of feasible solutions
            WHILE the terminating condition is not met DO
                identify N(s). /*Neighborhood set*/
                identify T(s, k).  /*Tabu set*/
                identify A(s, k). /*Aspirant set*/
                choose the best  s* ∈ N(s, k) = {N(s) – T(s, k) ∪ A(s, k)}
                memorize s* if it improves the previous best known solution
                s:= s*
                k:= k + 1
            END WHILE
```

The steps of the TS procedure are: initial seed (solution), generation of neighborhood solutions, selection of new seed (solution), aspiration, updating tabu list and current best solution list, and terminating (Figure 4.1). These steps are described below.

- **Initial solution:** The algorithm starts with a randomly created or a known feasible solution vector s_0. Starting with a known good solution vector can decrease the computation time.
- **Generation of neighborhood solutions:** A set of neighborhood solution set N(s) is created using predefined move strategies. These move strategies are

generally depend on the type of variables. The number of solutions in neighborhood called as “neighborhood size”.

- **Selection:** The objective function is evaluated for each neighborhood solution in N(s). The best solution is selected as a new seed.
- **Updating Tabu list and current best solution list:** The current best solution list is updated when the solution is improved. An iteration of the TS is said to be completed when the entire neighborhood of a current solution is evaluated. In order to avoid recycling a list called “tabu list” is used to forbid a predefined number of recent moves. The number of the entries in the tabu list is referred o as “tabu list size”. When iteration is performed, the move implemented is added to the tabu list as the new entry and the oldest one is removed from the list.
- **Aspiration:** If a move is the best move within the neighborhood of the current solution, it is implemented even if it is in tabu list.
- **Termination:** If a predefined number of iteration is reached or if there is no improvement in the current best move list in the last “t” iterations the algorithm terminates.

In TS, the parameters *maximum number of iterations*, *tabu list size*, *neighborhood size* should be determined. They depend on problem size, type of variables. Hence there is no certain rule for determining these parameters. It is a common practice to solve the problem with different parameter sets to find best parameter combination. If the variables are spread in a wide range it is suggested to work with higher number of neighborhood solutions. Otherwise, smaller number of neighborhood solutions can be safely used (Baykasoglu, 1999). The number of iterations should be big enough to assure convergence. For tabu list size, minimum 7 and maximum 11 is suggested (Glover and Laguna, 1993).

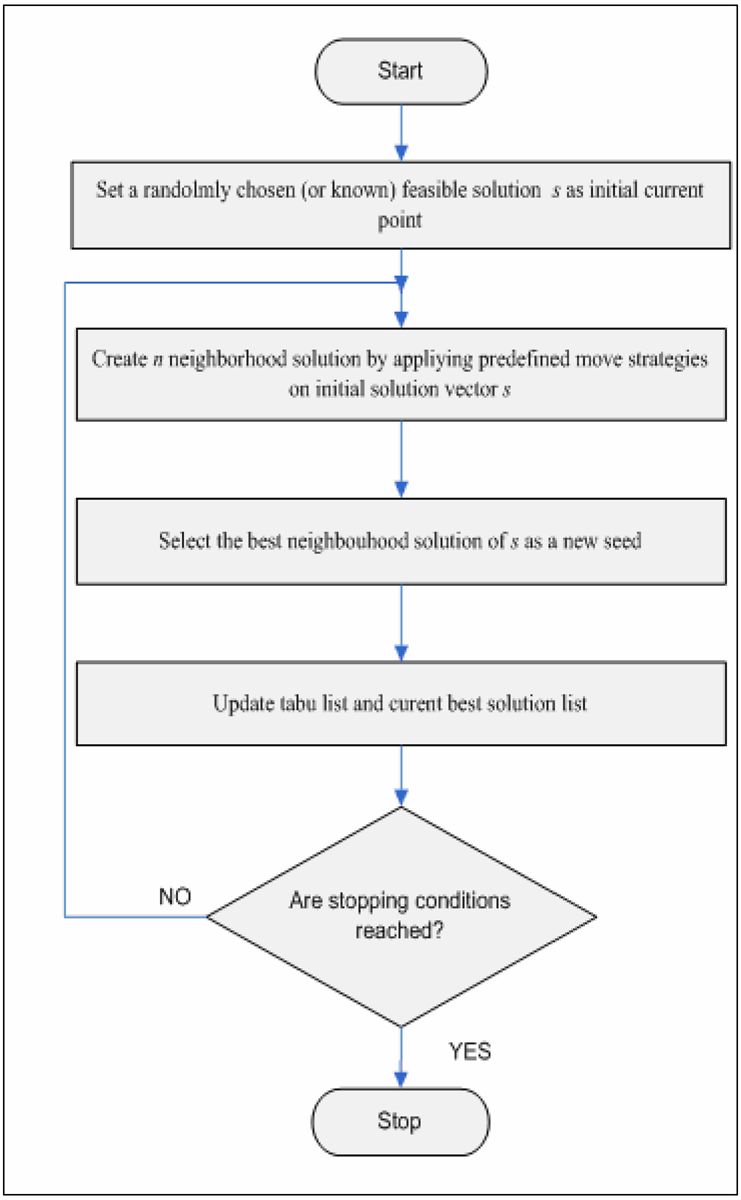


Figure 4.1 Flowchart of the basic tabu search procedure

TS is an adaptive search procedure that has been employed for solving combinatorial optimization problems from different research areas such as resource planning, telecommunications, financial analysis, scheduling, energy distribution, molecular engineering, logistics, flexible manufacturing, waste management, biomedical analysis, cell formation etc.

As stated in section 2.3, the inherent solution process of TS that involves working with more than one solution at a time (neighborhood solutions) gives great opportunity to deal with multiple objectives or goals easily (Baykasoglu, 2001). In the next section, a tabu search algorithm for solving the proposed FGP model will be presented.

4.2 Solution of Fuzzy Goal Programming Models Using Tabu Search

The solution process of TS involves working with more than one solution (neighborhood solutions) at a time. Baykasoglu (1999, 2001, and 2006) noted that this feature of TS gives a great opportunity to deal with multiple objectives or goals and proposed TS based solution methodology for multi-objective optimization problems and fuzzy goal programming models. In this dissertation, TS procedure will be used for solving FGP cell formation models. The steps of Tabu search algorithm used for solving FGP models is similar to original TS procedure except for selecting step. The steps of the TS procedure used for solving FGP models are as follows:

- **Initial solution:** Algorithm starts with a randomly generated feasible solution vector. Starting with a known good solution vector can decrease the computation time.

- **Generation of neighborhoods:** Different move strategies have been presented in TS literature. The move strategies depend on the type of variables. Since all variables are 0-1 variables in our cell formation model, the move strategy suggested by Baykasgolu (1999) is used for generation of neighborhoods.

$$x_i^* = \begin{cases} 1 & if \quad x_i = 0 \\ 0 & if \quad x_i = 1 \end{cases} \tag{4.1}$$

Where x_i = Value of the i th variable prior to the neighborhood move, x_i^* = Value of the i^{th} variable after the neighborhood move.

- **Selection:**
  - For simple additive method, the membership values of goals are calculated and summed. The neighbor with the highest sum ($\Sigma\mu$) is selected as the current best solution.
  - For max-min method, the membership functions and λ values are calculated for each neighborhood solution. The solution with the highest λ value is selected as a new seed. If there is more than one solution with the same λ value, check the $\Sigma\mu$ values and select one with the highest $\Sigma\mu$ as a new seed.
  - For preemptive method, the membership function values are calculated. Then the membership value that belongs to the first priority goal is checked for each neighborhood solution. The solution with the highest membership value is selected as a new seed. If there is more than one solution with the same membership value for the first priority goal, the membership values for other goals are checked in the order of priority levels.

    The preemptive method aims to maximize the satisfaction level of the highest priority goal. Therefore the results of the preemptive method are different form other two methods given above and depend on the priority levels. Hence the max-min method and additive method are used and compared in the experiments.

- **Updating Tabu list and current best solution list:** The current best solution list is updated when a better solution is obtained. A predefined number of previous moves are recorded in tabu list. Tabu list is updated at each iteration. When it is full, the first item of the list is removed and replaced with a new one.

- **Termination:** If a predefined number of iteration is reached or if there is no improvement in the current best move list in the last t iterations the algorithm terminates.

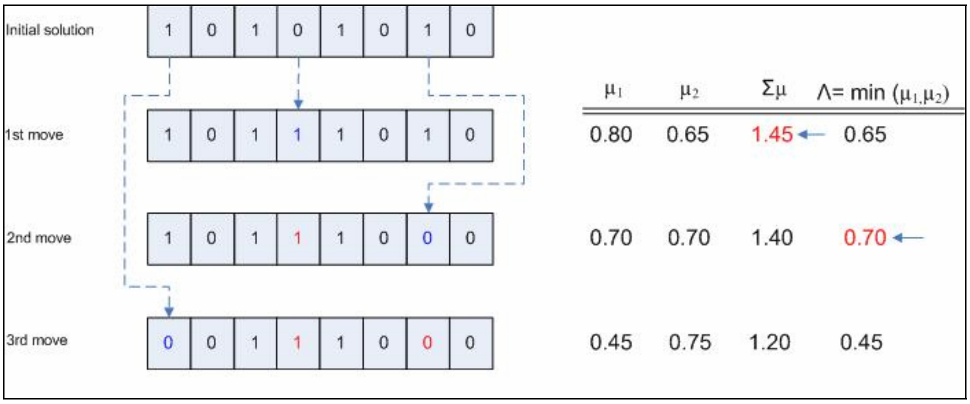


Figure 4.2 Illustrative example for TS procedure used for solving FGP models

An example is given in Figure 4.2. Assume that there are two goals, the neighborhood size and the tabu list size is 3. As seen from Figure 4.2, in the first neighborhood solution, the value of fourth variable in the initial solution is changed from 0 to 1. Hence this move becomes tabu. The second neighbor solution is generated by changing the seventh variable of the initial solution and this move is also recorded in tabu list. For generating the third solution in neighborhood, the first element of the initial solution vector is changed and this move becomes tabu. Now there are 3 different solutions in neighborhood. For selecting stage of TS process, the corresponding $\Sigma\mu$ (λ for max-min method) values are calculated for each neighborhood solution. For additive method, the first neighborhood solution is chosen as a new seed since it has the highest $\Sigma\mu$ value (1.45). For max-min method, the second neighbor solution which has the highest λ value (0.70). New neighborhood solutions are generated from the new seed and the other steps of TS procedure are repeated until termination conditions are satisfied.

4.2.1. Examples and Comparative Study

Before solving hybrid simulation-analytic FGP models that was proposed in Chapter 3, the effectiveness of TS procedure will be tested on several test problems selected from the literature to show the applicability of TS for solving FGP models. For each test problem, a C program is developed for solving FGP models using tabu search. The results are compared with the results obtained by LINGO commercial optimization software. Note that, since the cell formation model under consideration is a 0-1 FGP model, we will restrict ourselves with solving 0-1 FGP models. As mentioned in previous section, TS algorithm is able to deal with any type of variables and constraints.

4.2.1.1 Test Problem 1 (Baykasoglu et al, 1999)

Test problem 1 is taken form the study of Baykasoglu et al. (1999). The model is a standard GP model in its original form. It is adapted to FGP model and solved with tabu search algorithm given in previous section. The upper and lower bounds for membership functions are found by using payoff tables as mentioned in Section 3.1.1.

$$\max \quad \lambda$$

subject to

$$g_1(x) = 9x_1 + 9x_2 + 9x_3 + 9x_4 + 9x_5 + 9x_6 + 9x_7 \cong 80$$
$$g_2(x) = x_8 + x_9 + x_{10} + x_{11} + x_{12} + x_{13} + x_{14} \cong 5$$

$$\mu_1 = \begin{cases} 0 & ; g_1(x) < 70 \\ \dfrac{g_1(x) - 70}{80 - 70} & ; 70 \le g_1(x) \le 80 \\ 1 & ; g_1(x) = 80 \\ \dfrac{100 - g_1(x)}{100 - 80} & ; 80 \le g_1(x) \le 100 \\ 0 & ; g_1(x) \ge 100 \end{cases}$$

$$\mu_2 = \begin{cases} 0 & ; g_2(x) < 2 \\ \dfrac{g_2(x) - 2}{5 - 2} & ; 2 \le g_2(x) \le 5 \\ 1 & ; g_2(x) = 4 \\ \dfrac{5 - g_2(x)}{5 - 4} & ; 4 \le g_1(x) \le 5 \\ 0 & ; g_2(x) \ge 5 \end{cases}$$

$$x_1 + x_2 + x_3 + x_4 + x_5 - x_8 \geq 3$$
$$x_1 + x_3 + x_4 + x_5 + x_6 - x_9 \geq 3$$
$$x_1 + x_4 + x_5 + x_6 + x_7 - x_{10} \geq 8$$
$$x_1 + x_2 + x_5 + x_6 + x_7 - x_{11} \geq 8$$
$$x_1 + x_2 + x_3 + x_6 + x_7 - x_{12} \geq 8$$
$$x_2 + x_3 + x_4 + x_6 + x_7 - x_{13} \geq 3$$
$$x_2 + x_3 + x_4 + x_5 + x_7 - x_{14} \geq 3$$
$$x_i \geq 0$$
$$x_8, x_9, x_{10}, x_{11}, x_{12}, x_{13}, x_{14} \in [0,1]$$
$$\lambda \leq \mu_1$$
$$\lambda \leq \mu_2$$
$$\lambda \leq 1$$

The optimum solution obtained by LINGO is:

$x_1= 0$, $x_2=1$, $x_3=0$, $x_4=1$, $x_5=0$, $x_6=3$, $x_7=2$, $x_8=1$, $x_9=1$, $x_{10}=0$, $x_{11}=0$, $x_{12}=0$, $x_{13}=0$, $x_{14}=1$

$\mu_1=0.80$, $\mu_2=1.0$ and $\lambda=0.80$.

The solution obtained by TS algorithm is given in table 4.1:

Table 4.1 Solution for test problem 1. (Max-min method)

Parameter set	Tabu list size(T)=7; neighborhood size(N)=10; number of iterations(iter)=1000
CPU time	3sec
Solution	$x_1= 0$, $x_2=1$, $x_3=0$, $x_4=1$, $x_5=0$, $x_6=3$, $x_7=2$, $x_8=1$, $x_9=1$, $x_{10}=0$, $x_{11}=0$, $x_{12}=0$, $x_{13}=0$, $x_{14}=1$, $\mu_1=0.80$, $\mu_2=1.0$ and $\lambda=0.80$.

As can be seen from table 4.1, tabu search algorithm found the optimum solution. According to solution, the first goal is satisfied with 0.80 and the second goal is fully satisfied. So the overall satisfaction level λ is found as 0.80.

If we use additive model in which the sum of achievement degrees of goals are maximized, the optimum solution is found as in Table 4.2. LINGO also gives the same solution with TS.

Table 4.2. Solution of test problem 1 (additive method)

Parameter set	Tabu list size(T)=7; neighborhood size(N)=10; number of iterations(iter)=1000
CPU time	4 sec.
Solution	x_1= 4, x_2=0, x_3=0, x_4=0, x_5=0, x_6=1, x_7=4, x_8=1, x_9=1, x_{10}=0, x_{11}=0, x_{12}=0, x_{13}=1, x_{14}=1, μ_1=0.80, μ_2=1.0 and $\Sigma\ \mu$ =1.80.

4.2.1.2 Test Problem 2 (Baykasoglu et al., 1999)

Below test problem is also adopted from Baykasoglu et.al (1999). The original GP problem is modified in order to adopt FGP model. The upper and lower bounds for membership functions are obtained by pay-off tables. The modified model contains 8 fuzzy goals.

$$\max \quad \lambda$$

subject to

$$\begin{aligned}
g_1(x) &= 45.48x_1 + 37.32x_2 + 47.47x_3 + 30.23x_4 + 31.37x_5 \cong 120 \\
g_2(x) &= 150x_1 + 120x_2 + 90x_3 + 20x_4 + 80x_5 \cong 232 \\
g_3(x) &= 66.32x_1 + 48.37x_2 - 41.17x_3 - 3x_4 - 40x_5 \cong 4.2 \\
g_4(x) &= 58.13x_1 + 58.13x_2 + 48.13x_3 - 30x_4 + 38.61x_5 \cong 75 \\
g_5(x) &= 58.13x_1 + 39.20x_2 + 87.66x_3 + 72.12x_4 + 29.20x_5 \cong 195 \\
g_6(x) &= 58.70x_1 + 49.24x_2 + 96.72x_3 + 68.80x_4 + 29.24x_5 \cong 217 \\
g_7(x) &= 105.5x_1 + 83.4x_2 + 92.4x_3 + 46.8x_4 + 54.1x_5 \cong 225 \\
g_8(x) &= 1.086x_1 + 1.624x_2 + 0.946x_3 + 0.374x_4 + 0.438x_5 \cong 3
\end{aligned}$$

$$\mu_1 = \begin{cases} 0 & ; g_1(x) < 100 \\ \dfrac{g_1(x)-100}{120-100} & ; 100 \le g_1(x) \le 120 \\ 1 & ; g_1(x) = 120 \\ \dfrac{130-g_1(x)}{130-120} & ; 120 \le g_1(x) \le 130 \\ 0 & ; g_1(x) \ge 130 \end{cases} \qquad \mu_2 = \begin{cases} 0 & ; g_2(x) < 220 \\ \dfrac{g_2(x)-220}{232-220} & ; 220 \le g_2(x) \le 232 \\ 1 & ; g_2(x) = 232 \\ \dfrac{250-g_2(x)}{250-232} & ; 232 \le g_2(x) \le 250 \\ 0 & ; g_2(x) \ge 250 \end{cases}$$

$$\mu_3 = \begin{cases} 0 & ; g_3(x) < 3 \\ \dfrac{g_3(x) - 3}{4.2 - 3} & ; 3 \le g_3(x) \le 4.2 \\ 1 & ; g_3(x) = 4.2 \\ \dfrac{5 - g_3(x)}{5 - 4.2} & ; 4.2 \le g_3(x) \le 5 \\ 0 & ; g_3(x) \ge 5 \end{cases} \qquad \mu_4 = \begin{cases} 0 & ; g_4(x) < 70 \\ \dfrac{g_4(x) - 70}{75 - 70} & ; 70 \le g_4(x) \le 75 \\ 1 & ; g_4(x) = 75 \\ \dfrac{80 - g_4(x)}{80 - 75} & ; 75 \le g_4(x) \le 80 \\ 0 & ; g_4(x) \ge 80 \end{cases}$$

$$\mu_5 = \begin{cases} 0 & ; g_5(x) < 190 \\ \dfrac{g_5(x) - 190}{195 - 190} & ; 190 \le g_5(x) \le 195 \\ 1 & ; g_5(x) = 195 \\ \dfrac{210 - g_5(x)}{210 - 195} & ; 195 \le g_5(x) \le 215 \\ 0 & ; g_5(x) \ge 215 \end{cases} \qquad \mu_6 = \begin{cases} 0 & ; g_6(x) < 210 \\ \dfrac{g_6(x) - 210}{217 - 210} & ; 210 \le g_6(x) \le 217 \\ 1 & ; g_6(x) = 217 \\ \dfrac{220 - g_6(x)}{220 - 217} & ; 217 \le g_6(x) \le 220 \\ 0 & ; g_6(x) \ge 220 \end{cases}$$

$$\mu_7 = \begin{cases} 0 & ; g_7(x) < 215 \\ \dfrac{g_7(x) - 215}{225 - 215} & ; 215 \le g_7(x) \le 225 \\ 1 & ; g_7(x) = 225 \\ \dfrac{230 - g_7(x)}{230 - 225} & ; 225 \le g_7(x) \le 230 \\ 0 & ; g_7(x) \ge 230 \end{cases} \qquad \mu_8 = \begin{cases} 0 & ; g_8(x) < 2 \\ \dfrac{g_8(x) - 2}{3 - 2} & ; 2 \le g_8(x) \le 3 \\ 1 & ; g_8(x) = 3 \\ \dfrac{4 - g_8(x)}{4 - 3} & ; 3 \le g_8(x) \le 4 \\ 0 & ; g_8(x) \ge 4 \end{cases}$$

$x_i \le 1, \quad i = 1, 2, ..., 5;$
$x_i \in [0,1]$
$\lambda \le \mu_z, \quad z = 1, 2, ..., 8;$
$\lambda \le 1$

The optimum solution found by LINGO is: x1= 0, x2=1, x3=1, x4=1, x5=0 μ1=0.75, μ2=0.83, μ3=1, μ4=0.75, μ5=0.75,μ6=0.68, μ7=76 , μ8=0.84 , and λ=0.68. The same problem is solved using TS algorithm. TS found the optimum solution. The TS parameters and the obtained solution are given in Table 4.3.

Table 4.3 The solution of test problem 2

Parameter set	Tabu list size(T)=7; neighborhood size(N)=4; number of iterations(iter)=300
CPU time	4 sec.
Solution	x_1= 0, x_2=1, x_3=1, x_4=1, x_5=0 μ_1=0.75, μ_2=0.83, μ_3=1, μ_4=0.75, μ_5=0.75;μ_6=0.68, μ_7=76 , μ_8=0.84 , and λ=0.68.

4.2.1.3 Test Problem 3 (Gen, Ida, Tsujimira and Kim, 1993)

Gen et al (1993) proposed a 0-1 FGP model for solving reliability problems. They formulated the model as a preemptive FGP. This model is modified and formulated using max-min method. This modified model is solved by TS algorithm. The mathematical model is given as follows:

max λ

subject to

$$g_1(x) = 24.78x_1 + 23.3.x_2 + 38.7x_3 + 50.2x_4 + 123.3x_5 + 243.4x_6$$
$$+332.6x_7 + 417x_8 + 26.3x_9 + 112.1x_{10} + 161.3x_{11} + 204.6x_{12} \cong 308$$
$$g_2(x) = 16x_1 + 25x_2 + 36x_3 + 49x_4 + 1x_5 + 4x_6 + 9x_7 + 16x_8$$
$$+9x_9 + 16x_{10} + 25x_{11} + 36x_{12} \cong 51$$
$$g_3(x) = 27.4x_1 + 42.7x_2 + 60.1x_3 + 80.1x_4 + 27.4x_5 + 42.7x_6 + 60.1x_7$$
$$+80.1x_8 + 27.4x_9 + 42.7x_{10} + 60.1x_{11} + 80.1x_{12} \cong 130$$
$$g_4(x) = 15.6x_1 + 24.3x_2 + 28.3x_3 + 29.3x_4 + 15.6x_5 + 24.3x_6 + 28.3x_7 + 29.3x_8$$
$$+15.6x_9 + 24.3x_{10} + 28.3x_{11} + 29.3x_{12} \cong 70$$

$$\mu_1 = \begin{cases} 0 & ; g_1(x) < 200 \\ \dfrac{g_1(x)-200}{260-200} & ; 200 \le g_1(x) \le 260 \\ 1 & ; g_1(x) = 260 \\ \dfrac{300-g_1(x)}{300-260} & ; 260 \le g_1(x) \le 300 \\ 0 & ; g_1(x) \ge 300 \end{cases} \qquad \mu_2 = \begin{cases} 0 & ; g_2(x) < 40 \\ \dfrac{g_2(x)-40}{51-40} & ; 40 \le g_2(x) \le 51 \\ 1 & ; g_2(x) = 51 \\ \dfrac{60-g_2(x)}{60-51} & ; 51 \le g_2(x) \le 60 \\ 0 & ; g_2(x) \ge 60 \end{cases}$$

$$\mu_3 = \begin{cases} 0 & ; g_3(x) < 100 \\ \dfrac{g_3(x)-100}{130-100} & ;100 \le g_3(x) \le 130 \\ 1 & ; g_3(x) = 130 \\ \dfrac{135-g_3(x)}{135-130} & ;130 \le g_3(x) \le 135 \\ 0 & ; g_3(x) \ge 135 \end{cases} \qquad \mu_4 = \begin{cases} 0 & ; g_4(x) < 65 \\ \dfrac{g_4(x)-65}{70-65} & ;65 \le g_4(x) \le 70 \\ 1 & ; g_4(x) = 70 \\ \dfrac{75-g_4(x)}{75-70} & ;70 \le g_4(x) \le 75 \\ 0 & ; g_4(x) \ge 75 \end{cases}$$

$$x_i \le 1, \quad i = 1,2,...,12;$$
$$x_i \in [0,1]$$
$$\lambda \le \mu_z, \quad z = 1,...4;$$
$$\lambda \le 1$$

The optimum solution obtained by LINGO is: x1=0; x2=0; x3=1; x4=0; x5=1; x6=0; x7=0; x8=0; x9=0; x10=1; x11=0; x_{12}=0; μ_6=0.68, μ_7=0.76, μ_8=0.84 and λ=0.68.

Using max-min method, TS algorithm found the optimum solution given Table 4.4. The same solution vector is obtained by using additive method with Σμ=6.36.

Table 4.4 Solution of test problem 3 (max-min method)

Parameter set	Tabu list size(T)=7; neighborhood size(N)=4; number of iterations(iter)=100
CPU time	5 sec.
Solution	x_1= 0, x_2=1, x_3=1, x_4=1, x_5=0 μ_1=0.75, μ_2=0.83, μ_3=1, μ_4=0.75, μ_5=0.75 μ_6=0.68, μ_7=0.76, μ_8=0.84, and λ=0.68.

The results that have been presented above have shown us, the TS algorithm can be effectively used for solving FGP models. In the next section the deterministic version of cell formation model proposed in Chapter 3 will be solved by TS algorithm and the results are compared with LINGO results.

4.3 Solution of FGP Model for Cell Formation Using TS Algorithm (Deterministic Case)

In this section, a cell formation problem in which operations can be performed in alternative machines is solved by TS algorithm and LINGO to prove the applicability of TS in solving FGP models.

The manufacturing system considered in numerical example consists of 5 machines and performs three different jobs. Each job has three sub-operations and these operations can be performed in alternative machines. The machine requirements and the processing times of operations on alternative machines are given in Table 4.5. The minimum and maximum numbers of machines that can be included in a cell is given as 2 and 3 respectively. Aspiration levels for goals are given in Table 4.5

Table 4.5 Alterative process plans and processing times

Job	Operation	Alternative process plan	Processing time (min.)
	A1	1, 4, 5	5 - 4 - 2
JOB1	A2	3, 4	4 – 8
	A3	1, 5	7 – 6
	B1	4,5	4 – 7
JOB2	B2	1, 2, 3	3 – 5 – 6
	B3	5	4
	C1	4, 5	5 – 7
JOB3	C2	1, 4	4 – 4
	C3	1,2, 5	3 – 2 – 5

For the solution, FGP model developed in Section 3.2 is used. However for comparing the results of TS with LINGO, the goals obtained by simulation models such as utilization, mean tardiness etc. are not included and two analytic goals are employed in solving numerical example. Instead of stochastic demand and processing times, deterministic inputs are also used for comparing the results with LINGO. The LINGO model is given in Appendix A1.

Table 4.6 Aspiration levels for goals

Goal	Min-Max limits	
Number of exceptional elements	0	3
Total completion time of parts	35	40

The model is same with the model build in section 3.2 except for goals. The goals of the model are obtained by following equations:

$$g_1 = \sum_i \sum_o \sum_c D_{1ioc} \qquad (number\ of\ exceptional\ elements)$$
$$g_2 = \sum_i \sum_o \sum_c \sum_m A_{iom} X_{iocm} \qquad (total\ completion\ time\ of\ parts) \tag{4.1}$$

Where A_{iom} = The time required to accomplish the o th operation of i th job.

The membership functions of the goals are;

$$\mu_1 = \begin{cases} 1 & if\ g_1 \le L_1 \\ \dfrac{U_1 - g_1}{U_1 - L_1} & if\ L_1 \le g_1 \le U_1 \\ 0 & if\ g_1 \ge U_1 \end{cases}$$

$$\mu_2 = \begin{cases} 1 & if\ g_2 \le L_2 \\ \dfrac{U_2 - g_2}{U_2 - L_2} & if\ L_2 \le g_2 \le U_2 \\ 0 & if\ g_2 \ge U_2 \end{cases}$$

The FGP model for cell using max-min method is as given below:

$$Max Z = \lambda$$
$$subject\ to$$

$$\mu_1 \ge \lambda$$
$$\mu_2 \ge \lambda$$
$$0 \le \lambda \le 1$$
$$g_1 = \sum_i \sum_o \sum_c D_{1ioc} \prec goal\_a$$
$$g_2 = \sum_i \sum_o \sum_c \sum_m A_{iom} X_{iocm} \prec goal\_b$$

And other system constraints (Equations 3.28 - 3.36 in page 68 of Chapter 3).

Where goal_a and goal_b represents the aspiration levels for g_1 and g_2 respectively.

Table 4.7 LINGO solution for test problem 4

$Q_1=1$, $Q_2=1$	$\mu_1=0.6667$; $\mu_1=1$ and $\lambda=0.6667$
$Y_{12}=1$; $y_{14}=1$;$y_{21}=1$,$y_{23}=1$;$y_{25}=1$	
X_{1125}, X_{1223}, X_{1325}, X_{2114}, X_{2221}, X_{2325}, X_{3114}, X_{3214}, $X_{3312}=1$	

LINGO solution is illustrated in Table 4.7. According to LINGO solution, 2 cells are formed. The cell formations and part assignments are given in Table 4.8. The first cell is composed of machines 2 and 4 and the second cell composed of machines 1, 3 and 5. According to cell formation and part assignments obtained from LINGO, the number of exceptional element is 1. Hence the satisfaction level of the first goal is $\mu_1 = 0.6667$ ($\mu 1 = (3-1) / (3-0)$). Total completion time for parts is found as 35 min. So the satisfaction level of goal 2, $\mu_2 = 1$. The overall satisfaction level is λ is found as 0.6667.

Table 4.8 Cell formation and part assignments (LINGO solution)

CELL 1		CELL 2	
Machines	Operations	Machines	Operations
2	C3	1	B2
4	B1, C1, C2	3	A2
		5	A1, A3, B3

Then the same problem is solved using TS algorithm. The solution vector for TS algorithm is composed of Xiocm variables which show the assignment of parts to the specific machine in a cell.

$$X_{iocm} = \begin{cases} 1 & \textit{if } O_{th} \textit{ operation of job } j \textit{ is assigned to machine } m \textit{ in cell } c \\ 0 & \textit{Otherwise} \end{cases}$$

For example if $X_{1114}=1$, the first operation of the 1^{st} job is assigned to Machine 4 in cell 1. Considering the alternative routes and maximum number of cells, the solution vector is as given below:

X_{iocm}= [X_{1111}, X_{1114}, X_{1115}, X_{1121}, X_{1124}, X_{1125}, X_{1213}, X_{1214}, X_{1223}, X_{1224}, X_{1311}, X_{1315}, X_{1321}, X_{1325}, X_{2114}, X_{2115}, X_{2124}, X_{2125}, X_{2211}, X_{2212}, X_{2213}, X_{2221}, X_{2222}, X_{2223}, X_{2315}, X_{2325}, X_{3114}, X_{3115}, X_{3124}, X_{3125}, X_{3211}, X_{3214}, X_{3221}, X_{3224}, X_{3311}, X_{3312}, X_{3315}, X_{3321}, X_{3322}, X_{3325}]

The solution can be divided into 9 intervals each of which shows the assignment of an operation of a job. The intervals are illustrated in Figure 4.3.

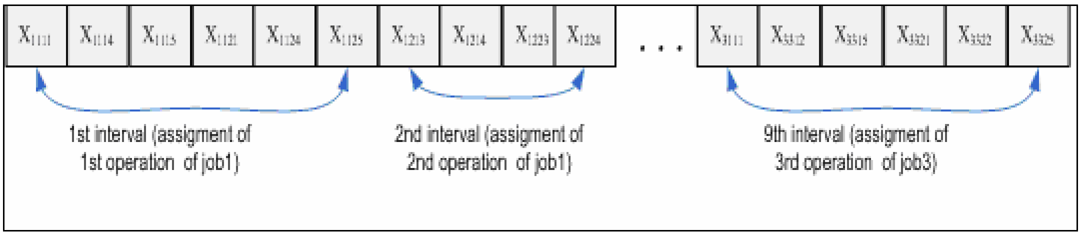


Figure 4.3 The structure of a solution vector.

According to the hard constraints of the model developed in 3.2, an operation can be assigned to a specific machine. Batch splitting or preemption is not allowed. Hence, in applying move strategy given by equation 1, only one Xiocm in each interval can take 1 value at a time. Other Xiocm variables in interval should be 0. By this way, we do not have to control the all constraints of the mathematical model in the solution process of TS algorithm. In other words, in TS solution, only the constraints (3.29), (3.33), (3.34), (3.35) and goal constraints are controlled for checking the feasibility of a solution. For instance in LINGO formulations of the model, 225 constraints are used. In TS, the problem is represented by 40 decision variables and 22 constraints. This relaxation will decrease the complexity of the model.

At each iteration, a predefined number of neighborhood solutions are generated from the initial solution. The generation process is illustrated in Figure 4.4. These neighbor solutions must be feasible. In other words, generated neighbor solution should satisfy the constraints (eq.3.29; eq.3.33; eq.3.34 and eq.3.35) of the model. For instance, assuming the neighborhood size is equal to 2, 1st neighbor solution is generated from the initial solution by changing the value of first interval. The 1 is moved from first digit to third digit. This means that the assignment of 1st operation of job 1 is changed from machine 1 in cell 1 to machine 5 in cell 2. This move recorded in tabu list and the corresponding λ value is calculated as 0.2 for that neighbor solution. The second neighbor is generated from the initial solution by changing the assignment of 2nd operation of job 2 from machine 5 in cell 2 to machine 4 in cell 2. This move is also recorded in tabu list. The corresponding λ value is calculated as 0.35. Hence the second neighborhood solution with highest λ (highest Σμ for additive method) value is taken as a new seed. Since the λ value of the new seed (λ =0.35) is better than the previous solution (λ =0.00). The best λ value is updated as 0.35. The nest neighborhood solutions are generated from this new seed. This process is repeated until termination conditions are reached.

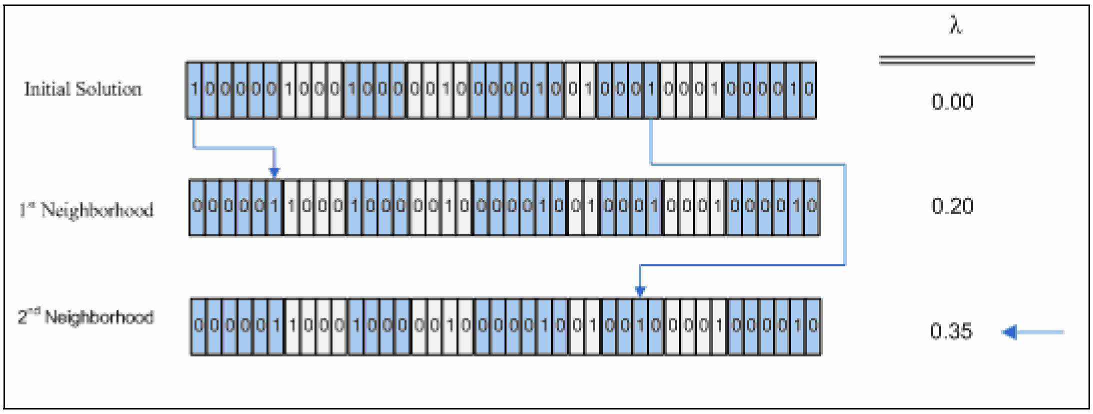


Figure 4.4 Neighborhood generation process

TS algorithm found the best solution at 67th step. The solution vector and corresponding μ and λ values are given in Table 4.9. According to the optimum solution, the first goal (min. number of exceptional elements) is fully satisfied and the satisfaction level of the second goal is μ_2=0.6667. The overall satisfaction level is λ=0.6667. The cell formation and the part assignments are same with the solution obtained by LINGO (see Table 4.8).

Table 4.9 Solution obtained by TS

Parameter set	Tabu list size(T)=7; neighborhood size(N)=4; number of iterations(iter)=150
CPU time	5 sec.
Solution	X_{1115}, X_{1213}, X_{1315}, X_{2124}, X_{2211}, X_{2315}, X_{3124}, X_{3224}, X_{3322} = 1 μ_1=1.00 , μ_2=0.6667 , and λ=0.6667.

In addition to optimum solution, TS algorithm gives the alternative solutions with the λ=0.6667. For instance the solution found in 45th step (see Figure 4.5) is also optimum (λ=0.6667). This solution is presented in Table 4.10.

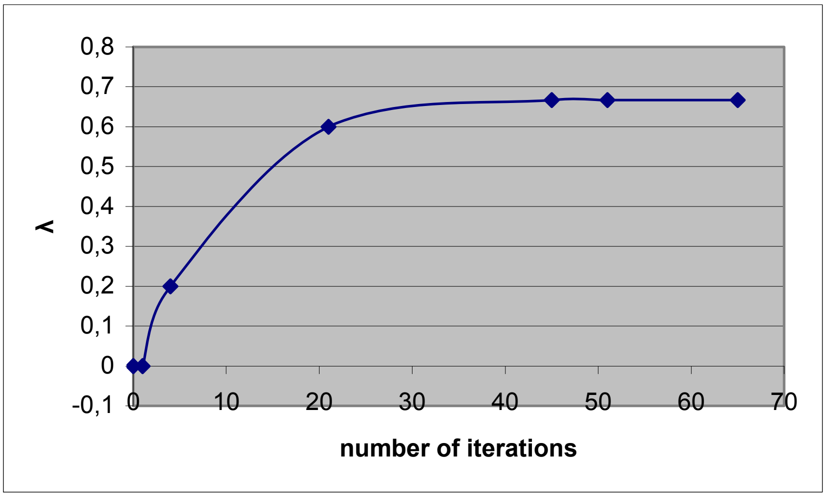


Figure 4.5. The behavior of λ (for max-min method)

According to this alternative solution, machine groups and part assignments are as shown in Table 4.11. As can be seen from Table 4.10, although this solution gives optimum λ value (0.6667), the satisfaction level of goal 1 is 0.80 which is lower than the solution presented in Table 4.9.

Table 4.10. An alternative solution obtained by TS algorithm

Parameter set	Tabu list size(T)=7; neighborhood size(N)=4; number of iterations(iter)=150
CPU time	5 sec.
Solution	X_{1115}, X_{1213}, X_{1315}, X_{2124}, X_{2222}, X_{2315}, X_{3124}, X_{3224}, X_{3322} = 1 μ_1=0.80, μ_2=0.6667, and λ=0.6667.

Table 4.11 Cell formation and part assignments for alternative solution

CELL 1		CELL 2	
Machines	Operations	Machines	Operations
3	A2	1	C2
5	A1, A3, B3	2	B2, C3
		4	B1, C1

Hence, when there is more than one solution with the same λ level, selecting the solution with the best $\Sigma\mu$ would be a good way for tie-breaking rule. For instance, although the solution presented in Table 4.9 and the solution given in Table 4.10 have same λ value (0.6667) the first solution has $\Sigma\mu$=1.6667 and the second one has $\Sigma\mu$=1.4667. Hence we can conclude that, the first solution is better than the first one.

Using additive method, LINGO and TS algorithm found same solution at 51st iteration (see Figure 4.6) with $\Sigma\mu$=1.6667. The corresponding solution vector is same with the solution presented in Table 4.6.

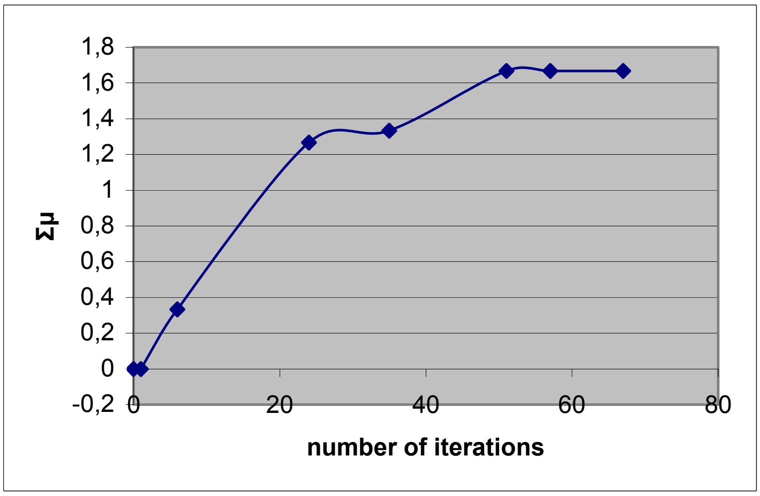


Figure 4.6. The behavior of $\Sigma\mu$ (additive method)

In this section, FGP models from the literature and the deterministic form of cell formation model developed in Chapter 3 are solved with LINGO optimization software and TS algorithm. In comparing results, both additive and max-min models are considered. The results showed us, TS algorithm can be used for solving FGP models effectively. In the next Chapter, hybrid simulation-analytic FGP model for cell formation in which some of goals are obtained by simulation model will be solved by TS.

4.5. Chapter Summary & Conclusions

Since the FGP model developed in Chapter 3 is a hybrid analytic-simulation model in which some of goals are obtained by a simulation model, the use of model independent solution approach such as tabu search, genetic algorithm, simulated annealing etc is required instead of classical solution approaches.

TS is an adaptive search procedure that can handle any type of variables and constraints. TS works with more than one solution at a time (neighborhood solutions). Among these solutions, a DM can choose any solution that satisfies his/her requirements best. This feature of TS is important in dealing with multiple objectives or goals. Hence, in this dissertation, a tabu search based solution methodology is used for solution of FGP models.

In this chapter, TS algorithm for solving FGP models is presented. The effectiveness of the algorithm is tested on several test problems from the literature. The results are compared with the results obtained by LINGO optimization software. Finally, the deterministic form of the FGP cell formation model developed in Chapter 3 is solved by TS algorithm considering two analytic goals (number of exceptional elements and total completion time for parts) and the results are again compared with results obtained by LINGO. Results showed that TS algorithm can be effectively used for solving both additive and max-min FGP models.

In the next chapter, TS algorithm will be used for solving the hybrid analytic-simulation FGP models in which some of goals are obtained by a simulation model.

CHAPTER FIVE

SOLUTION OF SIMULATION BASED HYBRID FGP MODELS FOR CELL FORMATION

In the previous chapter, a C-program was coded for solving FGP models. The effectiveness of TS based solution approach was also tested on several test problems. In this chapter, the solution approach will be extended for solving hybrid analytic-simulation FGP cell formation models. The general structure of the hybrid model was given in Chapter 3. As mentioned in Chapter 3, in this hybrid model, the achievement levels of some goals which are hard to represent analytically are obtained by a simulation model whereas the achievement levels of other goals are calculated analytically. As stated in earlier chapters, hybrid model also allows us to reflect the stochastic nature of the manufacturing system under consideration. The input such as processing times, part transfer times or part demand patterns can be easily reflected by simulation model which is integrated with TS based solution framework. Moreover, the goals such as mean tardiness, utilization etc. that are hard to represent analytically can be calculated by simulation model.

In Section 5.1 a base model is presented for illustrating the solution of hybrid FGP models using TS. Section 5.2 gives a brief explanation about ARENA simulation software and simulation models. The integration of TS based solution methodology with simulation will be discussed in Section 5.3. The structure of the C-program which is coded for solving hybrid analytic-simulation FGP models is presented in Section 5.4. The methodology is demonstrated on a small numerical example in Section 5.5. In this example, a manufacturing system which produces 3 different part types and consists of 5 machines is considered. The base hybrid model built in Section 5.1 which reflects the stochastic nature of the manufacturing system and routing flexibility is used for solution. Part demand rates, part processing times and part transfer times are all stochastic. The steps of TS algorithm for solving proposed hybrid model which is integrated with simulation model are illustrated on this numerical example.

Sections 5.6 through 5.10 cover the application of solution methodology for models with different objectives and assumptions.

In section 5.6, a manufacturing system which composed of 6 machines is considered. Six different jobs each of which have 3 sub-operations are performed. Each part can have different process plans. The processing times of operations on alternative machines are assumed to be equal. The base model built in Section 5.1 is used with the following fuzzy objectives: minimization of exceptional elements, maximization of system utilization, and minimization of time spent in the system. The achievement level of the first goal is calculated analytically whereas others are obtained from simulation. Max-min method and simple additive method are used for solution and the results are compared.

In Section 5.6, more complex manufacturing system which consists of 10 machines and 10 different part types (each part type is assumed to have 3 sub-operations). Again the base model built in Section 5.1 is used for solution. The solutions obtained by max-min method and additive method are compared.

Section 5.8 covers a numerical example in which a manufacturing system that composed 8 machines and 8 different parts. In this example the assumption of “equal number of sub-operations for each part type” is relaxed. In this example, parts can have different number of sub-opertions. The base model is adapted to the new assumption and used for solution. The max-min and simple additive methods are again used for solution and the results are compared.

In Section 5.9, a numerical example which composed of 6 machines and 6 products is presented. In this example, the base model is used with the following objectives: minimizing number of exceptional elements, maximizing system utilization, minimizing mean tardiness, and minimizing the percentage of tardy jobs. The achievement level of the first goal is calculated analytically whereas others are obtained from simulation. The simulation model and the C-codes are adapted for new objectives. The results obtained by max-min method and additive method are compared.

Section 5.10 contains a numerical example taken from Vin & De Lit (2005) and considers 12 parts and 6 machines. Number of operations for each part type varies between 1 and 3. Moreover, the processing times on alternative machines are also different. The model with tardiness objectives built in Section 5.9 is used for solution considering the max-min and additive methods and the results are compared.

Chapter summary and the conclusions obtained from experimental study are given in Section 5.11.

5.1. Solution of Simulation Based FGP Models for Cell Formation (Base model)

In this section, a base model will be used for illustrating the solution methodology and integration issues. The base model is similar to the model developed in Chapter 3 except for the goals. After illustrating the solution approach, models will be extended for new objectives and experiments.

The base model used to illustrate solution process contains three goals as given below:

$$Goal\_1: \sum_i \sum_o \sum_c D1_{ioc} \prec goal_{\text{int}} \qquad 5.1$$

$$Goal\_2: system \quad utilization \succ goal_{util} \qquad 5.2$$

$$Goal\_3: avg.time\ spent\ in\ system \prec goal_{time} \qquad 5.3$$

The objectives of the mathematical model are minimizing exceptional elements (Eq.5.1), maximizing system utilization (5.2), and minimizing time spent in the system (5.3). The first objective is determined by an analytical equation whereas other two objectives are determined by simulation model. The inter-cell movements should be substantially smaller than $goal_{excpt}$, system utilization should be substantially greater than $goal_{util}$, and the average time spent in the system should be substantially smaller than $goal_{time}$. Where, $goal_{int}$, $goal_{util}$, $goal_{time}$ represent the aspiration levels for goals. The other constraints of the model are same with the model developed in Chapter 3.

Constraints

$$\sum_c \sum_m X_{iocm} P_{iom} = 1$$

$$\sum_c Y_{cm} = 1 \qquad \forall m$$

$$X_{iocm} \leq K.Y_{cm} \qquad \forall i, o, c, m$$

$$X_{iocm} \leq K.Z_{ioc} \qquad \forall i, o, c, m$$

$$\sum_c Z_{ioc} = 1 \qquad \forall i, o$$

$$Z_{ioc} - Z_{ioc-1} = D_{1ioc} - D_{ioc} \qquad \forall i, o, c$$

$$\sum_m Y_{cm} \leq M_{\max} Q_c \qquad \forall c$$

$$\sum_m Y_{cm} \geq M_{\min} Q_C \qquad \forall c$$

$$X_{iocm}, Y_{cm}, Z_{ioc}, Q_c, D_{1ioc}, D_{ioc} = [0,1]$$

The membership functions of the goals are given below:

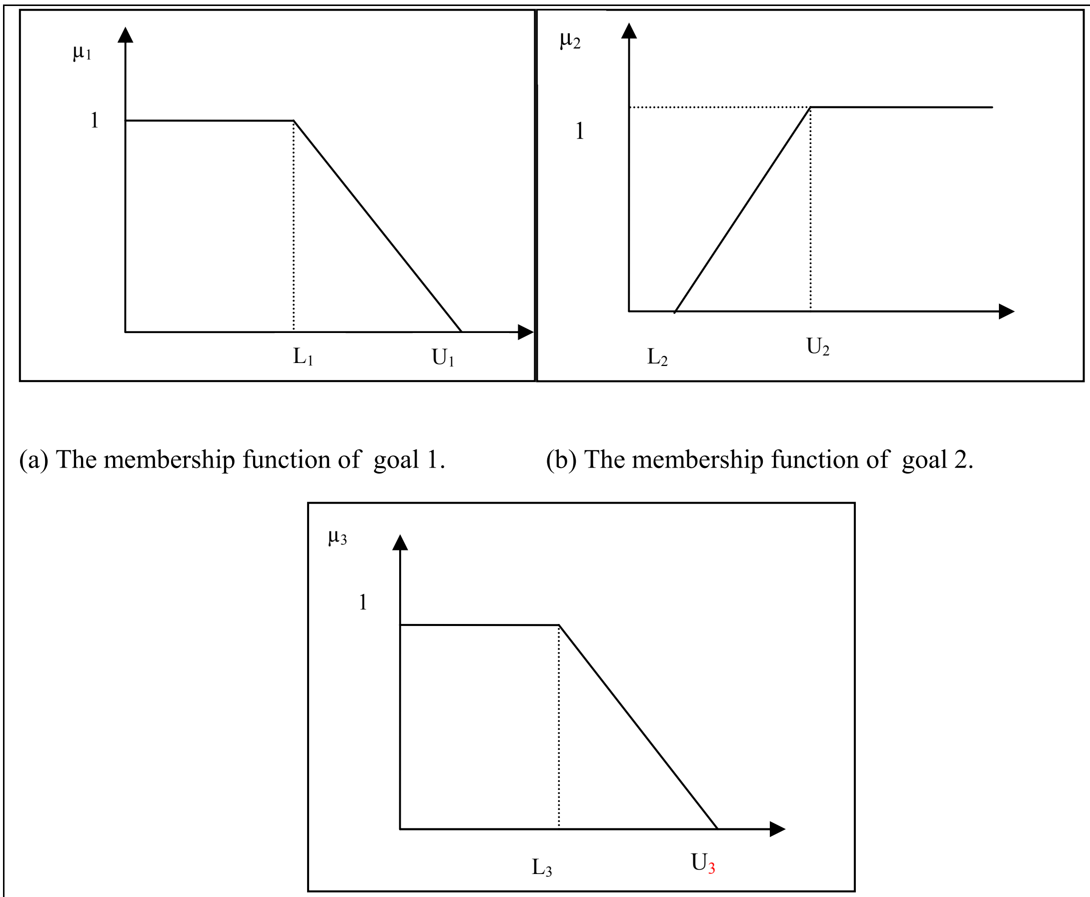
$$\mu_1 = \begin{cases} 1 & \text{if } f_1 \leq L_1 \\ \dfrac{U_1 - f_1}{U_1 - L_1} & \text{if } L_1 \leq f_1 \leq U_1 \\ 0 & \text{if } f_1 \geq U_1 \end{cases}$$

$$\mu_2 = \begin{cases} 1 & \text{if } f_2 \geq U_2 \\ \dfrac{f_2 - L_2}{U_2 - L_2} & \text{if } L_2 \leq f_2 \leq U_2 \\ 0 & \text{if } f_2 \leq L_2 \end{cases}$$

$$\mu_3 = \begin{cases} 1 & \text{if } f_3 \leq L_3 \\ \dfrac{U_3 - f_3}{U_3 - L_3} & \text{if } L_3 \leq f_3 \leq U_3 \\ 0 & \text{if } f_3 \geq U_3 \end{cases}$$

Where, f_i is the value of the i^{th} objective function; U_i and L_i are maximum and minimum limits of objectives.

The shapes of the membership functions are given in Figures 5.1. For example, in Figure1, the first goal is allowed to be spread to the right-hand side of L_1 with a certain range of r_1 ($r_1 = U_1 - L_1$).



(a) The membership function of goal 1. (b) The membership function of goal 2.

(c) The membership function of goal 3

Figure 5.1. The membership functions of goals.

As stated in Chapter 3, using the max-min operator λ, which is the overall satisfactory level of compromise, the standard goal programming formulation can be equivalently transformed as:

$$MaxZ = \lambda$$

$$\mu_1 \geq \lambda$$

$$\mu_2 \geq \lambda$$

$$\mu_3 \geq \lambda$$

And constraints (3.28-3.36).

In this hybrid model, the production the distributions of part processing times, part arrivals and part transfer times are included in simulation model. The goals such as system utilization and average time spent in the system are also obtained from simulation model. Number of exceptional elements is calculated using equation 5.1. The general structure of the simulation models will be given in the next section.

5.2. The General Structure of Simulation Models

Simulation models of manufacturing system are built in SIMAN-ARENA 3.0 simulation software. Arena is simulation and automation software developed by Rockwell Automation. It uses the SIMAN processor and simulation language.

Systems are typically modeled in ARENA using a process orientation where we model a particular system by studying the entities that flow through the system. The model consists of a graphical representation of the processes through which the entities move as they progress through the system.

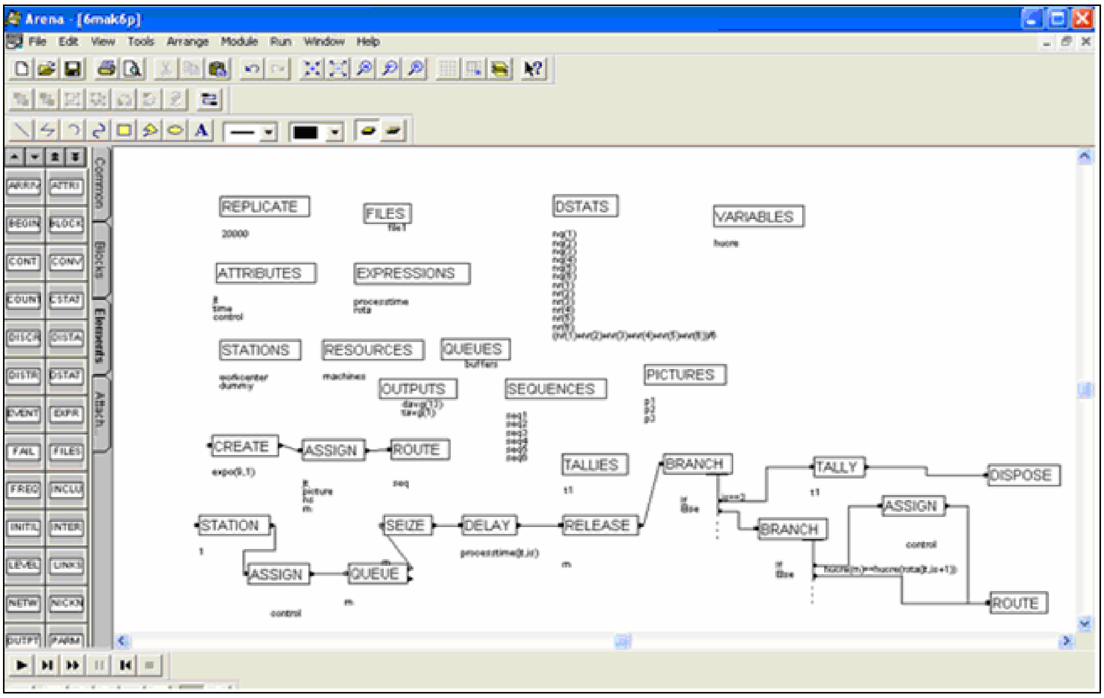


Figure 5.2 Sample model built in ARENA 3.0

The models are built in ARENA by placing and interconnecting modules to represent the process through which entities flow (Figure 5.2). Such models typically contain at least one source module that creates entity arrivals to the system and one or more sink modules that provide for the departure of these entities from the system.

A complete SIMAN-ARENA model consists of a **MODEL frame** and an **EXPERIMENT frame.** The model frame describes the logical flow of events within the system. Statements in the MODEL file are called “blocks”. The model file is a text file with the extension MOD (i.e. filename.mod). An example model file is given in Figure 5.3.

```
0$              CREATE,
1,0:expo(12,1):MARK(time);
1$              ASSIGN:
jt=disc(0.333,1,0.666,2,1,3,2):
                               picture=jt:
                               ns=jt:
                               m=dummy;
2$              ROUTE:         0.0,seq;


3$              STATION,       1-5;
17$             ASSIGN:        control=0;
4$              QUEUE,         m;
5$              SEIZE,         1:
                               m,1;
6$              DELAY:         processtime(jt,is);
7$              RELEASE:       m,1;
9$              BRANCH,        1:
                               If,is==3,11$,Yes:
                               Else,15$,Yes;
11$             TALLY:         t1,int(time),1;
10$             DISPOSE;

15$             BRANCH,        1:

If,hucre(m)==hucre(rota(jt,is+1)),16$,Yes:
                               Else,8$,Yes;
16$             ASSIGN:        control=1;
8$              ROUTE:
(control==1)*0+(control==0)*expo(2,3),seq;


12$             CREATE,        1,tfin:,1:MARK(time);
13$             WRITE,         file1,"%7.5f %8.3f\n":
                               davg(11),
                               tavg(1);
14$             DISPOSE;
```

Figure 5.3. Example MODEL file

The Experiment Frame specifies the experimental conditions for executing the model. Statements in the experiment file are called “elements”. The experiment file is a text file with the extension of exp (i.e., filename.exp). An example experiment file is given in Figure 5.4.

MODEL, EXPMT, and LINKER, translate SIMAN programs into a form the computer can understand. SIMAN itself takes these transformed programs and does the simulation. OUTPT then analyzes the output of the simulation. (Figure 5.5)

```
SEQUENCES:
                    1,seq1,5&3&1:
                    2,seq2,4&2&5:
                    3,seq3,4&1&5;
```

Part routing data

```
ATTRIBUTES:   1,jt:
          2,time:
          3,control;

FILES:      1,file1,"c:\endustri\arena\deneozg.txt",Sequential(),Free
Format,Error,No,Hold;

VARIABLES:           1,hucre(5),1,2,1,2,2;
```

Cell formation data

```
QUEUES:      5,buffers,FirstInFirstOut;

RESOURCES:   5,machines,Capacity(1,),-,Stationary;

STATIONS:    5,workcenter:
          6,dummy;

TALLIES:     1,t1;

EXPRESSIONS: 1,processtime(3,3),5,7,7,8,6,4,7,4,5:
          10,rota(3,3),5,4,4,3,2,1,1,5,5;

DSTATS:      1,nq(1):
          2,nq(2):
          3,nq(3):
          4,nq(4):
          5,nq(5):
          6,nr(1):
          7,nr(2):
          8,nr(3):
          9,nr(4):
          10,nr(5):
          11,nr(1)+nr(2)+nr(3)+nr(4)+nr(5);

OUTPUTS:     1,davg(11):
          2,tavg(1);

REPLICATE,   1,0.0,100000,Yes,Yes,10000;
```

Figure 5.4. An example experiment file.

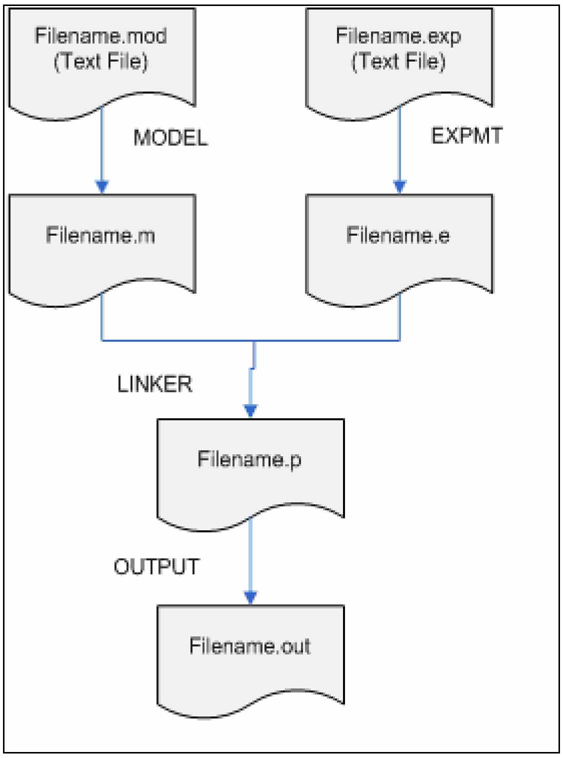


Figure 5.5 ARENA-SIMAN runtime procedure.

As stated above, the TS based solution approach presented in Chapter 3 is extended to solve hybrid simulation –analytic FGP models in which some goals are obtained by simulation model. For this purpose, the solution methodology is integrated with simulation model. The integration issues will be explained in next chapter.

5.3 The Integration with Simulation Model

A TS based solution procedure presented in previous chapter is used for solving the hybrid model. Since the hybrid-model is integrated with simulation model, the solution approach is extended to include the interactions between simulation model and solution approach. The codes of the developed C-program are given in Appendix A2. The flowchart of C program is illustrated in Figure 5.6.

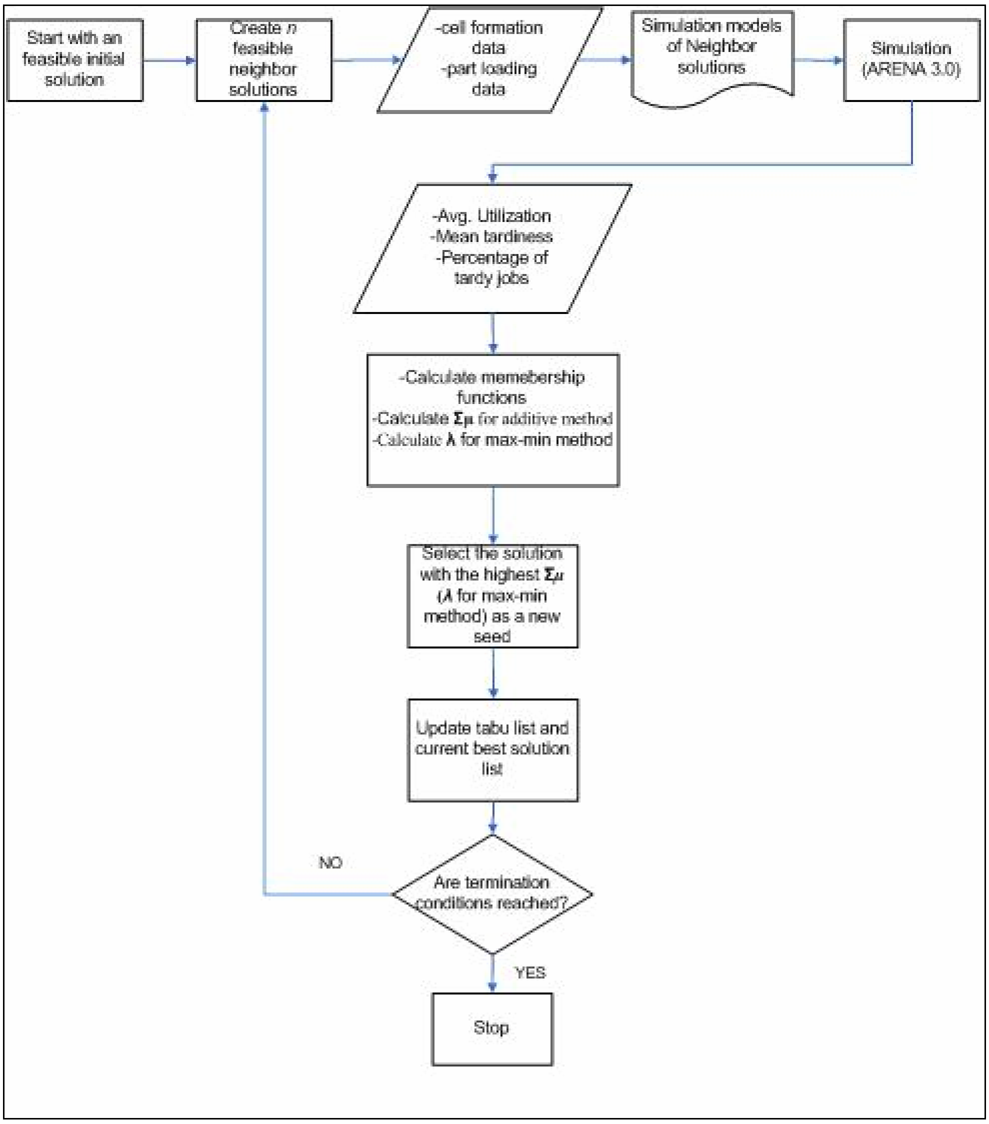


Figure 5.6 TS algorithm for solving hybrid analytic-simulation FGP model.

As seen in Figure 5.6, simulation model uses the *part routing data* and *cell formation* data. When TS algorithm creates neighborhood solutions, the part routing data and cell formation data is sent to the simulation model and the simulation model is modified for new conditions automatically. The interactions between TS algorithm and Simulation are illustrated in Figure5.7. When TS algorithm creates a new neighborhood solution ARENA experiment file is modified automatically. As can be seen from Figure 5.4, the modifications are take place in part routes (in Sequences Element) and cell formation data (in Variables Element). When a new solution is created, part routes and cell formation are changed in experiment file.

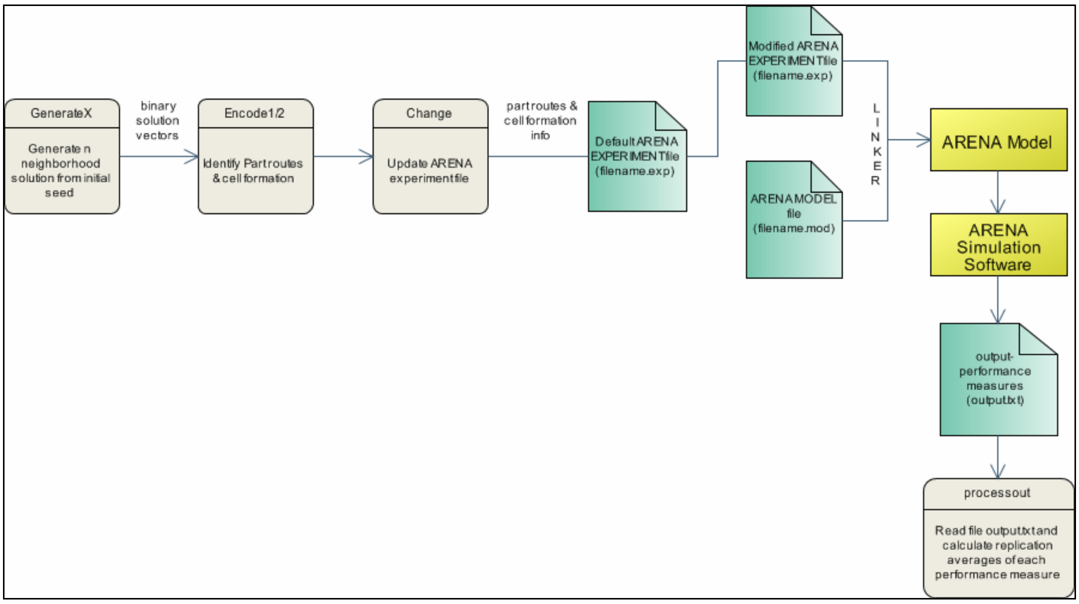


Figure 5.7 The interactions between TS algorithm and simulation model (Data flow diagram)

After the simulation model is updated automatically, ARENA simulation software executes the simulation model and provides the simulation based objectives such as utilization, average time spent in system etc. The other objective (number of exceptional element) is calculated using analytic equation (eq.5.1) by C-program. Then the membership functions and Σμ values (λ values for max-min method) for each neighborhood solution are calculated. The solution with the highest Σμ value (highest λ value for max-min method) is selected as a new seed. The procedure is terminated when termination conditions are reached.

5.4 The General Structure of the C-Program

The developed C-program consists of several functions. The main functions are as follows:

- **InitialX:** This function creates a feasible initial solution that satisfies the constraints of the model. The developed model also allows us to start with a known good solution.

- **GenerateX:** This function generates new solutions (neighborhood solutions) from initial solution using move strategy given in Section 4.2. Note that, the feasibility of neighborhood solution is also checked.

- **Encode1:** This function specifies the part assignments according to the solution under consideration. The machines that perform operations are specified and coded as to be recognized by simulation model.

- **Encode2**: Similar to Encode function, this function specifies the cell formation according to the neighborhood solution under consideration.

- **Change:** This function makes the necessary modifications on default experiment file of ARENA. The default experiment file contains the elements that are identical for all solutions. SEQUENCES element that contains part routes and VARIABLES element that contains machine cell formation data are modified according to solution under consideration. Other parts of the experiment file remain same for all solution. In default experiment file, the elements that will be modified are marked with character “ * ” (Figure 5.8). The change function of the C-Program opens the default experiment file and scans for “ * ” characters. The part routes are written in the field where the first “ * “ is encountered (in SEQUENCES element). Then the change function writes the cell formation info where the second “ * “ character is encountered in default experiment file. Part routes should be also included in rota expression in EXPRESSIONS element. Hence the change function writes the part route information in the field where the third ” * “ is encountered. Then modified experiment file is stored as a new file. Note that we can modify other elements in the same way if it is needed. For example if the processing times of alternative machines are different we should modify the “processtime” in EXPRESSION experiments according to the current solution. After the modified experiment file is obtained, the simulation software ARENA runs and provides the simulation based objective function values for the solution under consideration.

```
SEQUENCES:   *
ATTRIBUTES:  1,jt:
        2,time:
        3,control;
FILES:      1,file1,"c:\endustri\arena\deneozg.txt",Sequential(),Free Format,Error,No,Hold;
VARIABLES:   1,hucre(5),*
QUEUES:     5,buffers,FirstInFirstOut;
PICTURES:    1,p1:
        2,p2:
        3,p3;
RESOURCES:   5,machines,Capacity(1,),-,Stationary;
STATIONS:    5,workcenter:
        6,dummy;
TALLIES:     1,t1;
DSTATS:      1,nq(1):
        2,nq(2):
        3,nq(3):
        4,nq(4):
        5,nq(5):
        6,nr(1):
        7,nr(2):
        8,nr(3):
        9,nr(4):
        10,nr(5):
        11,(nr(1)+nr(2)+nr(3)+nr(4)+nr(5))/5;
OUTPUTS:     1,davg(11):
        2,tavg(1);
REPLICATE,   1,0.0,100000,Yes,Yes,10000;
EXPRESSIONS:
processtime(3,3),unif(5,9),unif(6,8),unif(5,9),unif(8,9),unif(7,9),unif(7,8),unif(6,9),unif(6,8),unif(7,9):
        rota(3,3),*
```

The fields marked with * are written when a new solution is

Figure5.8 Default EXPERIMENT file

- **Processoutfile:** The performance measures (goals) obtained by simulation model is recorded in a text file called "output.txt" for each replications. Processoutfile function reads output.txt file and obtains the replication averages of performance measures.

The general structure of the c-program and the interactions with the simulation model are demonstrated on the following numerical example.

5.5 An Illustrative Example

The manufacturing system under consideration consists of 5 machines and performs 3 different jobs. Each job consists of 3 operations and can have alternative process routes. The alternative routes for operations and processing times are given in Table 5.1. Processing times are uniformly distributed; part arrivals are exponentially distributed with a mean of 9 min. The minimum and maximum numbers of machines that can be assigned to a manufacturing cell are 2 and 3 respectively. Inter-cell part transfer times are exponentially distributed with a mean of 2 and the intra-cell transfer times are negligible. The base model in Section 5.1 will be considered for solving this example. Since the achievement levels of some goals are obtained by simulation model, the maximum and minimum levels of the goals are determined by pilot simulation runs. For this purpose, a number of simulation models that belong to different cell formations and part assignments are executed. The warm-up period for simulation runs is also determined by pilot studies. The variance reduction technique of *common random numbers* (Pegden, Shannon and Sadowski, 1990) is used for synchronization of random numbers so that the solutions are compared under similar conditions.

Table 5.1. Alternative routes and process times

Job	Operation	Alternative process plan	Processing time (min.)
	A1	1, 4, 5	Unif (5, 9)
JOB1	A2	3, 4	Unif (6, 8)
	A3	1, 5	Unif (5, 9)
	B1	4,5	Unif (8, 9)
JOB2	B2	1, 2, 3	Unif (7, 9)
	B3	5	Unif (7, 8)
	C1	4, 5	Unif (6, 9)
JOB3	C2	1, 4	Unif (6, 8)
	C3	1,2, 5	Unif (7, 9)

The tolerance values (min-max. limits) of goals are given in Table 5.2. The simulation model is built, tested and validated. The warm-up period is 10.000 and the replication length is chosen as 100.000 time units. Five independent replications for each alternative are performed.

Table 5.2 Aspiration levels for goals

Goal	Min-Max limits	
Min. Inter-cell movements	0	38
Max. System utilization	0.40	0.50
Min. Time spent in the system	30	40

5.5.1 Determination of TS Parameters

The TS parameters such as maximum number of iterations, tabu list size, neighborhood size should be determined. As stated in previous chapter, these parameters depend on the problem, type of constraints and variables. There is no certain rule for determining these parameters. These parameters are generally determined trial and error.

The maximum number of iterations should be big enough to assure convergence. The number of iterations should be increased for larger problems.

If variables are spread in a wide range, it is suggested to work with higher number of neighborhood solutions. Otherwise, smaller number of neighborhood solutions can be used. Since our model uses 0-1 variables, the small neighborhood sizes are used for solving the proposed models. As stated above, the related simulation model is run for predefined number of replication. So higher the number of neighborhood solutions longer the computation time. Plot studies showed that keeping the neighborhood size in range of 5-8 is enough for solving our examples.

In our experiments, tabu list size, minimum 7 and maximum 11 which is suggested by Glover and Laguna (1993).

For this example, tabu list size and neighborhood size are chosen as 7 and 5 respectively. Maximum number of iterations is chosen as 200.

5.5.2 Solution

Considering the alternative routes and maximum number of cells, the solution vector is composed of X_{iocm} variables (See Section 3.2) as given below:

S= [X_{1111}, X_{1114}, X_{1115}, X_{1121}, X_{1124}, X_{1125}, X_{1213}, X_{1214}, X_{1223}, X_{1224}, X_{1311}, X_{1315}, X_{1321}, X_{1325}, X_{2114}, X_{2115}, X_{2124}, X_{2125}, X_{2211}, X_{2212}, X_{2213}, X_{2221}, X_{2222}, X_{2223}, X_{2315}, X_{2325}, X_{3114}, X_{3115}, X_{3124}, X_{3125}, X_{3211}, X_{3214}, X_{3221}, X_{3224}, X_{3311}, X_{3312}, X_{3315}, X_{3321}, X_{3322}, X_{3325}]

At the first step, the default simulation model of the manufacturing system is built using ARENA 3.0. The parts of the default experiment file that will be modified are marked with the " * " character. As stated above, when TS creates a new solution (neighborhood solution) SEQUENCES (part routes) and VARIABLES (cell formations) elements in experiment file are modified. Assume that we started with the following initial solution:

S_1=[1,0,0,0,0,0,1,0,0,0,1,0,0,0,0,0,1,0,0,1,0,0,0,0,0,0,0,1,0,0,0,1,1,0,0,0,0,1,0,0,0,0]

The part assignments and machine cells that correspond to the initial solution are given in Table 5.3.

Table 5.3 Part assignment & cell formation (Initial solution)

CELL 1		CELL 2	
Machines	Operations	Machines	Operations
1	A1, A3, C2	4	B1
2	B2, C3	5	B3, C1
3	A2		

Encode 1 and Encode 2 functions of the C-Program convert the binary initial solution vector to part routes and machine cells that can be recognized by ARENA model. The change function of the C-program makes the necessary modifications on the default experiment file. Part routes and cell formations according to initial solution is written in the fields that is marked with " * " character in the default experiment file. The modified experiment file is as given in Figure 5.9

```
SEQUENCES:     1,seq1,1&3&1:
               2,seq2,4&2&5:
               3,seq3,4&1&2;

ATTRIBUTES:   1,jt:
              2,time:
              3,control;
FILES:     1,file1,"c:\endustri\arena\deneozg.txt",Sequential(),Free Format,Error,No,Hold;
VARIABLES:   1,hucre(5),   1,hucre(5),1,1,1,2,2;
QUEUES:    5,buffers,FirstInFirstOut;
PICTURES:   1,p1:
              2,p2:
              3,p3;
RESOURCES:   5,machines,Capacity(1,),-,Stationary;
STATIONS:   5,workcenter:
              6,dummy;
TALLIES:   1,t1;
DSTATS:    1,nq(1):
              2,nq(2):
              3,nq(3):
              4,nq(4):
              5,nq(5):
              6,nr(1):
              7,nr(2):
              8,nr(3):
              9,nr(4):
              10,nr(5):
              11,(nr(1)+nr(2)+nr(3)+nr(4)+nr(5))/5;
OUTPUTS:   1,davg(11):
              2,tavg(1);
REPLICATE,   1,0.0,100000,Yes,Yes,10000;
EXPRESSIONS:
processtime(3,3),unif  1.4.4.3.2.1.1.5.2:  (8,9),unif(7,9),unif(7,8),unif(6,9),unif(6,8),unif(7,9):
             rota(3,3),
```

The modified experiment file

Figure 5.9 Modified EXPERIMENT file

After the modifications are made, the modified experiment file is stored as a new file. After the modified experiment file is obtained, the simulation software ARENA runs and provides the simulation based objective function values (i.e. system utilization, average time spent in the system). The other objective (the number of exceptional elements) is calculated analytically by the C-Program. The objective functions for the initial solution are as follows:

G1 (number of exceptional elements) = 3; μ_1=0:
G2 (System utilization-obtained from simulation) = 0.4945; μ_2= 0.9450:
G3 (Avg. Time spent in the system-obtained from simulation) = 48.5960; μ_3= 0:

Hence $\lambda = \text{Min}(\mu_{1,}\ \mu_{2,}\ \mu_3) = 0$.

Then the feasible neighborhood solutions are created from initial seed. Assume that the below neighborhood solution is created.

S_2= [0,0,0,0,1,0,0,0,1,0,0,1,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0]

The part assignments and the cell formation that correspond to this solution are given in Table 5.4.

Table 5.4

CELL 1		CELL 2	
Machines	Operations	Machines	Operations
1	A3, C2	3	A2, B2
2	C3	4	A1, B1
		5	B3, C1

Part routings and machine cell formations are again written on default experiment file and the model for this solution is created automatically. ARENA executes the model and the objective function values for this solution are obtained as:

G1 (number of exceptional elements) = 2; μ_1=0.3333:
G2 (System utilization) = 0.4965; μ_2= 0.4965:

G3 (Avg. Time spent in the system) = 32.5860; μ_3= 0.7140:

The overall satisfaction level λ = min (μ_1, μ_2, μ_3) =0.3333.

The same steps are performed for each solution in a neighborhood. The solution with the highest λ (Σμ for additive method) is selected as a new seed. The other steps of the algorithm are same with original TS algorithm. Note that all these calculations are performed by C-program automatically.

The best solution is found at 38th iteration. The TS and simulation parameters and the best solution are given in Table 5.5.

Table 5.5 TS and simulation parameters and the best solution obtained.

Parameter set (TS)	Tabu list size(T)=7; neighborhood size(N)=5; number of iterations(iter)=150
Parameter set (Simulation)	Replication length:50000 min.; Number of replications:5: Warm up period: 10000 min.
CPU time	98 sec.
Solution	X_{1121}, X_{1223}, X_{1321}, X_{2124}, X_{2222}, X_{2325}, X_{3115}, X_{3214}, X_{3312} = 1 g_1=0 ; g_2=0.5030; g_3=31.2650 μ_1=1.0000, μ_2=1.0000, μ_3= 0.8735 and λ=0.8735.

According to the best solution, machine cell formation and part assignments are given in Table 5.6. As can be seen from Table.5.6, two machine cells are formed. The first cell is composed of machines 2, 4 and 5. The second cell covers the machines 1 and 3. According to the best solution, there are no exceptional elements and the membership function of goal1, μ_1=1.0000. The utilization level is obtained form the simulation model as 0.5030 which is greater than the upper limit of goal 2. Hence the membership function value of the second goal is μ_2=1.0000. The average time spent in the system is 31.265 min. Hence the membership function value of the third goal is μ_3= 0.8735. The overall satisfaction level of the system λ = min (μ_1, μ_2, μ_3) = 0.8735.

Table 5.6 Part assignment and cell formation according to the best solution.

CELL 1		CELL 2	
Machines	Operations	Machines	Operations
2	B2, C3	1	A1, A3
4	B1, C2	3	A2
5	B3, C1		

5.6 Numerical Example 1 (6 machines- 6 parts)

In this example, a manufacturing system which composed of 6 machines is considered. 6 different jobs each of which have 3 sub operations are performed in this manufacturing system. The alternative routes for operations and processing times are given in Table5.7. Processing times are uniformly distributed; part arrivals are exponentially distributed with a mean of 7 min. The minimum and maximum numbers of machines that can be assigned to a manufacturing cell are 2 and 3 respectively. Inter-cell part transfer times are exponentially distributed with a mean of 2 and the intra-cell transfer times are negligible. The tolerance values (min-max. limits) of goals are given in Table 5.8.

Note that, in this example, it is assumed that the processing times of alternative machines are same (i.e. alternative machines are identical). The number of operations for each job is also assumed to be equal (i.e. each job has 3 sub-operations). These assumptions will be relaxed in the next sections. It is also assumed that the parts arrive the system with batch size of 1. Job types are assigned with the discrete probability distribution as disc (0.1667, 1, 0.3333, 2, 0.5, 3, 0.6667, 4, 0.8333, 5, 1, 6). This means that the probability of all part types are equal (p = 0.1667).

The goals considered in this model are: number of exceptional elements (minimization.), system utilization (maximization), and time spent in the system (minimization), the first goal is determined by an analytical equation whereas others are obtained by simulation model. The Simulation model is built using ARENA 3.0 Simulation Software, tested and validated.

The warm-up period for simulation model is determined as 10.000 min. and the replication length is chosen as 100.000min. for this example. The number of independent replications is chosen as 5 for each alternative. The parameter set of tabu search algorithm is chosen by trial and error. Tabu list size and neighborhood size are chosen as 8 and 5 respectively. Maximum number of iterations is chosen as 300.

Table 5.7 Alternative process plans and processing times.

Job	Operation	Alternative process plan	Processing time
JOB1	A1	1, 5, 6	Unif(10, 13)
	A2	3, 4	Unif(7, 9)
	A3	1, 5	Unif(6, 9)
JOB2	B1	5, 6	Unif(8, 9)
	B2	1, 2, 3	Unif(8,10)
	B3	5	Unif(7, 9)
JOB3	C1	4,5	Unif(8,11)
	C2	1, 4	Unif(6,8)
	C3	1, 2, 5	Unif(8,10)
JOB4	D1	2	Unif(7, 9)
	D2	2, 3	Unif(7, 8)
	D3	3	Unif(7,10)
JOB5	E1	1, 2	Unif(8,10)
	E2	3	Unif(7, 9)
	E3	1, 4	Unif(7, 8)
JOB6	F1	4, 6	Unif(7,10)
	F2	6	Unif(8, 9)
	F3	3	Unif(9,10)

Table 5.8 Aspiration levels for goals

Goal	Min-Max limits	
Min. Inter-cell movements	3	8
Max. System utilization	0.30	0.75
Min. Time spent in the system	40	45

The proposed methodology was applied to the above case. The best solution is found at 89th iteration. The TS and simulation parameters and solution vector is given in Table 5.9

Table 5.9 TS and simulation parameters and the best solution obtained.

Parameter set (TS)	Tabu list size(T)=7; neighborhood size(N)=5; number of iterations(iter)=300
Parameter set (Simulation)	Replication length:100000 min.; Number of replications:5: Warm up period: 10000 min.
CPU time	262 sec.
Solution	X_{1126}, X_{1224}, X_{1315}, X_{2126}, X_{2211}, X_{2315}, X_{3115}, X_{3211}, X_{3312} , X_{4112} , X_{4212}, X_{4323}, X_{5111}, X_{5223}, X_{5324}, X_{6124}, X_{6226}, X_{6323}= 1 $g_1 = 4$; $g_2 = 0.5927$; $g_3 = 41.0875$ μ_1= 0.7500 , μ_2= 0.6504 , μ_3= 0.7825 and λ= 0.6504.

Using the max-min method, the best λ value is found as 0.6504. The solution is summarized in Table 5.10. According to the solution vector, 2 cells are formed. The first cell is composed of machines 1-2-5 and the second cell is composed of machines 3-4 and 6. There are 4 inter-cell movements and the satisfaction level of the first goal $\mu_1 = 0.75$. The system utilization level is found as 0.5927 and μ_2=0.6504. The average time spent in the system is found as 41.0875 min and μ_3=0.7825. So the overall satisfaction level is λ=0.6504.

Table 5.10 Machine Cells formation and part assignments according to the solution (max-min method)

CELL 1		CELL 2	
Machines	Operations	Machines	Operations
1	B2, C2, E1	3	D3, E2, F3
2	D1, D2, C3	4	A2, E3, F1
5	A3, B3, C1	6	A1, B1, F2

The proposed methodology is applied to the above case using simple additive method. The best solution is found at 197th iteration. The best $\Sigma\mu$ value is found as 2.6180. The solution is summarized in Table 5.11. According to the solution obtained using additive method, there are 2 inter-cell movements and the satisfaction level of the first goal $\mu_1 = 1$. The system utilization level is found as 0.5781 and μ_2=0.6180. The average time spent in the system is found as 39.376 min. and μ_3=1.

According to the solution obtained by max-min method, the achievement level of goal_2 (0.6504) is higher than simple additive method (0.6180). However, the achievement levels of other goals (μ_1=1 and μ_3=1) are higher than max-min method (μ_1=0.75, μ_3=0.7825). In simple additive method, the achievement levels of some goals will not decrease because of a particular goal that is difficult to achieve. This advantage makes the simple additive method appealing. As a whole, the sum of achievement levels of goals in the solution obtained by simple additive method is greater than max-min method.

Table 5.11 Machine Cells formation and part assignments according to the solution (additive method)

CELL 1		CELL 2	
Machines	Operations	Machines	Operations
1	A3, B2, C2	3	D3, E2, F3
2	D1, D2, E1	4	A2, E3, F1
5	B1, B3, C1, C3	6	A1, F2

5.7 Numerical Example 2 (10 machines-10 parts)

In this example, manufacturing system is composed of 10 machines and performs 10 jobs. Each job consists of 3 sub-operations. The alternative routes for operations and processing times are given in Table5.12. Processing times are uniformly distributed; part arrivals are exponentially distributed with a mean of 5 min.

Table 5.12 Alternative process plans and processing times.

Job	Operation	Alternative process plan	Processing time
JOB1	A1	1, 3, 10	Unif(4, 7)
	A2	2, 5, 7	Unif(5, 8)
	A3	4	Unif (6, 9)
JOB2	B1	6, 8, 10	Unif (7,10)
	B2	4, 8	Unif (8, 9)
	B3	9	Unif (2, 5)
JOB3	C1	2, 4, 7	Unif (6, 9)
	C2	6, 9	Unif (5, 7)
	C3	1, 3	Unif (4, 6)
JOB4	D1	4, 6	Unif (4, 7)
	D2	2, 4, 7	Unif (7, 9)
	D3	1, 5	Unif (6, 8)
JOB5	E1	5	Unif (8, 9)
	E2	3, 6, 10	Unif (7, 9)
	E3	9	Unif (6, 8)
JOB6	F1	8	Unif (7, 8)
	F2	5, 7, 9	Unif (8, 9)
	F3	3, 8	Unif (9, 11)
JOB7	G1	1, 2	Unif (5, 8)
	G2	3, 4, 6	Unif (8, 9)
	G3	6	Unif (5, 9)
JOB8	H1	2, 8, 9	Unif (6, 8)
	H2	10	Unif (8, 9)
	H3	7, 9	Unif (7, 8)
JOB9	I1	5, 6, 7	Unif (8, 9)
	I2	1, 2	Unif (7, 9)
	I3	3, 8	Unif (6, 8)
JOB10	J1	1	Unif (5, 9)
	J2	4, 7, 8	Unif (4, 7)
	J3	4	Unif (3, 6)

The minimum and maximum numbers of machines that can be assigned to a manufacturing cell are 2 and 4 respectively. Inter-cell part transfer times are exponentially distributed with a mean of 2 and the intra-cell transfer times are negligible. The tolerance values (min-max. limits) of goals are given in Table 5.13.

Table 5.13 Aspiration levels for goals

Goal	Min-Max limits	
Min. Inter-cell movements	0	5
Max. System utilization	0.30	0.65
Min. Time spent in the system	50	60

The proposed methodology was applied to the above case. The best solution is found at 311th iteration. The TS and simulation parameters and solution vector is given in Table 5.14.

Table 5.14 TS and simulation parameters and the best solution obtained.

Parameter set (TS)	Tabu list size(T)=7; neighborhood size(N)=5; number of iterations(iter)=300
Parameter set (Simulation)	Replication length:50000 min.; Number of replications:5: Warm up period: 10000 min.
CPU time	493 sec.
Solution	X_{1111}, X_{1212}, X_{1314}, X_{2128}, X_{2228}, X_{2329}, X_{3137}, X_{3236}, X_{3333} , X_{41124}, X_{4212}, X_{4311}, X_{5125}, $X_{522\ 10}$, X_{5329}, X_{6128}, X_{6225}, X_{6328}, X_{7111}, X_{7214}, X_{7336}, X_{8129}, $X_{822\ 10}$, X_{8329}, X_{9137}, X_{9212}, X_{9333}, $X_{10\ 111}$, $X_{10\ 214}$, $X_{10\ 314}$= 1 $g_1 = 3$; $g_2 = 0.5800$; $g_3 = 52.9644$ $\mu_1= 0.6000$, $\mu_2= 0.8000$, $\mu_3= 0.7036$ and $\lambda= 0.7036$

Using the max-min method, the best λ value is found as 0.7036. The solution is summarized in Table 5.15. According to the solution vector, 3 cells are formed. The first cell is composed of machines 1-2-4, the second cell is composed of machines 5-8-9-10 and cell 3 includes machines 3-6-7. There are 3 inter-cell movement and the satisfaction level of the first goal μ_1 = 0.6000. The system utilization level is found as 0.5800 and μ_2=0.8000. The average time spent in the system is found as 52.9694 min. and μ_3=0.7036. So the overall satisfaction level is λ=0.7036. The additive method gives the same solution with $\Sigma\mu$=2.104.

The problem size is depend on the number of alternatives for each operation, number of operations for each job, number of part types and the number of that can be formed .As can be seen from the solution, the computation time is increased when the problem size gets larger.

Table 5.15 Machine Cells formation and part assignments according to the best solution

CELL 1		CELL 2		CELL 3	
Machines	Operations	Machines	Operations	Machines	Operations
1	A1, D3, G1, J1	5	F1, F2	3	C3, I3
2	A2, D2, I2	8	B1, B2, F1, F3	6	C2, G3
4	A3, D1, G2, J2, J3	9	B3, E3, H1, H3	7	C1, I1
		10	E2, H2		

5.8 Numerical Example 3 (8 machines-8 parts)

In this example, the manufacturing system is composed of 8 machines and performs 8 jobs. The alternative routes for operations and processing times are given in Table5.16.

Table 5.16 Alternative process plans and processing times

Job	Operation	Alternative process plan	Processing time
JOB1	A1	1, 2, 8	Unif(6,8)
	A2	3,	Unif(8,10)
	A3	1, 7	Unif(7,9)
JOB2	B1	3, 4	Unif(5,8)
	B2	1, 3, 5	Unif(4,7)
JOB3	C1	3, 4	Unif(8,10)
	C2	1, 6	Unif(9,10)
JOB4	D1	6, 8	Unif(6,7)
	D2	4, 5	Unif(4,7)
	D3	4	Unif(5,8)
JOB5	E1	7, 8	Unif(7,8)
	E2	6, 7	Unif(4,6)
JOB6	F1	1, 4, 8	Unif(9,11)
	F2	5, 7	Unif(8,10)
	F3	3, 6	Unif(9,10)
JOB7	G1	1, 2	Unif(7,10)
	G2	2, 3	Unif(6,9)
	G3	4	Unif(5,8)
JOB8	H1	5	Unif(5,7)
	H2	5, 6	Unif(7,8)
	H3	3, 6	Unif(8,11)
	H4	2, 5	Unif(9,10)

Note that in this example, the number of sub-operations varies in the range of 2-4 for each job. Hence this example differs form the previous one by relaxing the assumption of “equal number of sub-operations”. Processing times are uniformly distributed, part arrivals are exponentially distributed with a mean of 7 min.

The minimum and maximum numbers of machines that can be assigned to a manufacturing cell are 2 and 3 respectively. Inter-cell part transfer times are exponentially distributed with a mean of 2 and the intra-cell transfer times are negligible. The tolerance values (min-max. limits) of goals are given in Table 5.17.

Table 5.17 Aspiration levels for goals

Goal	Min-Max limits	
Min. Inter-cell movements	2	6
Max. System utilization	0.30	0.60
Min. Time spent in the system	30	50

The proposed methodology is applied to the above case. The best solution is found at 121st iteration. The TS and simulation parameters and solution vector is given in Table 5.18.

Table 5.18 TS and simulation parameters and the best solution obtained.

Parameter set (TS)	Tabu list size(T)=7; neighborhood size(N)=5; number of iterations(iter) = 300
Parameter set (Simulation)	Replication length:100000 min.; Number of replications:5: Warm up period: 10000 min.
CPU time	315 sec.
Solution	X_{1112}, X_{1213}, X_{1311}, X_{2112}, X_{2225}, X_{3114}, X_{3211}, X_{4126}, X_{4214} , X_{4314}, X_{5128}, X_{5227}, X_{6128}, X_{6225}, X_{6326}, X_{7112}, X_{7212}, X_{7314}, X_{8125}, X_{8225},X_{8326},X_{8425}= 1 $g_1 = 3$; $g_2 = 0.4689$; $g_3 = 34.2260$ $\mu_1= 0.7500$, $\mu_2= 0.5630$, $\mu_3= 0.7887$ and $\lambda= 0.5630$

Using the max-min method, the best λ value is found as 0.5630. The solution is summarized in Table 5.19. According to the solution vector, 2 cells are formed. Machines 1-2-3-4 constitute Cell1 and the second cell is composed of machines 5-6-7-8. According to the solution, there are 3 inter-cell movements and the satisfaction level of the first goal μ_1 = 0.7500. The system utilization level is found as 0.4689 (μ_2=0.5630). The average time spent in the system is found as 34.2260 min (μ_3=0.7887). So the overall satisfaction level is λ=0.5630.

Table 5.19 Machine Cells formation and part assignments according to the best solution (max-min method)

CELL 1		CELL 2	
Machines	Operations	Machines	Operations
1	C2	5	B2, F2, H1, H2, H4
2	A1, B1, G1	6	D1, F3, H3
3	A2, C1, G2	7	A3, E2
4	D2, D3, G3	8	E1, F1

Using simple additive method the best solution is found at 220th iteration. The best $\Sigma\mu$ value is found as 2.3970. The solution is summarized in Table 5.20. According to the bet solution obtained by using additive method, there are 2 inter-cell movement and the satisfaction level of the first goal μ_1 = 1. The system utilization level is found as 0.4507 and μ_2=0.5025. The average time spent in the system is found as 32.9100 min and μ_3=0.8546. In additive method, the satisfaction levels of firs and third goal are better than max-min method. The satisfaction level of the second goal which is hard to achieve is higher in max-min method.

Table 5.20 Machine Cells formation and part assignments according to the best solution (additive method)

CELL 1		CELL 2	
Machines	Operations	Machines	Operations
1	A3, C2	5	B2, F2, H1, H2, H4
2	A1, B1, G1, G2	6	D1, F3, H3
3	A2	7	E2
4	C1, D2, D3, G3	8	E1, F1

5.9. Numerical Example 4 (model with tardiness objectives)

This numerical example is taken from Eski and Özkarahan (2007). In previous examples, the base model in which the number of exceptional elements, system utilization and average time spent in the system are considered as goals was used. In this example, the tardiness based objectives as in the model proposed in chapter 3 will be considered for solution. The goals considered in this model are:

$$Goal\_1: \sum_i \sum_o \sum_c D_{1ioc} \prec goal_{int} \quad 5.4$$

$$Goal\_2: system \quad utilization \quad \succ goal_{util} \quad 5.5$$

$$Goal\_3: mean\ tardiness \prec goal_{tardiness} \quad 5.6$$

$$Goal\_4: percentage\ of\ tardy\ jobs \prec goal_{tardyjobs} \quad 5.7$$

The number of exceptional elements (eq. 5.4), mean tardiness (eq. 5.6) and the percentage (eq. 5.7) of tardy jobs will be minimized whereas system utilization (eq. 5.5) will be maximized in minimum and maximum limits. The first objective is determined by an analytical equation whereas other objectives are determined by simulation model. The number of exceptional elements should be substantially smaller than $goal_{excpt}$, system utilization should be substantially greater than $goal_{util}$, the mean tardiness should be substantially smaller than $goal_{tardiness}$ and the percentage of tardy jobs should be substantially smaller than $goal_{tardyjobs}$.

Table5. 21 Alternative routes (process plans) and Processing times of operations.

Job	Operation	Alternative process plan	Processing time (min.)
	A1	1, 5, 6	Unif(6,7)
JOB1	A2	3, 4	Unif(5,8)
	A3	1, 5	Unif(4,7)
	B1	5, 6	Unif(5,6)
JOB2	B2	1, 2, 3	Unif(5,6)
	B3	5	Unif(6,7)
	C1	4, 5	Unif(5,8)
JOB3	C2	1, 4	Unif(3,4)
	C3	1, 2, 5	Unif(5,7)
	D1	2	Unif(7,8)
JOB4	D2	2, 3	Unif(5,6)
	D3	3	Unif(6,7)
	E1	1, 2	Unif(5,7)
JOB5	E2	3	Unif(7,9)
	E3	1, 4	Unif(6,8)
	F1	4, 6	Unif(7,8)
JOB 6	F2	6	Unif(4,5)
	F3	3	Unif(4,6)

The manufacturing system under consideration consists of 6 machines and performs 6 different jobs. Each job consists of 3 sub-operations and can have alternative process routes. The alternative routes for operations and processing times are given in Table 5.21. Processing times are uniformly distributed, part arrivals are exponentially distributed with a mean of 5 min. The minimum and maximum numbers of machines that can be assigned to a manufacturing cell are 2 and 3 respectively. Inter-cell part transfer times are exponentially distributed with a mean of 2 and the intra-cell transfer times are negligible. The tolerance values (min-max. limits) of goals are given in Table 5.22.

Table 5.22 The tolerance values of goals

Goal	Min-Max limits	
Inter-cell movements	2	5
System utilization	0.30	0.75
Mean tardiness	0	7
Percentage of tardy jobs	10	30

5.9.1 Due Date Assignment

For tardiness objectives, it is needed to assign due dates of parts. The type of due date assignment that allows the producer the freedom to set due dates are known as endogenous due date assignment. Sabuncuoğlu and Hommertzheim [10] found Total work content (TWK) rule (Blackstone et al., 1982) effective and it has been widely used in job shop studies. In these experiments, TWK rule is used to set part due dates using the following definition:

$$D = TNOW + k.P \qquad \textbf{(5.8)}$$

Where, D is the due date of job, $TNOW$ is the release time of the job, P is the total processing time of the job and k is the parameter specified by the management (k≥1). In this study, parameter k is taken as 3 (i.e. due date of a job is three times greater than its total processing time).

5.9.2 Solution

The warm-up period is determined as 5.000 min. and the replication length is chosen as 50.000min.The number of independent replications is chosen as 5 for each alternative. The parameter set of tabu search algorithm is chosen by trial and error. Tabu list size and neighborhood size are chosen as 8 and 5 respectively. Maximum number of iterations is chosen as 300.

The simulation model is extended in order to reflect due date assignment procedure and the objectives of mean tardiness and percentage of tardy jobs. The Model and experiment frames of the extended simulation model are given in Appendix A3. The C-program is also modified in order to employ new objectives (number of tardy jobs and percentage of tardy jobs).

First the TS based solution methodology is applied to case using max-min method. The best solution is found in 183rd iteration. The solution is summarized in Table 5.23. According to the solution using max-min method two cells formed. The first machine cell consists of machines 1-4-6 and the second cell composed of machines 2-3-5.The best λ value is found as 0.6517(μ1=1; μ2=0.6517; μ3= 0.6611; μ4= 0.8325).

Table 5.23 Machine Cells formation and part assignments according to the solution (max-min method)

CELL 1		CELL 2	
Machines	Operations	Machines	Operations
1	A3, C2, C3	2	D1, D2, E1,E3
4	A2, C1, F1	3	B2, D3,E2, F3
6	A1, F2	5	B1, B3

Then the proposed methodology is applied to the above case using simple additive method. The best solution is found at 233rd iteration. The best $\Sigma\mu$ value is found as 3.28. The solution is summarized in Table 5.24. According to the solution vector, 2 cells are formed. The first cell is composed of machines 1-4-5 and the second cell is composed of machines 2-3 and 6. There are 2 inter-cell movement and the satisfaction level of the first goal μ_1 = 1. The system utilization level is found as

0.5820 and μ_2=0.6267. The mean tardiness is found as 1.9541 min. and μ_3=0.7208. The percentage of tardy jobs is found as 11.35% and μ_4=0.9325.

Table 5.24 Machine Cells formation and part assignments according to the solution (Simple additive method).

CELL 1		CELL 2	
Machines	Operations	Machines	Operations
1	A3, B2, C3	2	D1, D2, E1
4	A2, C2, E3	3	D3, E2, F3
5	A1, B3, C1	6	B1, F1, F2

Based on the solution obtained by simple additive method, the achievement degrees of the second goal (max. of system utilization) is small (0.6267) because it is difficult to achieve. However the achievement levels of other goals are between 0.7208 and 1. According to the solution obtained by max-min method, the achievement level of goal_2 (0.6517) is higher than simple additive method. However the achievement levels of goal_3 and goal_4 are lessen.

The proposed methodology is applied to the same case using the preemptive method (priority level 1: goal_2; Priority level 2: goal_3; priority level 3: goal_4; priority level 5: goal_1). The best solution obtained by preemptive method is given in Table 5.25.

Table 5.25 Machine Cells formation and part assignments according to the solution (Preemptive method).

CELL 1		CELL 2	
Machines	Operations	Machines	Operations
1	B2, E1	2	D1, D2
5	A3, B3, C1	3	C2, D3, E2, F3
6	A1, B1, F2	4	A2, C3,E3, F1

According to the solution obtained by preemptive method, 2 cells formed. The first cell is composed of machines 1-5-6 and the second cell is composed of machines 2-3 and 4. There are 5 inter-cell movements and the satisfaction level of the first goal μ_1 = 0. The system utilization level is found as 0.6034 and μ_2=0.674. The mean tardiness is found as 2.544 min. and μ_3=0.6365. The percentage of tardy jobs is found as 13.43% and μ_4=0.8295. As stated in Section 4.2, the preemptive method aims to maximize the satisfaction level of the highest priority goal. Therefore the results of the preemptive method are different form other two methods given above and depend on the priority levels. As can be seen from the results, the achievement level of the second goal (system utilization) which has the highest priority is higher than max-min method and additive method. However the achievement levels of other goals are lessen.

It is obvious that a decision maker can find different cell configurations by using different tolerance limit sets. The changes in the part arrival rates or part processing times or *k* parameter also lead different part assignments and cell configurations. Since the proposed model is based on a parametric simulation model, the system can be easily modified for different production requirements. For example when k parameter in TWK rule is taken as 2 instead of 3 the manufacturing cells would form as in table 5.26. In this case, $\Sigma\mu$ value is found as 2.6854 ($\mu 1= 1$; $\mu 2=0.6502$; $\mu 3= 0.4408$; $\mu 4= 0.5944$). It is obvious that, the achievement levels of mean tardiness and percentage of tardy jobs are decreased when tight due dates are used.

Table 5. 26 Machine Cells formation and part assignments (k = 2)

CELL 1		CELL 2	
Machines	Operations	Machines	Operations
1	A1, C3	2	D1, D2, E1
4	A2, C2,E3	3	B2, D3, E2,F3
5	A3, B3, C1	6	B1,F1, F2

5.10 Numerical Example 5 (6 machines-12 parts)

This example is taken from Vin and De Lit (2005) and considers 12 parts and 6 machines. Some data about processing times has been modified in order to get all information needed by the solution approach. The data used in this example is presented in Table 5.27. In previous examples, we assumed that the processing times of alternative machines are same (identical machines). As can be seen from the table, in this example, the processing times of alternative machines are different. The number of operations is also different for each job. The number of operations varies between 1and 3 for each operation.

Table 5.27 Alternative process plans and processing times

Op.	Alternative machines & op.times			Op.	Alternative machines & op.times		
O_{1-1}	**m1**:unif(6,8)	**m3**:unif(5,8)		O_{7-1}	**m3**:unif(6,8)	**m4**:unif(8,9)	
O_{1-2}	**m1**:unif(4,7)	**M3**:unif(6,8)		O_{7-2}	**m1**:unif(5,7)	**m3**:unif(5,8)	
O_{1-3}	**m3**:unif(4,6)			O_{7-3}	**m2**:unif(6,8)	**m3**:unif(6,9)	
O_{1-4}	**m4**:unif(3,7)	**m5**:unif(4,7)		O_{8-1}	**m1**:unif(6,9)	**m3**:unif(7,8)	
O_{2-1}	**m4**:unif(3,7)	**m5**:unif(4,7)		O_{8-2}	**m1**:unif(7,8)	**m4**:unif(5,7)	
O_{2-2}	**m1**:unif(4,9)	**m4**:unif(5,8)		O_{8-3}	**m1**:unif(4,8)	**m5**:unif(7,8)	
O_{2-3}	**m5**:unif(3,7)			O_{8-4}	**m2**:unif(6,8)	**m3**:unif(7,9)	
O_{3-1}	**m4**:unif(6,8)	**m5**:unif(5,7)		O_{9-1}	**m2**:unif(7,8)	**m3**:unif(4,9)	
O_{3-2}	**m3**:unif(6,8)	**m4**:unif(5,9)		O_{9-2}	**m3**:unif(7,9)	**m4**:unif(6,7)	
O_{4-1}	**m2**:unif(4,8)	**m6**:unif(7,9)		O_{9-3}	**m1**:unif(4,8)	**m3**:unif(5,6)	
O_{4-2}	**m2**:unif(4,8)	**m4**:unif(5,8)		O_{10-1}	**m4**:unif(5,9)	**m5**:unif(4,8)	
O_{4-3}	**m6**:unif(5,7)			O_{10-2}	**m2**:unif(5,8)	**m5**:unif(6,9)	
O_{4-4}	**m1**:unif(5,7)	**m2**:unif(4,8)	**m6**:unif(4,7)	O_{10-3}	**m2**:unif(7,9)	**m5**:unif(5,8)	
O_{5-1}	**m2**:unif(5,9)	**m3**:unif(6,8)		O_{10-4}	**m5**:unif(5,7)		
O_{5-2}	**m1**:unif(4,6)	**m3**:unif(5,8)		O_{11-1}	**m1**:unif(6,8)	**m3**:unif(5,9)	
O_{5-3}	**m1**:unif(4,6)	**m2**:unif(3,7)		O_{11-2}	**M1**:unif(6,8)	**M3**:unfi(6,7)	
O_{5-4}	**m1**:unif(5,6)	**m3**:unif(4,7)		O_{11-3}	**m1**:unif(4,6)	**m3**:unif(6,8)	**m5**:unif(6,7)
O_{6-1}	**m2**:unif(3,8)	**m6**:unif(5,8)		O_{12-1}	**m3**:unif(4,6)	**m4**:unif(6,8)	**m5**:unif(6,9)
O_{6-2}	**m2**:unif(4,6)	**m6**:unif(5,8)		O_{12-2}	**m4**:unif(5,9)	**m5**:unif(8,9)	
O_{6-3}	**m1**:unif(4,7)	**m3**:unif(3,7)		O_{12-3}	**m4**:unif(6,8)	**m5**:unif(7,9)	
O_{6-4}	**m2**:unif(7,9)	**m3**:unif(5,8)	**m6**:unif(6,9)				

Part arrivals are exponentially distributed with a mean of 7 and the part types are assigned with the probabilities given in Table 5.28.

Table 5.28 Part assignment probabilities

Part Type	Probability	Part type	Probability
Part 1	0.110	Part 7	0.100
Part 2	0.120	Part 8	0.050
Part 3	0.080	Part 9	0.050
Part 4	0.150	Part 10	0.090
Part 5	0.050	Part 11	0.050
Part 6	0.060	Part 12	0.090

In previous examples, the processing times are written in simulation model in advance. In this case, like part routes and cell formation data, processing times should be also updated according to the alternative machine selected. The c-program and the experiment file of the ARENA model are modified according to the new conditions. In ARENA experiment file, the field where the processing times are written in is marked with " * " character. the change function of the c-code is also modified in order to write the processing time of the related machine in EXPRESSIONS element of experiment file.

The objectives considered for solving the problem are: minimizing number of exceptional elements, maximizing system utilization, minimizing mean tardiness and minimizing the percentage of tardy jobs. The maximum and minimum limits of the goals are determined by pilot simulation studies as in Table 5.28.

Table 5.28 Aspiration levels for goals

Goal	Min-Max limits	
Inter-cell movements	2	5
System utilization	0.30	0.75
Mean tardiness	0	30
Percentage of tardy jobs	10	40

The warm-up period is determined as 10.000 min. and the replication length is chosen as 100.000min.The number of independent replications is chosen as 5 for each alternative. The parameter set of tabu search algorithm is chosen by trial and error. Tabu list size and neighborhood size are chosen as 7 and 5 respectively. Maximum number of iterations is chosen as 300. The *k* parameter is chosen as 2 for TWK rule.

Then the proposed methodology is applied to the above case using simple additive method. The best solution is found at 211th iteration. The best $\Sigma\mu$ value is found as 2.5277. The solution is summarized in Table 5.29. According to the solution vector, 3 cells are formed. The first cell is composed of machines 4-5. The second cell is involves machines 2 and 6 and the third cell is composed of machines 1 and 3. According to the solution, there are 3 exceptional elements and the satisfaction level of the first goal μ_1 = 0.6667. The system utilization level is found as 0.5994 and μ_2=0.6654. The mean tardiness is found as 12.910 min. and μ_3=0.5700. The percentage of tardy jobs is found as 21.23% and μ_4=0.6256.

Table 5.29 Part assignment and cell formation according to the best solution

CELL1		CELL2		CELL3	
Mach.	Operations	Mach.	Operations	Mach.	Operations
4	O_{1-4},O_{2-1},O_{2-2},O_{3-1},O_{3-2},O_{7-1},O_{10-1}	2	O_{4-2}, O_{6-1},O_{4-4}	1	O_{1-1}, O_{1-2}, O_{5-2}, O_{8-1}, O_{8-2}, O_{8-3}, O_{9-3}, O_{11-1},O_{11-2},O_{11-3}
5	O_{2-3}, O_{10-2}, O_{10-3}, O_{10-4}, O_{12-1}, O_{12-2}, O_{12-3}	6	O_{4-1}, O_{4-3}, O_{6-2}	3	O_{1-3}, O_{5-1}, O_{7-2}, O_{7-3},O_{8-4},O_{9-1}, O_{9-2}

5.11 Chapter Summary & Conclusions

In this chapter, the solution approach presented in Chapter 4 is extended for solving the proposed hybrid FGP models for cell formation. The applicability of the proposed models and the solution approach were tested on several numerical examples. First, the base model with three fuzzy goals (number of exceptional elements, system utilization and average time spent in the system) is considered in examples in Section 5.6 and Section 5.7 .The assumption of “equal number of operations for each part” is relaxed in the example presented in Section 5.8. Then the model with tardiness objectives (mean tardiness and percentage of tardy jobs) is considered for solving the example in Section 5.9. Finally a numerical example taken from the literature was solved by proposed solution approach in Section 5.10. In this example, the assumptions of “equal number of operations for each part” and “equal processing times for alternative operations” were relaxed.

The results of the numerical examples show that the proposed hybrid FGP models can be effectively solved with TS algorithm. In numerical examples, The Max-min and additive methods were considered in handling fuzzy goals. The results of experiments shown us, the results of the additive method generally dominate the result of max-min method. The superiority of additive method is significant when the goals that are difficult to achieve are considered. In additive method, the achievement levels of some goals will not decreased because of a particular goal that is difficult to achieve.

As stated before, in the hybrid FGP models, the achievement levels of some goals that are hard to represent analytically are obtained from simulation model. This feature decreases the complexity of mathematical models. The goals that are difficult to represent analytically can be easily obtained from simulation model. Moreover, the stochastic nature of the manufacturing system can be also reflected by simulation models. This feature of the hybrid FGP models leads more realistic cell designs. Moreover, since the proposed hybrid model is integrated with a parametric simulation model, the system changes such as changes in demand rate, processing times etc. can be easily reflected

.

CHAPTER SIX

CONCLUSIONS AND FUTURE RESEARCH

6.1 Conclusions

The researches have been extensively conducted in design of manufacturing cells since 1970's. However, there is limited number of publications that have addressed uncertain production requirements. Another important issue which is rarely addressed in designing manufacturing cells is the existence of alternative process plans which is common in real world applications. The primary goal of this research was to develop a design methodology that addresses the uncertain production requirements and the existence of alternative process plans.

The most of the current researches depend on deterministic models. Stochastic nature of the manufacturing systems is generally omitted. However, relaxations in modeling assumptions such as certainty of cost factors, deterministic demand, deterministic processing times etc. effects the implementation of cellular manufacturing systems. Such relaxations lead to cell designs that are far from meeting the requirements of real world applications. Simulation and fuzzy set theory are useful tools in dealing with uncertainty. The stochastic nature of the manufacturing systems is generally reflected by simulation models. Moreover, the performance measures, which are hard to obtain by analytically, can be easily obtained by simulation models. Fuzzy set theory is used for representing probabilistic and linguistic vagueness and uncertainty. By employing fuzzy mathematical programming models, linguistic vagueness in information pertaining to many other design parameters can be modeled and the optimal or near optimal solutions may be obtained by using mathematical programming tools such as linear programming, goal programming etc. Application of fuzzy mathematical programming approaches to cell formation is a relatively new research area.

Considering the gaps in the current cell formation literature as mentioned in Chapter 2, in this research, a hybrid simulation-analytic fuzzy goal programming model was developed in order to achieve the development of the new CM design methodology that addresses the uncertainty issues and routing flexibility. In this model, the achievement levels of some goals which are difficult to represent analytically are obtained by simulation model whereas other goals are calculated analytically. The stochastic nature of the manufacturing system is also reflected by simulation model. Part demand rates, part processing and transfer times are all stochastic. A tabu search based solution approach was used since the classical solution approaches are not appropriate for solving the proposed hybrid simulation-analytic FGP models.

In the first chapter of the study, brief explanations about cellular manufacturing were given.

In the second chapter, the existing cell formation literature was classified and categorized.

General explanations about fuzzy mathematical programming and fuzzy goal programming were given in Chapter 3. The hybrid FGP model was also proposed in the third chapter.

The TS based solution approach was presented in the fourth chapter. The applicability of the TS based solution approach is also tested on several deterministic models from the literature. The results are compared with the results obtained by optimization software LINGO. The results shown us TS based solution approach can be effectively used for solving FGP models.

In the fifth chapter, TS based solution approach was extended to solve hybrid FGP models in which the achievement level of some of the goals are obtained from simulation model. The integration with simulation models were explained and illustrated in detail. Then the proposed hybrid FGP models were solved using TS based solution approach. Then the effectiveness of proposed models and solution approach were tested on the numerical examples with different sizes under different assumptions.

The results of the numerical examples show that the proposed hybrid FGP models can be effectively solved with TS algorithm. The proposed algorithm differs from the existing approaches in three ways:

First the application of FGP to cell formation problem allows for the vague aspirations of a decision maker. The main difficulty related with Goal Programming is the need for the precise aspiration levels. FGP models have the advantage of allowing vague aspirations of a decision maker and provide solutions considering uncertainty in target levels of design objectives.

Second, the proposed hybrid FGP models consider the stochastic nature of manufacturing system under consideration. The stochastic nature of the manufacturing system is reflected by simulation models. Demand patterns, process times and part transfer times are stochastic. Most of the current approaches in cell formation based on the relaxations in modeling assumptions such as deterministic product demand, deterministic processing times etc. However, real manufacturing systems tend to have uncertainty and vagueness in design parameters. Hence the issue of uncertainty is important and has to be considered in design process of manufacturing cell. High realism can be achieved by this way. The proposed methodology handles the issue of uncertainty in cell formation process by integrating two powerful tools as Fuzzy set theory and Simulation.

Third, the routing flexibility which is an important feature for manufacturing cells when flexible machines are utilized. In the presence of routing flexibility, parts can have different process plans. This feature can improve the groupability of parts. The ignorance of routing flexibility may lead increases in operation costs and additional investment in machines. Covering the issue of routing flexibility also increases the way of forming manufacturing cells. This also increases the complexity of the problem. The proposed methodology covers the issue of routing flexibility in cell formation process.

In conclusion, proposed methodology can be used for solving cell formation problems considering the uncertainty in design parameters (stochastic, demand, part processing times and part transfer times) and vagueness in target levels of design objectives. The issue of routing flexibility is also considered in proposed methodology.

In numerical examples presented in section 5.6 and 5.8, the effectiveness of the proposed algorithm is tested with different sized problems. Although the CPU time is increased when the problem size gets larger, proposed algorithm provides good solutions considering the stochastic nature of manufacturing systems and the routing flexibility. Since the proposed model is a hybrid analytic-simulation model, any objectives that are hard to represent analytically can be evaluated by simulation model. In Section 5.9 and 5.10 the objectives such as mean tardiness and percentage of tardy jobs are considered. These objectives have not been considered in most of the current CF literature probably due to the complex nature of cell formation problems. The proposed methodology allows us to consider such objectives in cell formation process.

Since the proposed model is integrated with a parametric simulation model, the system can be easily adapted for different production requirements. Part demand rates, part processing times, part routes etc. can be easily changed from parametric simulation model and then, corresponding cell formations can be obtained.

In numerical examples given in Chapter 5, the Max-min and additive methods were considered in handling fuzzy goals. The results of experiments showed us, the results of the additive method generally dominate the result of max-min method. The superiority of additive method is significant when the goals that are difficult to achieve are considered. In additive method, the achievement levels of some goals will not decreased because of a particular goal that is difficult to achieve.

The proposed methodology has also some shortcomings. The pilot simulation runs are required to determine the upper and the lower limits of fuzzy goals. Determination of some simulation parameters such as the length of *warm-up period* also requires some pilot simulation runs.

Determination of TS parameters is another difficulty related with solution approach. There is no certain rule for determining TS parameters such as tabu list size, neighborhood size, number of iterations etc. A trial and error process is also required for determining these parameters.

6.2 Contributions

The original contributions of this research can be summarized as follows:

1) A hybrid analytic-simulation fuzzy goal programming model that considers the uncertain production requirements and the existence of alternative process plans was developed.

2) The performance based objectives such as system utilization, time spent in the system, mean tardiness and percentage of tardy jobs, which are not considered by the exiting studies in cell formation are considered.

3) The TS based solution approach was used for solving hybrid analytic-simulation FGP models.

6.3 Future Research

Although the several beneficial conclusions and observations were made according to experimental study, the research can be improved as follows:

1) Since the classical solution approaches such as simplex based methods are not appropriate, a TS based solution approach was used to solve hybrid analytic simulation FGP models. TS algorithm works with more than one solution at a time and the previous studies concluded that this feature of TS is important in dealing with multiple objectives. However, other meta-heuristics such as simulated annealing, genetic algorithms etc. can be also used for solving hybrid FGP models. The effectiveness of the algorithms should be tested and compared.

2) Since the hybrid model is integrated with simulation model, the computation time is increased when the complexity of the simulation models are increased. AI based techniques such as neural networks which are trained by simulation results can be employed instead of simulation model. Hence the performance measures can be obtained by neural networks in a reasonable computation time.

3) Machine breakdowns, part transfer systems (AGV, conveyor etc.), worker skills and movements which are hard to cover with mathematical programming models can be included in the hybrid FGP model. Hence, more realistic cell designs can be obtained.

4) Reconfiguration issues are not addressed in this research. The model should be extended to make multi period cell formation decisions under changing circumstances.

REFERENCES

Agarwal, A. and Sarkis, J. (1988).A Review and Analysis of Comparative Performance Studies on Functional and Cellular Manufacturing Layouts. Computers and Industrial Engineering, 34(1), 77-89.

Arıkan, F., & Güngör, Z. (2001). An Application of fuzzy goal programming to a multiobjective project network problem. *Fuzzy Sets and Systems*, 119, 49-58.

Askin, R.,& Standridge, C.(1993) Modeling and Analysis of Manufacturing Systems. New York: John Wiley & Sons.

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(1), 281-309.

Ballakur, A. (1985). An Investigation of Part Family/Machine Group Formation in Designing a Cellular Manufacturing System. PhD thesis, University of Wisconsin, Madison, WI.

Baykasoglu, A.(1999).Solution of Goal Programming Models Using a Basic Taboo Search Algorithm. *Journal of the Operational Research Society*, 50, 960-973.

Baykasoglu, A., & Gindy, N., Z. (2000). MOCACEF 1.0: multiple objective capability based approach to form part-machine groups for cellular manufacturing applications. International Journal of Production Research, 38(5), 1133-1161.

Baykasoglu, A. (2001). Goal programming using multiple objective tabu search. *Journal of the Operational Research Society*, 52, 1359-1369.

Baykasoglu, A., & Gokcen, T.(2006) A Tabu Search Approach to Fuzzy Goal Programs and an Application to Aggregate Production Planning, *Engineering Optimization*, 31, 155-177.

Bellman, R. E., & Zadeh, L. A. (1970). Decision-making in a fuzzy environment. *Management Science*, 17, 141-164.

Ben-Arieh, D.,& Trianthaphyllou, E. (1992) Quantifying data for group technology with weighted fuzzy features. International Journal of Production Research, 30, 1285- 1299.

Black, J.T. (1983). Cellular manufacturing systems reduce setup time, make small lot production economical. Industrial Engineering, 29(10), 36-48.

Boctor, F., F. (1996). The minimum cost, machine-cell formation problem. Ibid, 34, 1045-1063.

Cao, D., & Chen, M. (2005). A robust cell formation approach for varying product demands. International Journal of Production Research, 43(8), 1587–1605.

Car, Z., & Mikac, T. (2006). Evolutionary approach for solving cell-formation problem in cell manufacturing. Advanced Engineering Informatics, 20, 227-232.

Chan, H.M., & Milner, D.A. (1982). Direct clustering algorithm for group formation in cellular manufacture, Journal of Manufacturing systems, 12(5), 428-439.

Chen, L.H., & Tsai, F.C., (2001). Fuzzy Goal Programming with Different Importance and Priorities. *European Journal of Operation Research*, 133, 548-556.

Chu, C. H. (1989). Cluster analysisin manufacturing cellular formation. OMEGA: International Journal of Management Science, 17, 289-295.

Chu, C. H., & Hayya, J. C., (1991). A fuzzy clustering approach to manufacturing cell formation. *International Journal of Production Research,* 29, 1475-1487.

Djassemi, M. (2005). A simulation analysis of factors influencing the flexibility of cellular manufacturing. *International Journal of Production Research,* 43(10), 2101–2111.

Durmusoglu. M.B. (1993). Analysis of the conversion from a job shop system to a cellular manufacturing system. *International Journal of Production Economics*, 30/31, 427-436.

Eski, O., & Ozkarahan, I. (2007).Design of Manufacturing Cells for Uncertain Production Requirements with Presence of Routing Flexibility. *Lecture Notes in Artificial Intelligence*, 4682, 670-681.

Gen, M., Ida, K., Tsujimura, Y., & Kim, C. E. (1993).Large Scale 0-1 Fuzzy Goal Programming and Its Application to Reliability Optimization Problem. *Computers Ind. Engineering*, 24, 539-549.

Gill, A., & Bector, C. R.(1997) A fuzzy linguistic approach to data quantification and construction of distance measures for part family formation problem. *International Journal of Production Research*, 35, 2565-2578.

Gindy, N. N., Ratchev, T.M., & Case, K. (1995). Component grouping for GT applications - a fuzzy clustering approach with validity measure. *International Journal of roduction Research,* 33, 2493-2509.

Glover, F., (1989).Tabu search: Part I, *ORSA J. Comput.*, 1, 190–206.

Glover, F., (1990). Tabu search: Part II, *ORSA J. Comput*., 2, 4–32.

Greene, T., & Sadowski, R. (1984).A Review of Cellular Manufacturing Assumptions, Advantages, and Design Techniques. *Journal of Operations Management*, 4(2), 85-97.

Gupta, R.M., & Tompkins, J.A.(1982) An examination of dynamic behavior of part-families group technology. *International Journal of Production Research*, 20, 73-86.

Gupta, Y., P., Gupta, M., Kumar, A., & Sundram, C. (1995). Minimizing total intercell and intracell moves in cellular manufacturing: A genetic algorithm approach. *International Journal of Computer Integrated Manufacturing*, 8, 92-101.

Hannan E. L. (1981). Some further comments on fuzzy priorities. *Decision Scienc*e, 13, 337-339.

Ho, Y., C., & Moodie, C., L.(1996). Solving cell formation problem in a manufacturing environment with flexible processing and routing capabilities. *International Journal of production Research*, 34, 2901-2923.

Howard, M., & Newman, R.(1993). From Job Shop to Just-in-Time-A Successful Conversion. *Production and Inventory Management Journal*, 34(3),70-74.

Hu, L., & Yasuda, K. (2006). Minimising material handling cost in cell formation with alternative processing routes by grouping genetic algorithm. *International Journal of Production Research*, 44(11), 2133-2167.

Inuiguchi, M., & Ramik, J. (2000). Possibilistic linear programming: A brief review of fuzzy mathematical programming and a comparison with stochastic programming in portfolio selection problem. *Fuzzy Sets and Systems*, 111 (1), 3-28.

Jayaswal, S., & Adil, G. K. (2004). Efficient algorithm for cell formation with sequence data, machine replications and alternative process routings. *International Journal of Production Research*, 42(12), 2419-2433.

Joines, J., Culbreth, C., & King, R. A.(1996). Comprehensive Review of Production Oriented Cell Formation Techniques. *International Journal of Factory Automation andInformation Management*, 3(4),225-265.

Josien, K., & Liao, T., W. (2000). Integrated use of fuzzy c-means and fuzzy KNN for GT part family and machine cell formation. *International Journal of Production Research*, 38(15), 3513-3536.

Kamrani, A., Parsaie, H., & Chaudhry, M.(1993). A Survey of Design Methods for Manufacturing Cells. *Computers and Industrial Engineering*, 25(1-4), 487-490.

Kamrani, A., Hubbard, K., Parseai, H., Leep, H.R. (1998). Simulation based methodology for machine cell design. *Computers & Industrial Eng*., 34(1), 173-188.

Kannan, V.R., & Ghosh, S. (1996) A virtual cellular manufacturing approach to batch production.*Decision Science*, 27, 519–539.

Khoo, L. P., Lee, S. G., & Yin, X. F. (2003). Multiple-objective optimization of machine cell layout using genetic algorithms. *International Journal of Computer Integrated Manufacturing*, 16(2), 145-155.

Kim, C. O., Baek, J. G., & Baek, J. K. (2004) A two-phase heuristic algorithm for cell formation problems considering alternative part routes and machine sequences. *International Journal of Production Research*, 42(18),3911-3927.

King, J. R. (1980). Machine- component group formation in group technology, *OMEGA*, 8(2), 193-199.

Lee, S., D., & Chen, Y., L. (1997). A weighted approach for cellular manufacturing design: minimizing intercell movement and balancing workload among duplicated machines. *International Journal of Production Research*, 35, 1125-1146.

Lei, D., & Wu, Z. (2006). Tabu search for multiple-criteria manufacturing cell design., *International Journal of Advanced Manufacturing Technology,* 28, 950-956.

Li, J., Chu, C. H., Wang, Y., & Yan, W. (2007). An improved fuzzy clustering method for cellular manufacturing. *International Journal of Production Research,* 45 (5) 1049–1062.

Liang, M., &Taboun, S., M.(1995). Converting functional manufacturing systems into focused machine cells: a bi-criterion approach. *International Journal of Prouction Research*, 33, 2147-2161.

Logendran, R. (1993). A binary integer programming approach for simultaneous machine-part grouping in cellular manufacturing systems. *Computers and Industrial engineering*, 24, 329-336.

Logendran, R., & Talkington, D. (1997). Analysis of cellular and functional manufacturing systems in the presence of machine breakdown. International *Journal of Production Economics,* 53, 239-256.

Lozano, S., Guerrero, F., Eguia, I., & Onieva, L. (1999). Cell design and loading in the presence of alternative routing. *International Journal of Production Research*,37 (14), 3289-3304.

Lozano, S., Dobado, D., Larraneta, J., & Onieva, L. (2002). Modified fuzzy C means algorithm for cellular manufacturing, *Fuzzy Sets and Systems*, 126, 23-32.

Mahesh, O., & Srinivasan, G. (2006). Multi-objectives for incremental cell formation problem. *Ann. Operation Res*.,143, 157–170.

Mahmoodi, F., Dooley, K.J., & Starr, P.J.(1990). An evaluation of order releasing and due date assignment heuristics in a cellular manufacturing system. *J. Op. Management*, 9, 548–572.

Mansouri, S., A., Mottar Husseini, S., M., & Newman, S., T. (2000). A review of modern approaches to multi-criteria cell design. *International Journal of Production Research*, 38 (5), 1201-1218.

Mc Auley, J. (1972). Machine grouping for efficient production. *Production Engineer*, 5(12), 53-57.

Mc Cormik, W. T., Schweitzer, P. J., & White, T. W. (1972). Problem decomposition and data reorganization by a clustering technique. *Operations Research*, 20(5), 993-1009.

Mehrabad, M., & Safaei, N. (2005). A new model of dynamic cell formation by a neural approach. *International Journal of Advanced Manufacturing Technology*, in press.

Min, H., & shin, D. (1993). Simultaneous formation of machine and human cells in group technology: a multiple objective approach. *International Journal of Production Research*, 31, 2307-2318.

Mohamed, R. H. (1997). The relationship between goal programming and fuzzy programming. *Fuzzy Sets and Systems*, 89, 215-222.

Morris. J.S., & Tersine, R.J. (1990) A simulation analysis of factors influencing the attractiveness of group technology cellular layouts. *Management Science*, 36(12), 1567-1578.

Mosier, C.T. (1989). An expeirment investigating the application of clustering procedures an similarity coefficients to the GT machine cell formation problem. *International Journal of Production Research,* 27(10), 1811-1835.

Mungwattana, A. (2000). *Design of Cellular Manufacturing Systems for Dynamic and Uncertain Production Requirements with Presence of Routing Flexibility*, PhD thesis, Blacksburg, Virginia.

Narasimhan, R. (1980). Goal programming in a fuzzy environment. *Decision Science*, 11, 325-336.

Offodile, O., Mehrez, A., & Grznar, J.(1994) Cellular Manufacturing: A Taxonomic ReviewFramework. *Journal of Manufacturing Systems*, 13(3):196-220.

Onwubolu, G. C., & Songore, V. (2000). A tabu search approach to cellular manufacturing Systems.*Production Planning & Control*, 11(2), 153-164.

Pegden C.D., Shannon R.E, & Sadowski R. P. (1990), *Introduction to simulation using SIMAN*. United States. Mc Graw Hill.

Pierreval, H., Caux, C., Paris, J. L., & Viguier, H. (2003).Evolutionary approaches to the design and organization of manufacturing systems. *Computers & industrial engineering*, 44(3), 339-364.

Rajamani, D., Singh, N., & Aneja, Y., P. (1996). Design of cellular manufacturing systems. *International Journal of Production Research,* 34, 1917-1928.

Saad, S. M., Baykasoglu, A., & Gindy, N. Z. (2002). An integrated framework for reconfiguration of cellular manufacturing systems using virtual cells. *Production Planning & Control*, 13(4), 381-393.

Sabuncuoglu, I., & Hommertzheim, D. L. (1989). Dynamic Dispatching Algorithm for Scheduling Machines and Automated Guided Vehicles in a Flexible Manufacturing System. *International Journal of Production Research*, 30, 1059-1079.

Sakawa, M. (1993). *Fuzzy sets and interactive multi-objective optimization*. NY: Plenum Press.

Sankaran, S. (1990). Multiple objective decision making approach to cell formation: A goal programing model. *Mathematical and Computer Modeling*, 13, 71-81.

Schaller, J (2005). Tabu search procedures for the cell formation problem with intra-cell transfer costs as a function of cell size. *Computers & Industrial Engineering*, 49, 449– 462.

Seifoddini, H. (1990), A probabilistic model for machine cell formation. *Journal of Manufacturing Systems*, 9, 69-75.

Seifoddini, H.(1986). Improper machine assignment in machine component grouping in group technology, *IIE transactions*, 21(4), 382-388.

Selim, H., Askin, R., & Vakharia, A. (1998). Cell Formation in Group Technology: Review, Evaluation and Directions for Future Research. *Computers and Industrial Engineering*, 34(1):3-20.

Sethi, A., & Sethi, S. (1990).Flexibility in Manufacturing: A Survey. International *Journal of Flexible Manufacturing Systems*, 2, 289-328.

Shafer, S., M., & Rogers, D., F. (1991). A goal programming approach to cell formation problem. *Journal of Operations Management*, 10, 28-43.

Shanker, R., & Vrat, P. (1996). Design of cellular manufacturing system. A chance constrained programming approach. Mubeen, S., & Rizvi, A.H. (Ed.) *CAD / CAM, Automation, Robotics and Factories of Future* , (110-116), New Delhi Narosa Publishing House.

Shanker, R., & Vrat, P. (1999). Some design issues in cellular manufacturing using the fuzzy programming approach. *International Journal of Production Research*, 37(11), 2545-2563.

Solimanpur, M., Vrat, P., & Shankar, V. (2004). A multi-objective genetic algorithm approach to the design of cellular manufacturing systems. International Journal of Production Research, 42(7), 1419-1441.

Su, C., T., & Hsu, C., M. (1998). Multi-objective machine-part cell formation through parallel simulated annealing. *International Journal of Production Research*, 36, 2185-2207.

Suresh, N., (1992). Partitioning work centers for group technology: Analytical extension and shop-level simulation investigation. *Decision Sciences*, 23 (2), 267–289.

Suresh, N., C., Slomp, J., & Kaparthi, S. (1995). The capacitated cell formation problem: a new hierarchical methodology. *Internationl Journal of Production Research,* 32, 1693-1713.

Szwarc, D., Rajamani, D., & Bectort, C. R. (1997). Cell Formation Considering Fuzzy Demand and Machine Capacity. *International Journal of Advanced Manufacturing Technology*, 13, 134-147.

Suresh, N. C., & Slomp, J. (2001). A multi-objective procedure for labour assignments and grouping in capacitated cell formation problems. *International Journal of Production Research*, 39(18), 4103-4131.

Tiwari, R. N., Dharmar, S., & Rao, J. R. (1986). Priority structure in fuzzy goal programming. *Fuzzy Sets and Systems*, 19, 251-259.

Tiwari, R.N., Dharmar, S., & Rao, J.R. (1987). Fuzzy goal programming: An additive method. *Fuzzy Sets and Systems*, 24, 27-34.

Tsai, C., Chu, C., & Wu, X. (2006). An Evolutionary Fuzzy Multi-objective Approach to Cell Formation. *Lecture Notes in Artificial Intelligence*, 4247, 377-383.

Vakharia, A., J., & Chang, Y., L.(1997). Cell formation in group technology: A combinatorial search approach. *International Journal of Production Research,* 35(7), 2025-2043.

Venugopal, V., & Narendran, T., T. (1992). A genetic algorithm approach to the machine component grouping problem with multiple objectives. *Computers and Industrial Engineering*, 22, 469-480.

Vin, E., De Lit, P., & Delchambre, A. (2005). A multiple-objective grouping genetic algorithm for the cell formation problem with alternative routings. *Journal of Intelligent Manufacturing,* 16, 189-205.

Wang, H. F., & Fu, C. C. (1997). A generalization of fuzzy goal programming with preemptive structure. *Computers & Operations Research*, 24, 819-828.

Wang, H. F., & Wang, M. L. (1997). A fuzzy multiobjective linear programming. *Fuzzy Sets and Systems*, 86, 61-72.

Wei, J. C., & Gaither, N. (1990). A capacity constrained multiobjective cell formation method. *Journal of Manufacturing Systems*, 9, 222-232.

Wemmerlöv, U., & John, D.(1997).Cellular Manufacturing at 46 User Plants: Implementation experiences and Performance Improvements. *International Journal of Production Research,* 35(1), 29-49.

Wemmerlöv, U, & Hyer, N.L. (1986). Procedures for the part family /machine group identification problem in cellulamanufacturing. *Journal of Operations management*, 6, 125-147.

Xu, H.,& Wang, H., (1989).Part family formation for group technology applications based on Fuzzy mathematics. *International Journal of Production Research,* 27, 1637-1651.

Yang, M. S., Hung, W., & Cheng, F. (2006). Mixed-variable fuzzy clustering approach to part family and machine cell formation for GT applications. *International Journal of Production Economics*, 103, 185-198.

Zadeh, L. A. (1965). Fuzzy sets. *Information and Control*, 8, 338-353.

Zhang, C., Wang, H. P.(1992). Concurrent formation of part family and machine cells based on fuzzy set theory. *Journal of Manufacturing System*, 11, 61- 67.

Zhao, C., & Wu, Z. (2000). A genetic algorithm for manufacturing cell formation with multiple routes and multiple objectives. *International Journal of Production Research*, 38(2), 385-395.

Zimmermann, H. J. (1978). Fuzzy programming and linear programming with several objective function. *Fuzzy Sets and Systems*, 1, 45-55.

APPENDICES

A1. LINGO Model of Numerical Example 4. 3

```
MODEL:
! ;
 SETS:
  JOB/1..3/;

  OPERATION/1..3/;

  CELL/1..2/:Q;

  MACHINE/1..5/:R;

  SET1(CELL,MACHINE):Y,T;
  SET2(JOB,OPERATION,MACHINE):A,P;
  SET3(JOB,OPERATION,CELL):Z,D1,D;
  SET4(JOB,OPERATION,CELL,MACHINE):X;
  SET5(JOB,OPERATION):;

ENDSETS

DATA:
P=1 0 0 1 1
  0 0 1 1 0
  1 0 0 0 1
  0 0 0 1 1
  1 1 1 0 0
  0 0 0 0 1
  0 0 0 1 1
  1 0 0 1 0
  1 1 0 0 1;

MMAX=3;
MMIN=2;
A=5 0 0 4 2 0 0 4 6 0 7 0 0 0 6
  0 0 0 4 7 3 5 6 0 0 0 0 0 0 4
  0 0 0 5 7 4 0 0 4 0 3 2 0 0 5;

ENDDATA
Max=lamda;

lamda<=((3-Intcell)/(3-0));
lamda<=((40-totalt)/(40-32));
lamda<=1;

@SUM(SET3(I,O,C):D1(I,O,C))=Intcell;
@sum(set4(I,O,C,M):A(I,O,M)*X(I,O,C,M))=totalt;
@FOR(SET5(I,O):(@SUM(SET1(C,M):X(I,O,C,M)*P(I,O,M)))=1);
@FOR(MACHINE(M):(@SUM(CELL(C):Y(C,M)))=1);
@FOR(SET4(I,O,C,M):X(I,O,C,M)<=1000*Y(C,M));
@FOR(SET4(I,O,C,M):X(I,O,C,M)<=1000*Z(I,O,C));

@FOR(SET5(I,O):(@SUM(CELL(C):Z(I,O,C)))=1);
```

```
@FOR(SET3(I,O,C)|O#GT#1:(Z(I,O,C)-Z(I,O-1,C))=D1(I,O,C)-D(I,O,C));

@FOR(CELL(C):@SUM(MACHINE(M):Y(C,M))<=MMAX*Q(C));
@FOR(CELL(C):@SUM(MACHINE(M):Y(C,M))>=MMIN*Q(C));

@FOR(SET4(I,O,C,M):@BIN( X(I,O,C,M)));
@FOR(SET1(C,M):@BIN( Y(C,M)));
@FOR(SET3(I,O,C):@BIN(Z(I,O,C)));
@FOR(CELL(C):@BIN( Q(C)));
@BIN(intcell);
END
```

A2. C Codes of Numerical Example 5.5

(The functions *initialX, GenerateX, GenYZDQ, Change, and Processoutfile* are not included.)

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
#include<time.h>
#define REP 5 /* the number of replications */
#define K 99  /*the number of Xiocm variables*/
#define M 7   /* Tabu list size */
#define N 5   /* neighborhood size */
#define INTERVAL 18 /* number of intervals in a seed */
#define MU1LL 0         /* lower limit for goal_1*/
#define MU1UL 3         /* upper limif for goal_1*/
#define MU2LL 40
#define MU2UL 60
#define MU3LL 0.3
#define MU3UL 0.75
#define ITER 300        /*maximum  number  of  iterations
*/
#define MIN(a,b) (((a)<(b))?(a):(b))
FILE *fp,*fp1,*fp2,*fp3;

char mak[6]; /*machine index*/
char  ad[  ]="c:\\endustri\\arena\\output6.txt";  /*output
file*/
char  file1[  ]="c:\\endustri\\arena\\deneme6.exp";  /*the
path of default ARENA EXPERIMENT file*/
char         file2[]="c:\\endustri\\arena\\deneme66.exp";
/*modified ARENA EXPERIMENT FILE*/
char   file3[   ]="c:\\endustri\\arena\\deneme6.txt";   /*
ARENA output document*/
double per[N][2];
char result[18];
double tabul[M][K],neil[M][K];
double lamda[N][K+1];
double tindex,nindex=0;
double dlam[N];
char char Y[18]={0},Z[54]={0},D[36]={0},Q[3]={0};
char X[K]={0};
char XC[K]={0};

char
sinir[INTERVAL][2]={0,8,9,14,15,20,21,26,27,35,36,38,39,4
4,45,50,51,59,60,62,63,68,69,71,72,77,78,80,81,86,87,92,9
3,95,96,K-1};
```

```
void initial(char a[ ],char b[ ])
{
int i;
for(i=0;i<K;i++)
   a[i]=b[i];
}

void initialX(char a[ ]) /* creates an initial solution*/

void generateX(char a[ ]) /*generates neighborhood
solutions from initial seed */

void genYZDQ() /*calculates Y, Z, D and Q variables */

char constraints() /*constraints of the model */
{
char k1,k2,k3,k4,k5,k6,k7,k8,k9,k10,k11,k12;
char kisit1=0,kisit2=0;
char result;
k1=Y[0]+Y[6]+Y[12];
k2=Y[1]+Y[7]+Y[13];
k3=Y[2]+Y[8]+Y[14];
k4=Y[3]+Y[9]+Y[15];
k5=Y[4]+Y[10]+Y[16];
k6=Y[5]+Y[11]+Y[17];
k7=Y[0]+Y[1]+Y[2]+Y[3]+Y[4]+Y[5]-3*Q[0];
k8=Y[6]+Y[7]+Y[8]+Y[9]+Y[10]+Y[11]-3*Q[1];
k9=Y[12]+Y[13]+Y[14]+Y[15]+Y[16]+Y[17]-3*Q[2];
k10=Y[0]+Y[1]+Y[2]+Y[3]+Y[4]+Y[5]-2*Q[0];
k11=Y[6]+Y[7]+Y[8]+Y[9]+Y[10]+Y[11]-2*Q[1];
k12=Y[12]+Y[13]+Y[14]+Y[15]+Y[16]+Y[17]-2*Q[2];
if((k1==1)&&(k2==1)&&(k3==1)&&(k4==1)&&(k5==1)&&(k6==1))
kisit1=1;
if((k7<=0)&&(k8<=0)&&(k9<=0)&&(k10>=0)&&(k11>=0)&&(k12>=0
)) kisit2=1;
result=kisit1*kisit2;
return result;
}

int objective1() /*calculate the objective of number of
exceptional elements by an analytic equation */
{
int sum=0,i;
for(i=0;i<36;i++)
   sum+=D[i];
return sum;
}
```

```
double mu1() /*calculate the the membership function of
the first goal */
{
return ((MU1UL-objective1())/(double)(MU1UL-MU1LL));
}

double mu2(double x) /* calculate the membership function
of the second goal*/
{
return ((MU2UL-x)/(double)(MU2UL-MU2LL));
}

double mu3(double x) /* calculate the membership function
of the third goal*/
{
return ((x-MU3LL)/(double)(MU3UL-MU3LL));
}

double compare(char a[], double c[][K], double p)
{
int i,j,z;
double l=1;
for(i=0;i<p;i++)
   {
   z=0;
   for(j=0;j<K;j++)
      {
      if(a[j]==c[i][j]) z++;
      }
   if(z==K) {l=0;break;}
   else l=1;
   }
return l;
}

double findmax(double a[], int b)
{
     double max1;
     int i;
     max1=a[0];
          for (i=1;i<b;i++)
               if (a[i]>max1) max1=a[i];
                    return max1;
}

void encode1(char c[]) /* part routes are sepecified */
{
char i,l;
for(i=0;i<K;i++)
```

```
 if((i>=0)&&(i<=8))
    {
    if(c[i]==1)
    {
    l=i%3;
    if(l==0) result[0]=1;
    else if(l==1) result[0]=5;
    else if (l==2) result[0]=6;
    else;
    }
    }
 else if((i>=9)&&(i<=14))
    {
    if(c[i]==1)
    {
    l=i%2;
    if(l==0) result[1]=4;
    else if(l==1) result[1]=3;
    else ;
    }
    }
 else if((i>=15)&&(i<=20))
   {
    if(c[i]==1)
    {
    l=i%2;
    if (l==0) result[2]=5;
    else if (l==1) result[2]=1;
    }
    }
 else if((i>=21)&&(i<=26))
{
    if(c[i]==1)
    {
    l=i%2;
    if(l==0) result[3]=6;
    else if(l==1) result[3]=5;
    else ;
    }
    }
   else if((i>=27)&&(i<=35))
    {
    if(c[i]==1)
    {
    l=i%3;
    if(l==0) result[4]=1;
    else if(l==1) result[4]=2;
    else if (l==2) result[4]=3;
    else ;
```

```
       }
       }

      else if((i>=36)&&(i<=38))
      {
       if(c[i]==1)
       {
       result[5]=5;
        }
       }
 else if((i>=39)&&(i<=44))
 {
       if(c[i]==1)
       {
      l=i%2;
      if (l==0)  result[6]=5;
      else if (l==1) result[6]=4;
       }
       }

      else if((i>=45)&&(i<=50))
      {
       if(c[i]==1)
       {
       l=i%2;
       if(l==0) result[7]=4;
       else if(l==1) result[7]=1;
       else ;
       }
       }
      else if((i>=51)&&(i<=59))
      {
       if(c[i]==1)
       {
       l=i%3;
       if(l==0) result[8]=1;
       else if(l==1) result[8]=2;
       else if(l==2) result[8]=5;
       else ;
       }
        }
  else if((i>=60)&&(i<=62))
      {
       if(c[i]==1)
       {
        result[9]=2;
        }
        }
```

```
else if((i>=63)&&(i<=68))
     {
      if(c[i]==1)
      {
      l=i%2;
      if(l==0) result[10]=3;
      else if(l==1) result[10]=2;
      else ;
      }
      }

     else if((i>=69)&&(i<=71))
     {
      if(c[i]==1)
      {
       result[11]=3;

      }
      }
      else if((i>=72)&&(i<=77))
     {
      if(c[i]==1)
      {
      l=i%2;
      if(l==0) result[12]=1;
      else if(l==1) result[12]=2;
      else ;
      }
       }
     else if((i>=78)&&(i<=80))
      {
      if(c[i]==1)
      {
      result[13]=3;
      }
       }

     else if((i>=81)&&(i<=86))
     {
      if(c[i]==1)
      {
      l=i%2;
      if(l==0) result[14]=4;
      else if(l==1) result[14]=1;
      else ;
      }
 }
 else if((i>=87)&&(i<=92))
     {
```

```
      if(c[i]==1)
      {
      l=i%2;
      if(l==0) result[15]=6;
      else if(l==1) result[15]=4;
      else ;
      }
 }
 else if((i>=93)&&(i<=95))
     {
      if(c[i]==1)
      {
      result[16]=6;
      }
 }
else if((i>=96)&&(i<=98))
     {
      if(c[i]==1)
      {
      result[17]=3;
      }
 }

}

void change() /* modify the default experiment file */

void processoutfile(int x) /* obtains simulation based
objectives from the output file of ARENA*/


main()
{
double mini,lam,lamdabest;
int i,j=0,k,z,w,q,k1,k2,k3;
long l;
clrscr();
randomize();
fp=fopen(ad,"w");
do
{
initial(X,XC);
initialX(X);genYZDQ(); /*create an initial solution and
calculate Y, Z, D, Q variables */
}while(constraints()!=1);
initial(XC,X);
for(i=0;i<K;i++)
   tabul[0][i]=XC[i];
tindex=1;
```

```
for(l=1;l<=ITER;l++)
{
j=0;z=0;
while(j<N)
   {
   if (z>1000) break;
   else
      {
      initial(X,XC);
      generateX(X); /*generate neighborhood solution from
initial seed*/
      genYZDQ();    /*calculate Y, Z, D, Q */
      nindex=(int)nindex%N;
      k1=k2=k3=0;
      if(compare(X,neil,nindex)==1)  k1=1;  /*  check  the
neighborhood list */
      if(constraints()==1) k2=1;         /*check
constraints */
      if(compare(X,tabul,tindex))  k3=1;     /*check  tabu
list*/
      if(k1*k2*k3==1)                              /*      the
solution  is  feasible  if  above  conditions  are  met,
otherwise create a new neighbor solution*/
      {
      j++;
      for(i=0;i<K;i++)
         lamda[nindex][i]=neil[nindex][i]=X[i];

encode(X);encode2(Y);change();system("c:\\endustri\\arena
\\tabu4.bat");processoutfile(nindex); /* call functions/
execute   ARENA   model   for   current   solution/obtain
simulaiton based objectives */

      mini=mu1()+mu2(per[nindex][1])+mu3(per[nindex][0]);
/*calculate  sum  of  membership  functions-  for  additive
method */

           lamda[nindex][K]=mini;
      nindex++;
      }
      else z++;
      }
   }
for(i=0;i<N;i++)
   dlam[i]=lamda[i][K];
lam=findmax(dlam,N-1);  /*  find  the  solution  with  the
highest total mu in a neighborhood*/
if(l==1) lamdabest=lam;
```

```
if(lamdabest<lam) lamdabest=lam; /* if the solution is
the current best, keep the solution */
for(i=0;i<N;i++)
   if(lam==lamda[i][K])
      {
      for(j=0;j<K;j++)
      {
      XC[j]=lamda[i][j];
      }
      if(tindex<M)
      {
      for(j=0;j<K;j++)
         tabul[tindex][j]=XC[j]; /* if the tabu list is
not full, put the last move in tabu list */
      tindex++;
      }
      else
      {
      for(k=1;k<M;k++)            /* if the tabu list is
full, replace the current move with the oldest one in
tabu list*/
         for(j=0;j<K;j++)
            tabul[k-1][j]=tabul[k][j];
      for(j=0;j<K;j++)
         tabul[k-1][j]=XC[j];
      }
      break;
      }
if(lam==lamdabest)  /*  keep  the  best  solutions  in
output.txt file */
   {
   printf("%6ld ",l);
   fprintf(fp,"%6ld ",l);
   for(i=0;i<K;i++)
      {
      printf("%d",XC[i]);
      fprintf(fp,"%d,",XC[i]);
      }
    printf(" %11.9lf %11.9lf \n",lam,lamdabest);
    fprintf(fp," %11.9lf %11.9lf \n",lam,lamdabest);
   }
}
fclose(fp);
return 0;
}
```

A3. ARENA Model File of Example 5.9

```
0$              CREATE,         1:expo(9,1):MARK(time);
1$              ASSIGN:
jt=disc(0.1667,1,0.3333,2,0.5,3,0.6667,4,0.8333,5,1,6,2):
                                picture=jt:
                                ns=jt:
                                m=dummy:

tps=processtime(jt,1)+processtime(jt,2)+processtime(jt,3)
:
                                dd=tnow+k*tps;
2$              ROUTE:          0.0,seq;


3$              STATION,        1-6;
17$             ASSIGN:         control=0;
4$              QUEUE,          m;
5$              SEIZE,          1:
                                m,1;
6$              DELAY:          processtime(jt,is);
7$              RELEASE:        m,1;
9$              BRANCH,         1:
                                If,is==3,11$,Yes:
                                Else,15$,Yes;
11$             TALLY:          t1,int(time),1;
18$             BRANCH,         1:
                                If,tnow>dd,19$,Yes:
                                Else,20$,Yes;
19$             COUNT:          c1,1;
20$             COUNT:          c2,1;
21$             ASSIGN:         tar=tar+(tnow>dd)*(tnow-dd):
                                meantar=tar/nc(c2);
10$             DISPOSE;

15$             BRANCH,         1:

If,hucre(m)==hucre(rota(jt,is+1)),16$,Yes:
                                Else,8$,Yes;
16$             ASSIGN:         control=1;
8$              ROUTE:
(control==1)*0+(control==0)*expo(2,3),seq;


12$             CREATE,         1,tfin:,1:MARK(time);
13$             WRITE,          file1,"%7.5f %8.3f %8.0f
%8.2f\n":
```

```
                              davg(13),
                              tavg(1),
                              tar,
                              meantar;
14$           DISPOSE;
```

A.4 Default ARENA Experiment file of example 5.9

```
SEQUENCES:    *

ATTRIBUTES:   1,jt:
              2,time:
              3,control:
              dd:
              tps;

FILES:
1,file1,"c:\endustri\arena\denemed.txt",Sequential(),Free
Format,Error,No,Hold;

VARIABLES:    1,hucre(6),*
              meantar:
              k,1:
              tar;

QUEUES:       6,buffers,FirstInFirstOut;


RESOURCES:    6,machines,Capacity(1,),-,Stationary;

STATIONS:     6,workcenter:
              7,dummy;


COUNTERS:     1,c1,,Replicate:
              2,c2,,Replicate;

TALLIES:      1,t1,"c:\endustri\arena\t1.dat";

DSTATS:       1,nq(1):
              2,nq(2):
              3,nq(3):
              4,nq(4):
              5,nq(5):
```

```
                6,nq(6):
                7,nr(1):
                8,nr(2):
                9,nr(3):
                10,nr(4):
                11,nr(5):
                12,nr(6):
                13,(nr(1)+nr(2)+nr(3)+nr(4)+nr(5)+nr(6))/6;

OUTPUTS:        1,davg(13):
                2,tavg(1):
                meantar:
                tar;

REPLICATE,      5,0.0,22000,Yes,Yes,10000;

EXPRESSIONS:
1,processtime(6,3),unif(11,14),unif(7,10),unif(6,11),unif
(7,10),unif(8,10),unif(8,10),unif(7,9),unif(8,11),

unif(8,11),unif(9,12),unif(7,9),unif(6,10),unif(8,11),uni
f(7,9),unif(7,8),unif(7,10),unif(8,9),unif(9,11):
                35,rota(6,3),*
```