APPLICATION OF SOFT COMPUTING TECHNIQUES FOR CELL FORMATION CONSIDERING OPERATIONAL TIME AND SEQUENCE

A THESIS SUBMITTED IN FULFILLMENT OF THE REQUIREMENT FOR THE AWARD OF THE DEGREE

OF

Doctor of Philosophy

IN

MECHANICAL ENGINEERING

BY

R. SUDHAKARAPANDIAN

(ROLL NO. 50503001)

NATIONAL INSTITUTE OF TECHNOLOGY
ROURKELA - 769008, INDIA
August - 2007

CERTIFICATE

This to certify that the thesis entitled **"APPLICATION OF SOFT COMPUTING TECHNIQUES FOR CELL FORMATION CONSIDERING OPERATIONAL TIME AND SEQUENCE"** being submitted by **R.Sudhakara Pandian** for the award of the degree of Doctor of Philosophy (Mechanical Engineering) of NIT Rourkela, is a record of bonafide research work carried out by him under my supervision and guidance. Mr. R.Sudhakara Pandian has worked for more than two and half years on the above problem at the Department of Mechanical Engineering, National Institute of Technology, Rourkela and this has reached the standard fulfilling the requirements and the regulation relating to the degree. The contents of this thesis, in full or part, have not been submitted to any other university or institution for the award of any degree or diploma.

Rourkela

Date:

Dr. Siba Sankar Mahapatra

Professor
Department of Mechanical Engineering
National Institute of Technology
Rourkela –769008, India
E-mail: ssm@nitrkl.ac.in,
mahapatrass2003@yahoo.com
Phone: 91-661-2462512
Fax: 91-661-247 2926

ACKNOWLEDGEMENT

*The author expresses his deep sense of gratitude and indebtedness to his supervisor **Prof. Siba Sankar Mahapatra**, for his invaluable encouragement, helpful suggestions, moral support and affection. His priceless and meticulous supervision at each and every phase of work inspired me in innumerable ways.*

*His sincere thanks are due to **Prof. Sunil Kumar Sarangi,** Director, and **Prof. B.K.Nanda,** Head of the Department, Mechanical Engineering, N.I.T., Rourkela for granting permission to use the facilities available in the institute for this study. He thanks **Prof. N. Kavi** of Mechanical Engineering Department, the former Head of the Department, Mechanical Engineering for extending infrastructural facilities to carry out this work smoothly. The author expresses his indebtedness to **Prof.S.K.Rath** Dean (Academic Affairs) and **Prof.B.B.Biswal** Training and Placement Officer, NIT Rourkela for their continuous encouragement.*

*The author also expresses his thankfulness to the faculty and staff members of the Mechanical Engineering Department for their continuous encouragement and suggestions. Among them, **Sri Prasanta Kumar Pal** deserves special thanks for his kind cooperation in non-academic matters during the research work.*

*He will carry in his memory the company of Research Scholars **Mr. M.S. Khan**, **Mr. S.R. Mohanty** and **Mr. Amar Patnaik** and all other Research Scholars at NIT Rourkela for exchanging ideas in academic matters, supporting and co-operating in non-academic matters. The author expresses special thanks to **Mr. N. Shesaiah.***

*The author feels pleased and privileged to fulfill his parents ambition and greatly indebted to them for bearing the inconvenience of his stay away from them in time of their need. The author expresses his appreciation to his wife **A.Madhumathi** for her understanding, patience and active cooperation throughout the course of his doctoral dissertation.*

*Finally, the author wishes to acknowledge the financial support given to him by the **TEQIP**, Government of India during his tenure of stay at National Institute of Technology, Rourkela. Last, but not the least, the author thanks all those who have directly or indirectly assisted him for completion of this thesis.*

(R.Sudhakara Pandian)

ABSTRACT

In response to demand in market place, discrete manufacturing firms need to adopt batch type manufacturing for incorporating continuous and rapid changes in manufacturing to gain edge over competitors. In addition, there is an increasing trend toward achieving higher level of integration between design and manufacturing functions in industries to make batch manufacturing more efficient and productive. In batch shop production environment, the cost of manufacturing is inversely proportional to batch size and the batch size determines the productivity. In real time environment, the batch size of the components is often small leading to frequent changeovers, larger machine idleness and so lesser productivity. To alleviate these problems, “Cellular Manufacturing Systems” (CMS) can be implemented to accommodate small batches without loosing much of production run time. Cellular manufacturing is an application of group technology (GT) in which similar parts are identified and grouped together to take advantage of their similarities in design and production. Similar parts are arranged into part families and each part family processes similar design and manufacturing characteristics. Cellular manufacturing is a good example of mixed model production and needs to resolve two tasks while implementing cellular manufacturing. The first task is to identify the part families and the next task is to cluster the production machines into machine cells known as cell formation (CF). GT ideas were first systematically presented by Burbidge following the pioneering work of Mitrofanov in U.S.S.R. Burbidge developed the concept of production flow analysis and successfully implemented in industries. After this, many countries started following GT concepts in their manufacturing lines. Researchers initiated to develop various methods like similarity coefficient method, graph theoretic approaches and array based methods in this field. In this trend, modeling of CMS through mathematical programming was started to incorporate more real life constraints on the problem. Later researchers started developing heuristics and meta-heuristics to explore the best optimal solutions for the CF problems. Since soft computing techniques nowadays expand their applications to various fields like telecommunications, networking, design and

manufacturing, current research in CMS is being carried out using soft computing techniques.

As for as representation of the cell formation problem is concerned, most of the researchers use zero-one binary machine part incidence matrix (MPIM) that is obtained from the route sheet of the manufacturing flow shop. The 1's in the binary matrix represent the visit of the parts to the corresponding machines and 0's represent the non-visit. The final output is a block diagonal structure from which the part families and corresponding machine cells where the part families are to be manufactured can be identified. In such an input representation, the process of clustering machines into machine cells and parts into part families is done without using real life information which may lead to inferior manufacturing plans. Therefore, there is a need to make use of as many as real life production information in the input matrix for representing the CF problem.

In this research work, the real life production factors like, operational time of the parts in the machines known as workload data or ratio level data, operational sequence of the parts known as ordinal level data and batch size are considered for the problem representation. The methodology uses soft computing techniques like genetic algorithm (GA) and neural network to tackle the CF problem. In recent years, soft computing techniques have fascinated scientists and engineers all over the world because such techniques possess the ability to learn and recall as similar to the main functions of the human brain. They find better approaches to real world problems since soft computing incorporates human knowledge effectively. It deals with imprecision and uncertainty and learn to adapt to unknown or changing environment for better performance. In neural network, adaptive resonance theory (ART1) gives good results for binary MPIM CF problem. ART1 is not suitable for non-binary input pattern. Hence, in this work, suitable modification is included in the basic ART1 to incorporate the operational time of the parts, a ratio level non-binary data. For dealing with sequence of operations of the parts, an ordinal level non-binary data, a supplementary procedure is first implemented to convert the non-binary data into a suitable binary data and subsequently by feeding to the basic ART1 networks to solve the CF problem. Finally both operational time and operational

sequence are combined and represented in a single matrix. The modified ART1 used for solving CF problem with operational time is applied to solve the problem with combination of operational time and sequence. The CF problem without any objective function is solved effectively by ART1 approach.

For solving the CF problem with objective functions like total cell load variation (CLV) and exceptional elements, GA is proposed in this research work. CLV is calculated as the difference between the workload on the machine and the average load on the cell. Exceptional elements are the number of non-zero elements present in off diagonal blocks of the output matrix. Both the objective functions are combined to get a multi objective CF problem and solved by using GA. In the past, several performance measures like grouping efficiency and grouping efficacy have been proposed to find out the goodness of the output clusters. But most of them are applicable only for binary data representation. In this research work, suitable performance measures are proposed to measure the goodness of the block diagonal structure of the output matrix with ratio level data, ordinal level data and combination of both data. The algorithms are designed to handle problem of any size and they are coded with C^{++} and run on Pentium IV PC. Computational experience with the proposed techniques is presented and the results are compared with the problems available in open literature. The results are encouraging and the methodologies are found more appropriate for large scale production industries. Computational results suggest that the proposed approaches are reliable and efficient both in terms of quality and in speed in solving CF problems. Several directions for future studies are also addressed in this research.

Keywords: *Cell Formation; Adaptive Resonance Theory; Genetic Algorithm; Ratio level data; Ordinal level data; Cell load variation; Exceptional Elements; Grouping Efficiency.*

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

GLOSSARY OF TERMS

ART 1	–	Adaptive resonance theory 1
CF	–	Cell formation
CLV	–	Total cell load variation
CMS	–	Cellular manufacturing systems
EE	–	Exceptional elements
GA	–	Genetic algorithm
GER	–	Grouping efficiency for operational time data
GT	–	Group technology
GTE	–	Group technology efficiency
IM	–	Inter-cell moves
JIT	–	Just-in-time
MCD	–	Matrix of combined data
MGE	–	Modified grouping efficiency
MPIM	–	Machine-part incidence matrix
PFA	–	Production flow analysis
PMIM	–	Part-machine incidence matrix
PMPM	–	Part-machine precedence matrix
ROCE	–	Ratio ordinal combined efficiency
TM	–	Total moves
Vs	–	Versus
WIP	–	Work-in-process

NOMENCLATURE

English symbols

a_{ij}	–	Element of Zero-one MPIM
C	–	Number of cells
i	–	Machine index
I_p	–	Maximum number of inter-cell travel possible in the system.
I_r	–	Number of inter-cell travel actually required by the system.
j	–	Part index
k	–	Cell index
M	–	Number of machines
m_{kj}	–	Cell part matrix of average load
N	–	Number of parts
n_o	–	Number of operations (w = 1,2,3,…,n_o)
N_e	–	Total number of exceptional elements
N_{ek}	–	Number of elements in cell ‘k’
N_p	–	number of parts having operation in i^{th} machine
N_{vi}	–	Number of voids in i^{th} machine.
N_{vk}	–	Number of voids in cell ‘k’
T_{ik}	–	Total time taken by i^{th} machine in cell ‘k’
T_o	–	Total elements in the matrix
T_{ptk}	–	Total processing time inside cell ‘k.’
T_{pto}	–	Total processing time outside the diagonal blocks
T_w	–	Total processing time of the matrix
t_{njw}	–	‘1’ if w,w+1 operations are in same cell, ‘0’ otherwise
W_{ji}	–	Workload matrix
X_{ik}	–	‘1’ if machine ‘i’ is in the cell k, otherwise ‘0’
Y_{jk}	–	‘1’ if part j is in the cell k, otherwise ‘0’
Z_1	–	CLV
Z_2	–	EE

Greek symbols

η	–	Grouping efficiency
τ	–	Grouping efficacy
ρ	–	Vigilance threshold
γ	–	Grouping index
Ψ	–	Proportion of Exceptional elements
φ	–	Proportion of Voids
μ	–	Matching score of neuron

Chapter – 1

Introduction

1.1 INTRODUCTION

Today, the manufacturing industries of all countries play an important role in realising the real prosperity. With the growth of the seller's and buyer's markets towards globalization, the manufacturing industries need to deal with the challenges facing it. This has resulted in the materialization of automated industries with high performance of manufacturing systems. Traditional manufacturing systems are not able to satisfy these requirements (Saravanasankar 2005). Hence, the manufacturing industries are motivated to enhance the productivity and flexibility of the system towards achieving a competitive edge. Cellular Manufacturing Systems (CMS) evolved as a solution to efficient batch type production of a variety of part types with low set up time, low work-in-process inventory (WIP), short manufacturing lead time, high machine utilization and high quality.

1.2 MANUFACTURING SYSTEMS

Manufacturing systems traditionally fall into three categories of layouts. They are job shop production, batch production and mass production. The job shop production is designed to manufacture with the maximum flexibility, wide variety of products with small lot sizes. The job shop manufacturing which follows process layout is shown in Figure 1.1. In batch production, the parts move in batches for efficient processing. Therefore, each part in a batch must wait for the remaining parts in its batch to complete processing before it moves to the next stage. This will lead to increased production time, high level of in-process inventory, high production cost and low production rate. For instance, if a batch (medium quantity i.e. 100 units to 10000 units per year) of one product is made and then the facility is changed over to produce a batch of the next product and so on as orders for each product are frequently repeated, the changeover time or setup time is more in the batch production system. The loss of production time is a major disadvantage of batch production system. In contrast, the product layout is preferred for high volume and low variety of products to improve the production rate. A typical product layout is shown in Figure 1.2.

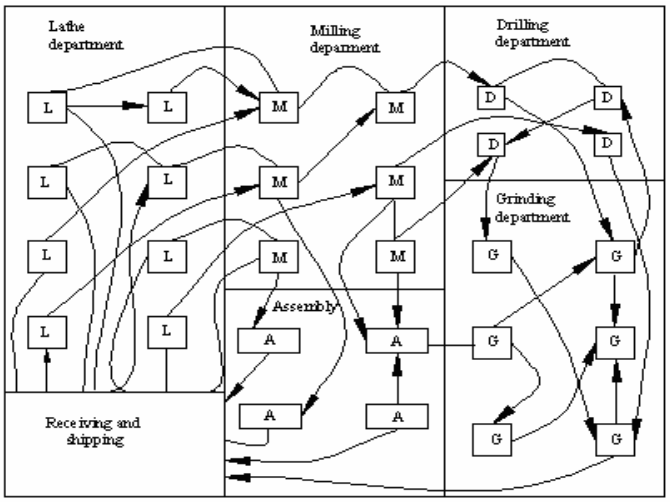


Figure 1.1 Process Layout

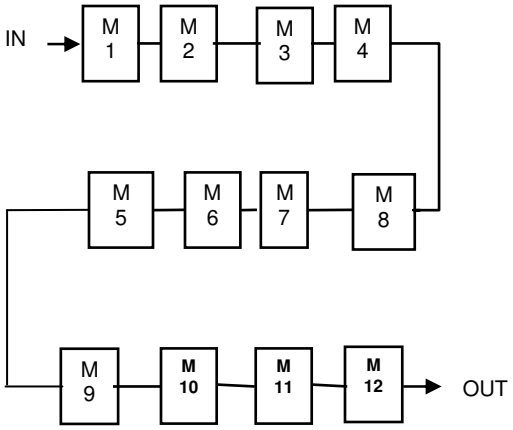


Figure 1.2 Product Layout

The manufacturing industries having batch production environment are determined to achieve reduced lead time, reduced setup time, and increased machine utilization. Cellular manufacturing stands as one of the efficient proposition of achieving the goal in this direction. Cellular manufacturing is one of the most important technological improvements applied to the batch processing industries. Cellular Manufacturing is the application of Group Technology (GT). GT is the management philosophy that believes similar activities should be done similarly. Cellular manufacturing is a hybrid system linking the advantages of both the job shop (process layout) and the flow shop (product layout) of the continuous flow line. It focuses on the creation of manufacturing cells within which a number of part families are manufactured. A cell consists of a set of functionally dissimilar machines, which are placed in close proximity to one another and dedicated to the manufacture of a set of part families. A part family is a set of parts that are similar in terms of processing requirements. A fundamental issue of cellular manufacturing is the determination of part families and machine cells.

Research into the application of group technology for manufacturing first began during the late 1950's. Around this time, researchers began to recognize that some parts share common manufacturing approaches. They soon concluded that parts with common manufacturing attributes could be grouped together and processed in a manner similar to mass production. Using this theory, they would

create groups of similar parts and then dedicate groups of machines and tools specific to the production of these parts to reduce setup times. The first researcher to propose this theory was S.P. Mitrofanov of the U.S.S.R.

In subsequent years, several classifications and coding systems for forming part family were proposed. Companies first started to reorganize manufacturing facilities along GT lines in the early 1960's and the concept of GT was strongly accepted round the globe. The approach of Production Flow Analysis (PFA) introduced by Burbidge (1963) considers wider aspects of production such as factory flow system, plant layout etc. Once the part families are identified, the machines are arranged in machine cells to produce a specific part family. When the machines are organized in cells, the system is known as cellular manufacturing system (CMS). CMS has been considered as an alternative to conventional batch production system where different products are produced intermittently in small lot sizes. Cellular manufacturing overcomes major problems of batch-type manufacturing including frequent setups, excessive in-process inventories, long throughput times, complex planning and control functions etc. and provides the basis for implementation of manufacturing techniques such as Just-In-Time (JIT) and Flexible Manufacturing Systems (FMS). The advantages and limitations including field of application of CMS are discussed in Appendix I.

1.3 CELL FORMATION PROBLEM

In Cellular Manufacturing, the main objective is to group the machines in to machine cells and the parts into part families based on similarities in design and manufacturing attributes. The identification of machine groups and corresponding part families is known as "cell formation". For that purpose the machine part incidence matrix (MPIM) or part machine incidence matrix (PMIM) is constructed which consists of '0's and '1's inside the each blocks of the matrix where '1's represent the visit of parts to machines and '0's represent non-visit. The problem of grouping involves decision making of various parameters like number of cells, number of machines to be accommodated in a cell etc. A GT layout after identification of cells is shown in Figure 1.3.

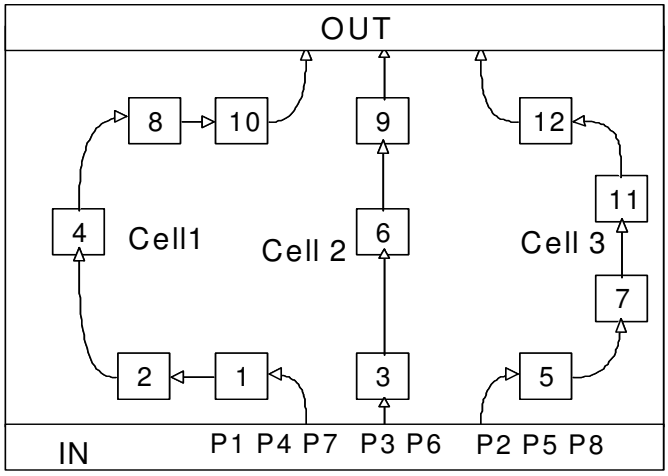


Figure.1.3 A Typical GT layout P1, P2 ….P8 represent parts
Numbers inside the blocks represent machines

1.4 CELL FORMATION APPROACHES

The problem of cell design is a very complex exercise with wide ranging implications for any organizations. Normally, cell design is understood as the problem of identifying a set of part types that are suitable for manufacture on a group of machines (Wemmerlov and Hyer. 1987).

Various approaches have been developed to solve the CF problem (Miltenburg and Zhang, 1991), each of them have their own advantages and drawbacks. Kandiller (1994) has made a comparative study on CF problems.
The three methods of cell formation are:

(i) Machine grouping

(ii) Part family grouping

(iii) Machine-part grouping

1.4.1 Machine Grouping

Some researchers have attacked the problem of group formation as a two-stage process. In this first stage of their analysis, they group machines and form cells based on the information contained in the part routings. The next stage usually consists of allocating parts to cells and re-evaluating the cells on the other factors such as machine utilization. Gupta (1991) made a study on

clustering algorithms for CF problems. The techniques existing for machine grouping can be broadly classified as follows:

- Non- Algorithmic Procedures
- Algorithmic Procedures

1.4.2 Part Family Grouping

In this grouping, the part families are first identified and then, grouping of machines into cells are made. This method is of restricted value nowadays, but it is still useful in single machining centers. Existing techniques for part family grouping based on routing sheet information are:

- Classification and coding
- Cluster analysis

1.4.3 Machine – Part Grouping

When one attempts to group parts into part families and machines into cells simultaneously, then such a procedure is defined as machine-part grouping. The three main sub classifications are

- Manual Technique.
- Combinatorial procedures.
- Algorithmic methods.

Table 1.1 shows some bench mark clustering methods. Apart from above techniques, soft computing is adopted these days for cell formation due to their generalization capability and easiness. A brief overview of cell formation using soft computing techniques is provided in Appendix II.

1.5 CELL FORMATION CONSIDERING OPERATIONAL TIME AND SEQUENCE

In cell formation problem, usually zero-one MPIM or PMIM, which is built from route sheet information, has been used as input. Later, researchers started to make use of other information or production factors like workload on the machines, operational sequence of the parts, batch size of the parts, machine

capacity, etc. that are available in the shop floor. The process of clustering machines into machine cells and parts into part families without using such information may lead to inferior manufacturing plans. Hence, the need arises to use non-binary data for obtaining groups or clusters of machines and parts. In this research work, some of the real life production factors like operational time and sequence of the parts are considered to make cell formation.

Table 1.1 Some clustering methods available in the literature

Clustering Methods	Approaches
Machine cell formation	Adaptive Resonance Theory (ART) Networks (Kao and Moon, 1991), Simulated Annealing (Boctor, 1991), Genetic Algorithm (Venugopal and Narendran, 1992b), CASE (Nair and Narendran 1998), ACCORD (Nair and Narendran, 1999), Dissimilarity coefficients (Prabhakaran et al. 2002), Tabu Search (Logendran et al. 1994), Ants colony Systems (Solimanpur et al. 2004)
Part family formation	Production flow analysis (Burbidge, 1977), Generalized part family formation (Moon and Chi, 1992), Part assignment (Chen and Cheng, 1995), Coding systems (Singh and Rajamani, 1996), Fuzzy ART (Suresh et al. 1999), Membership index (Zolfaghari and Liang, 2003).
Concurrent Machine cell-part family formation	Bond Energy Approach (McCormick et al. 1972), Graph Theoretical Approach (Rajgopalan and Batra, 1975), Set Lattice Theoretic Approach (Purcheck, 1975), Manual Technique (Burbidge, 1977), Rank Order Clustering (ROC) (King, 1980), Direct Clustering Algorithm (Chan and Milner, 1982) MACE (Waghodekar and Sahu, 1984), Ideal Seed Methods (Chandrasekaran and Rajagopalan, 1986a), Linear Programming (Kusiak, 1987), MODROC (Chandrasekaran and Rajagopalan, 1986b), ZODIAC (Chandrasekharan and Rajagopalan, 1987), CLOSE Neighbour Algorithm (Boe and Cheng, 1991), GRAFICS (Srinivasan and Narendran, 1991)

1.5.1 Ratio level data

The workload information is commonly considered as ratio level data in CF problem and a modified incidence matrix is formed with this data. The total processing time of a part is computed by multiplying the production quantity of the part with its unit processing time. The workload (or ratio) value replaces '1's

in the incidence matrix. The resultant workload values will take any value in the ratio scale, and they represent the ratio level data (George et al. 2003).

1.5.2 Ordinal level data

The operational sequence of the parts is usually considered as ordinal level data. In general, if the resultant values in the incidence matrix take any value in the ordinal scale, they constitute the ordinal level data. For example, operation sequence of the parts and the number of parts in a batch are well known ordinal level data.

The goodness of resulting cells may be tested using performance measures specifically designed for cell formation. The performance measures for resulting cells, when different kinds of data sets are used as inputs, are discussed in Appendix III.

1.6 OBJECTIVES OF THE RESEARCH WORK

The main objectives of this research is to

(1) Undertake in-depth study on cell formation for designing cellular manufacturing systems.

(2) Appraise critically the existing approaches for cell formation and find out research trend for cell formation considering production factors.

(3) Propose suitable methodologies for cell formation considering real time production factors.

Specifically, the research work focuses on the followings:

(i) To develop suitable methods for cell formation problem considering practical production factors like operational time of the parts and sequence of the parts.

(ii) To propose improved algorithms based on soft computing techniques for cell formation using above production factors.

(iii) To compare the results obtained from proposed algorithms with existing methods through exhaustive computation.

(iv) To propose performance measures for assessment of goodness of the block diagonal structure (outputs).

1.7 NEED FOR THE RESEARCH

As customers strive for quality products in short lead time, the batch type production industries must orient towards meeting increasing demand in volume and variety in short throughput time in order to have competitive edge. The research towards design of layout for machines and materials in CMS provides a platform for managers to arrive at some useful solutions in this direction. CMS necessitates clustering of parts into part families and machines are allocated in machine cells so that a machine cell is responsible for producing certain part families. Majority of the clustering techniques use binary representation for input data without taking into account other production factors. Such an approach leads to inefficient flow of materials resulting in deterioration of system effectiveness. This study focuses on cell formation considering real life production factors like operational time and sequence of the parts. The proposed methodologies provide fast solutions of the problem since they make use of soft computing techniques so that it can be used conveniently in the shop floor. Further, it has been attempted to propose performance measures suitable for cell formation considering practical production factors.

1.8 ORGANIZATION OF THE THESIS

Seven chapters are presented in this thesis including chapter 1 and the rest of the thesis is organized as follows:

Chapter 2: The literature review is presented through exhaustive study.

Chapter 3: Machine cell formation using operational time of parts is developed using a neural network approach

Chapter 4: Machine cell formation using operational time of parts is approached using GA

Chapter 5: Cell formation considering operational sequence of parts using neural network is developed

Chapter 6: Cell formation with combined objective function (combination of operational time and sequence of parts) using neural network is developed

Chapter 7: The conclusion and scope for future work are given.

Chapter – 2

Literature Review

2.1 INTRODUCTION

In this chapter, critical appraisal of clustering techniques used in CMS is made through exhaustive literature review. Various existing approaches for cell formation are discussed subsequently. Group technology ideas were first systematically presented by Burbidge (1963) following the pioneering work of Mitrofanov (1959). The literature on cell formation can be broadly classified in two ways – one based on techniques used for cell formation and other one the way the cell formation problem is modeled. Crama and Oosten (1996) made a study on various models available for CF problems.

2.2. CLASSIFICATION OF CELL FORMATION FROM TECHNIQUE POINT OF VIEW

For the technique based classification, a Taxonomic framework of GT is shown in Figure 2.1 followed by discussions.

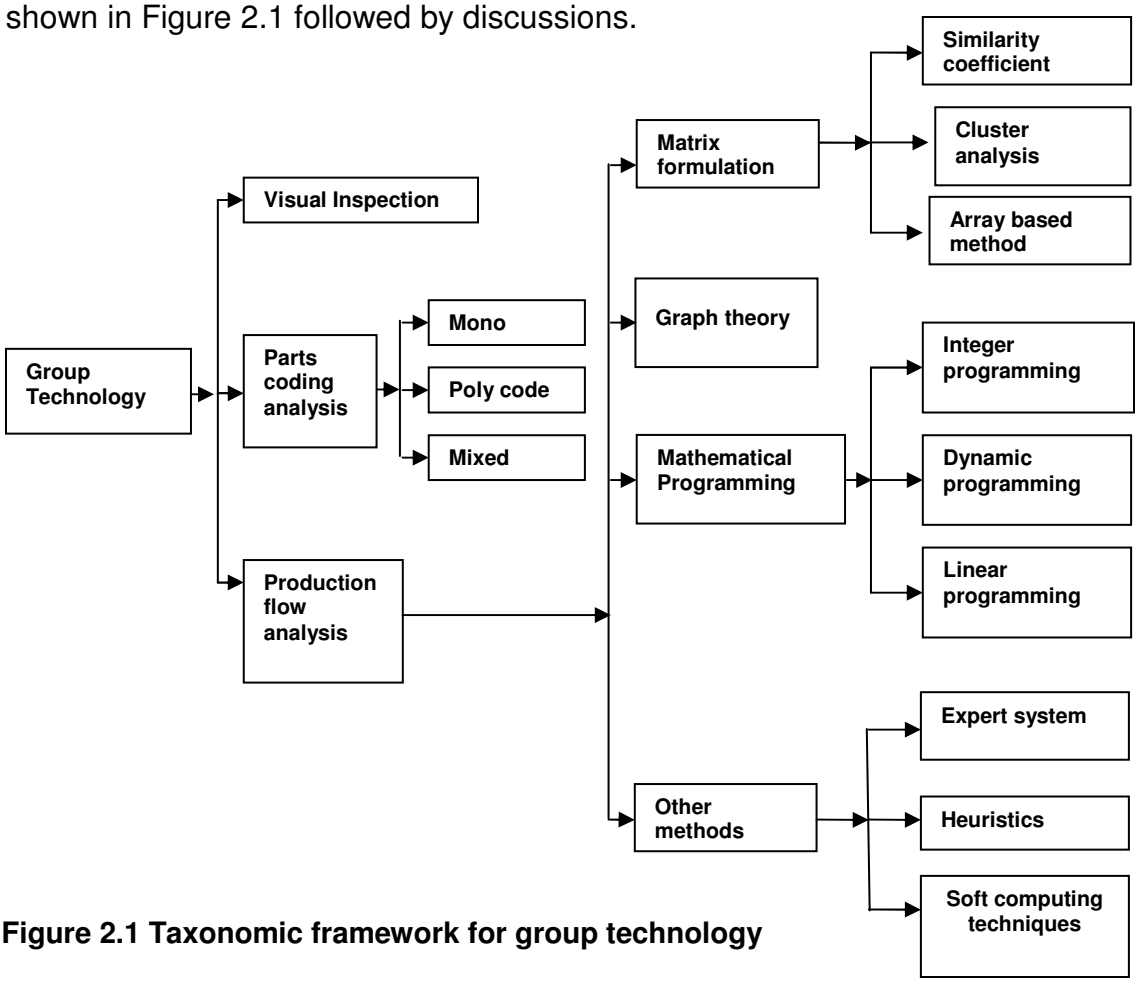


Figure 2.1 Taxonomic framework for group technology

2.2.1 Visual Inspection

The easiest approach in part family formation is visual inspection method. This method is to examine the information and perform the classification using human eye, also known as eyeballing method. The visual inspection method is the least sophisticated and least expensive method. The time consumed in this method is very less when compared to all other methods. Nevertheless this method is considered least accurate when compared to the other methods.

2.2.2 Classification and Coding

Classification and coding is an essential and effective tool for successful implementation of group technology concept. A code may be numbers or letters or a combination of numbers and/or letters which are assigned to the parts for information processing (Ham et al. 1985). Parts are classified based on relevant characteristics such as dimensions, type of material, tolerance, operations required, basic shapes, surface finish etc. In this approach, each part is assigned a code which is a string of digits that store information about the part. The digits include numerical numbers and alphabetical letters. Singh and Rajamani (1996) described about coding systems in their work. Some of the coding systems include hierarchical structure (also called monocode), chain-type structure (also known as polycode) and mixed mode structure which is a hybrid of monocode and polycode.

2.2.3 Production Flow Analysis (PFA)

The concept of production flow analysis was introduced by Burbidge (1963). The aim of the technique as stated by Burbidge (1971) is finding the families of components and associated groups of machines for group layout by a progressive analysis of the information in route cards. It is based on the idea that parts with similar routes can be made in the same group, and it finds both a division of machines into groups and of parts into families of parts, which they make. This concept is used also by other cell formation approaches. It can concurrently form machine groups as well as part families. The main

disadvantage with implementation of PFA is the manual work involved in grouping parts and machines. Burbidge (1971) did not give any other way for grouping, but trying all the possibilities and combinations manually. It is practically impossible to form cells in a factory, which may have thousands of parts and hundreds of machines. But, the basic principle of PFA builds the foundation for developing sophisticated approaches later. Burbidge (1971) suggested that a part can have more than one routing and a process can be done on more than one type of machines. This was a major and very important suggestion which helped to explore various economic and technical possibilities in forming cells. Burbidge (1975) introduced a holistic approach to GT called Production Flow Analysis. It discussed the production situation and recommended a systematic solution to the problems of batch production. Burbidge (1977) introduced a two dimensional representation with a tick mark used to indicate the visit of a component to a machine. The method uses hand computations, which limits its applicability.

2.2.4 Similarity Coefficient Method

The similarity coefficient approach was first suggested by McAuley (1972). The basis of these methods is to measure the similarity between each pair of machines and then to group the machines into families based on their similarity measurements. Some studies have proposed to measure dissimilarity coefficients in stead of similarity coefficient for cell formation. Prabhakaran et al. (2002) have used dissimilarity coefficients for generalized cell formation taking into account the operation sequences and production volumes of parts. Most similarity based methods employ machine – component chart. Some of the methods, which use this approach, are Single linkage clustering algorithm (McAuley, 1972), Average linkage clustering algorithm (Seifoddini and Wolfe, 1986) etc.

2.2.5 Cluster Analysis

These methods can be classified as hierarchical and non-hierarchical methods. Standard or specially designed clustering techniques can be used to

make clusters of either parts or machines. Among these, McAulay (1972), McCormick et al. (1972), Carrie (1973), King (1980), King and Nakornchai (1982), Chan and Milner (1982), Waghodekar and Sahu (1984), Kusiak (1985), Mosier and Taube (1985), Stanfel (1985), Chandrasekharan and Rajagopalan (1986a, b), Kusiak (1987), Seifoddini and Wolfe (1986), Seifoddini (1989), Chu and Tsai (1990), Srinivasan and Narendran (1991), Shafer and Rogers (1993a) found in the literature are popular methods. Machine – component group analysis (MCGA) is based on production flow analysis. In MCGA based methods the machine-component groups are formed by permuting rows and columns of the machine-component chart in the form of a zero-one matrix. Some of the MCGA methods are Rank order clustering by King (1980), Bond energy algorithm by McCormick et al. (1972) etc. Dimopoulos and Mort (2001) has developed a hierarchical algorithm for a simple cell formation.

2.2.6 Array Based Method

These methods treat the rows and columns of the zero-one matrix as binary words and rearrange them to obtain a block-diagonal structure. The rank order clustering algorithm is the most popular array-based method for GT (King 1980). Subsequent modifications and improvements over rank order clustering algorithm have been described by King and Nakornchai (1982), Chandrasekharan and Rajagopalan (1986a). The direct clustering analysis (DCA) has been proposed by Chan and Milner (1982), and bond energy analysis by McCormick et al. (1972).

2.2.7 Graph Theoretic Approaches

Graph Theoretic Approach represents the machines as vertices and the similarity between machines as the weights of the arcs. Rajagopalan and Batra (1975) suggested the use of graph theory to form machine groups. Chandrasekaran and Rajagopalan (1986a) proposed an ideal seed non-hierarchical clustering algorithm for cellular manufacturing. Ballakur and Steudel (1987) developed graph searching algorithms which select a key machine or component according to a pre-specified criterion. Vohra et al (1990) presented a

non-heuristic network approach to from manufacturing cells with minimum inter-cellular interactions. Srinivasan (1994) presented an approach using minimum spanning tree for the machine cell formation problem. A minimum spanning tree for machines is constructed and the seeds to cluster components are generated from this tree. Veeramani and Mani (1996) described a polynomial-time algorithm based on a graph theoretic approach for optimal cluster formation called as vertex-tree graphic matrices.

2.2.8 Mathematical Programming

A number of research studies for cell formation using mathematical programming approach appeared in literature. They are classified under integer programming (Kusiak 1987, Co and Araar 1988), dynamic programming (Ballakur and Steudel 1987), goal programming (Shafer and Rogers, 1993a), and linear programming (Boctor 1991). Ramabhatta and Nagi (1998) developed a branch-and-bound procedure to obtain the cell configuration that tends to yield minimum inter-cell flows under the assumption of alternative routings for each part. Mathematical programming received extreme attention because its ability to consider practical constraints and objectives of the company when designing cells. The approach goes in two steps. First, a mathematical model representing the objectives and constraints of an organization is formulated and then the model is optimized.

Kusiak (1987) developed clustering problem known as p-median model. The objective function is to maximize the total sum of similarities and the constraints are (i) one part belongs to exactly one family (ii) number of part families are specified. Once the part families are formed, corresponding machines are assigned to the cells. Choobineh (1988) uses a sequential approach forming part families in the first stage and then a cost based mathematical programming method to allocate machines to part families to form cells. Rajamani et al. (1990) developed integer programming models to form cells sequentially as well as simultaneously. Factors such as inventory cost, machine setup and material handling, and machine depreciation were considered by Askin

and Subramanian (1987). An assignment model was given by Srinivasan et al (1990) to form part families in group technology. A mathematical programming approach to joint cell formation and operation allocation in cellular manufacturing was proposed by Atmani et al. (1995). A zero-one integer programming model is formulated. The objective considered in this model is, to form machine groups and allocate operations in such a way as to minimize operation costs, re-fixture costs and transportation costs. Zahir Albadawi et al (2005) has developed a mathematical model using eigen value matrix for cell formation problems.

2.2.9 Cell Formation using Heuristics

Heuristic algorithms are used to provide quick approximate solutions to hard combinatorial optimization problems. A heuristic algorithm is called an approximate algorithm where the performance of the heuristic is assessed in terms of worst and average case behaviour. They do not guarantee optimal solutions. Waghodekar and Sahu (1984) proposed an algorithm called MACE to solve the GT problem. The method uses similarity among machines. Panneerselvam and Balasubramanian (1985) developed a method, which groups the components having approximately the same process sequences so that they can be processed on the same line. Wemmerlov and Hyer (1986) provided a framework for classifying descriptive and analytic procedures for the component-family and machine-cell formation problems. Beaulieu et al.(1997) considers the machine selection problem for the design of new CMS. This method considers machine capacity, alternative routing and constraints on cell size. MODROC was developed by Chandrasekaran and Rajagopalan (1986b) which is an extension of basic ROC method. ZODIAC has been proposed by (Chandrasekharan and Rajagopalan 1987). Similarly, close neighbour algorithm by Boe and Cheng (1991). GRAFICS by Srinivasan and Narendran (1991), CASE by Nair and Narendran (1998), ACCORD by Nair and Narendran (1999) are some well known heuristics for solving CF problem found in the literature.

2.2.10 Cell Formation using Soft Computing Techniques

Since cell formation problems are non-polynomially complete in nature (Nair and Narendran 1999), it is difficult to obtain solutions that satisfy all constraints. Therefore, it is expected to make use of simple but efficient computing techniques. Soft computing technique is found more suitable for such type of problems and capable of producing good results (Venugopal 1999). Soft computing is an emerging approach to computing which parallels the remarkable ability of the human mind to reason and learn in an environment of uncertainty and imprecision (Jang et al. 2002). Soft computing is an innovative approach for constructing computationally intelligent systems. It is realized that complex real world problems require intelligent systems that combine knowledge, techniques and methodologies from various sources. Soft computing techniques make the integration of neural network, fuzzy systems and other meta-heuristics together with certain derivative free optimization techniques. Soft computing constitutes Genetic Algorithm (GA), Simulated Annealing (SA), Artificial Neural Networks (ANN), fuzzy set theory etc. Since 1990 the applications of soft computing techniques to GT problems have been encouraging (Venugopal 1999). The literature concerning CMS using three major soft-computing techniques like fuzzy set theory, meta-heuristics, and artificial neural networks are discussed in followings.

(i) CMS using fuzzy set theory

Few studies have appeared in the areas of artificial intelligence and fuzzy clustering approaches to cell formation. Kusiak and Ibrahim (1988) developed a knowledge based system which takes advantage of expert system and optimization considering machine capacity, material handling capabilities, technological requirements and cell dimensions to form cells. Singh (1993) introduced the concept of multi-dimensional similarity coefficient using syntactic pattern recognition and developed and algorithm to form natural part families. Most of the approaches to cell formation discussed earlier assume that the information about processing cost, processing time, part demand, etc. is precise. It is also assumed that each part can only belong to one part family. However

there exist parts whose lineages are less evident. Fuzzy clustering provides a solution to such problems. But only in few studies made by Xu and Wang (1989) and Chu and Hayya (1991), issues of vagueness in cell formation has appeared. Other fuzzy logic approaches are given by Ben-Arieh and Triantaphyllou (1992) and Burke and Kamal (1992). FACT proposed by Kamal and Burke (1996) is an algorithm based on fuzzy ART to solve CF problem.

(ii) CMS using metaheuristics

Harhalakis et al. (1990) developed a procedure based on SA for the design of manufacturing cells for minimizing the inter-cell traffic with cell size constraint. They also demonstrated the application of their algorithm on an industrial problem. Venugopal and Narendran (1992a) considered several real-life factors such as processing time, volume of components and capacity of machine and solved the problem using SA. A solution procedure based on GA for cell formation with multiple objectives was developed by Venugopal and Narendran (1992b). Logendran et al. (1994) proposed an approach based on Tabu Search (TS) for the design of CMS when alternative process plans are considered. Vakharia and Chang (1997) developed two heuristic methods based on combinatorial search methods SA and Tabu search to address the cell formation problem. Murthy and Srinivasan (1995) developed a SA and a heuristic algorithm for fractional cell formation. In their algorithm the movement of component from GT cells to remainder cell is allowed but not among GT cells to minimize exceptional elements among GT cells. Goncalves and Resende (2004) developed evolutionalry algorithm for cell formation Vitanov et al. (2007) developed a heuristic algorithm known as heuristic rule based logic algorithm (HERBAL) as a tool for designing cellular layout. Tabu searches have been successfully used to generate solutions for a wide variety of combinatorial problems (Jeffrey Schaller 2005). Popular meta-heuristics applied in the field of CMS are SA by Boctor (1991), GA by Zhao and Wu (2000), TS by Wu et al (2004) and Ants Colony Systems by Solimanpur et al. (2004).

(iii) CMS using neural networks

Neural networks are now of major interest because when it is connected to computer, it mimics the brain and bombard people with much more information. The existing ANN approaches for GT application is shown in Table 2.1 These have shown promise for solving many combinatorial optimization problems.

Kao and Moon (1990) proposed an interactive activation and competition where part similarities and machine similarities are considered together in the formation of part families and machine cells and suggested generalized part family formation methods (Moon and Chi 1992). Kaparthi and Suresh (1992) and Dagli and Huggahali (1991) used ART1 (Carpenter and Grossberg, 1987, 1988) to group parts or machines. Malave and Ramachandran (1991) have applied a competitive learning rule to the parts and machines formation problem. For the part family formation problem, the input to the neural network is the process plan of each part. This network offers a mechanism to identify the ratio of the number of shared (bottleneck) machines to the total number of machines used in each cell. A notable development with neural networks in recent years is that they have been found to applicable for sequence-based clustering, using networks such as Kohonen's self-organizing feature maps (SOFM) by Melody (2001) and Fuzzy ART neural network by Suresh et al. (1999).

Godfrey C Onwabolu (1999b) used self-organizing map (SOM) neural network for design of parts for cellular manufacturing. Lozano et al. (1993) used Harmony theory model neural network for CF problem. Dobado et al. (2002) used fuzzy neural network for part family formation. The performance of ART1 network based CF has been investigated by Kusiak and Chung (1991), Kaparthi and Suresh (1992), Liao and Chen (1993), Dagli and Huggahalli (1995), Chen and Cheng (1995), Chen et al. (1996), Enke et al. (2000) and Ming-Laing et al (2002). Chen and Cheng (1995) pointed out the weakness of the ART1 approach that the ability of a grouping solution is highly dependent on the initial disposition of the MPIM especially in the presence of bottleneck machines and parts.

Table 2.1 Neural network approaches with source applied to group technology

Application	Neural network models	Source
Part family and Machine CF	IAC Models	Moon (1990)
		Moon (1992)
		Moon and Chi (1992)
	ART Based Models	Kusiak and Chung (1991)
		Dagli and Huggahalli (1991)
		Kaparthi and Suresh (1992)
		Liao and Chen (1993)
		Dagli and Huggahalli (1995)
		Chen and Cheng (1995) Enke et al. (1998) (2000) Venkumar and Haq (2005)
	ART + SOFM	Venugopal and Narendran (1994)
	Fuzzy ART Models	Suresh and Kaparthi (1994)
		Burke and Kamal (1992) (1995)
		Peker and Kara (2004)
	Competitive Learning	Chu (1993)
	Harmony Theory Model	Lozano et al. (1993)
	Self Organizing	Rao and Gu (1995)
	Competitive Learning + Self Organizing	Malakooti and Yang (1995)
	Ortho-Synapse Hopfield	Zolfaghari and Liang (1997)
	Adaptive Hamming net	Kyung-Mi Lee et al. (1997)
	Kohonen Self-Organizing Map	Melody (2001) Venkumar and Haq (2006)
	Fuzzy Min-Max	Dobado et al. (2002)
Feature-based part family formation and new part assignment	Back Propagation (BP) Models	Kao and Moon (1991a)
		Moon and Roy (1992)
		Kao and Moon (1998)
		Godfrey Y Onwabolu (1999a)
	ART Models	Kao and Moon (1991b)
		Liao and Lee (1994)
		Sung Youl Lee and Fischer (1999)
	Self Organizing	Godfrey Y Onwabolu (1999b)
	Feed-Forward	Kusiak and Lee (1996)
Design retrieval	Fuzzy Associative Memory	Bahrami, Dagli and Modarress (1991)
	Hopfield Models	Venugopal and Narendran (1992c)
Bit image coding and part grouping	BP Models	Kaparthi and Suresh (1991)
		Chung and Kusiak (1994)

Ming-Laing et al. (2002) developed a modified ART1 network which integrated with an effective Tabu Search optimization technique to solve CF problem.

2.3 CLASSIFICATION OF CELL FORMATION FROM MODELLING POINT OF VIEW

From the literature of CMS, it is observed that cellular manufacturing aims at formation of machine cells for achieving the benefits of mass production to batch production with higher values of variety, product-mix and total quantity. Researchers have proposed various algorithms based on different approaches to obtain disjoint machine cells. Usually zero–one matrix, referred as MPIM obtained from the route sheet information is used to form machine cells and part families. Some of the studies explicitly focus on cell formation with real life production factors mentioned as second type of classification. In such studies again, few of the studies consider objective function while solving cell formation problem and few of them do not consider objective function. Based on the exhaustive collection of literature, they are integrated as given in the followings.

2.3.1 Cell Formation without considering Production Factors

Iri (1968) developed the cluster identification algorithm. The algorithm finds mutually separable clusters in a binary matrix provided they exist. The algorithm is claimed to be computationally more efficient for problems not involving exceptional elements. The other popular algorithms in this category are ROC I, ROC II, MODROC, ZODIAC, and MACE.

2.3.2 Cell Formation with Production Factors

Researchers started considering production factors while processing Cell Formation. It can be classified into two categories in this section. i) Cell Formation with single production factors and ii) CF with multiple Production factors

(i) Cell Formation with Single Production Factor

Vannelli and Ravikumar (1986) proposed a method to find minimum number of bottleneck cells for grouping part-machine families. Vakhaira and Wemmerlov (1990) considered a cell formation which integrates the issue of cell formation and within cell material flows using similarity co-efficient approach to

cluster parts and machines. But, this approach failed to take into account the issues of number of cells and duplication. Heragu and Kakuturi (1997) proposed a three stage heuristic approach incorporating material flow considerations with alternative process plans for grouping and placement of cells.

(ii) Cell Formation with Multiple Production Factors

Askin and Subramanian (1987) used a binary clustering algorithm for grouping parts and machines. They evaluate candidate configurations based on fixed and variable machine costs, set-up costs, cycle inventory, work-in-process inventory and material handling. Factors such as inventory cost, machine setup and material handling, and machine depreciation were considered by Askin and Chiu (1990). Taylor and Taha (1993) performed sensitivity analysis relative to several parameters in an effort to identify factors affecting cellular manufacturing system design in general. Sankaran and Kasilingam (1993) developed an integer programming model to determine simultaneously cell size, capacity selection and cell membership in a GT based flexible manufacturing system. Sung-lyong and Wemmerlov (1993) suggested a new heuristic methodology that incorporates the concept of reallocating operations to alternative machines, while meeting capacity constraint for manufacturing cell formation. Hsu and Su (1998) also considered genetic algorithm based solution for CMS design. Here the factors considered are inter-cell and intra-cell part transport factor, machine investment cost, intra-cell load unbalance and inter cell load unbalance. Won and Lee (2001) considered operation sequences and production volumes. Prabhakaran et al. (2002) have used dissimilarity coefficients for generalized cell formation taking into account the operation sequences and production volumes of parts. Yin and Yasuda (2002) made a study that takes alternative process routing, operational sequence, operational time and production volume into account. George et al. (2003) considered operational time and operational sequence of the parts and combination of these factors in a single matrix in their study, an analytical-iterative clustering algorithm for cell formation.

2.3.3 Cell Formation without considering Objective Function

GRAFICS by Srinivasan and Narendran (1991) is a nonhierarchical clustering algorithm for CF problem without considering objective function. CASE Nair and Narendran (1998) found out the similarity coefficient without considering objective function. Suresh et al (1999) made a study on sequence-dependent clustering of parts and machines without considering any objective function. Park and Suresh (2003) used Fuzzy ART neural network for CF problem without any objective function. Venkumar and Haq (2005) have adapted ART 1 to apply for CF problem without any objective function and come out with improved results.

2.3.4 Cell Formation with Objective Function

Han and Ham (1986) suggested that part families can be formed more effectively based on the classification codes because both the manufacturing and design characteristics are considered. They proposed a multi objective cluster analysis algorithm for forming part families. Kusiak (1987) developed clustering problem known as p-median model with a objective to maximize the total sum of similarities and the constraints are (i) one part belongs to exactly one family (ii) number of part families are specified. Wei and Kern (1991) developed a linear clustering algorithm for grouping machines into manufacturing cells. The algorithm is based on a class of single linkage clustering methods and it also presents a method for reducing the number of inter-cell moves caused by the existence of exceptional elements. The methods that consider multiple objectives have been proposed by Venugopal and Narendran (1992a, b).

A mathematical programming approach to joint cell formation and operation allocation in cellular manufacturing was proposed by Atmani et al (1995). A zero-one integer programming model is formulated. The objective considered in this model is, to form machine groups and allocate operations in such a way as to minimize operation costs, re-fixture costs and transportation costs. Verma and Ding (1995) use a sequence-based procedure incorporating the costs of inter-cell flows and intra-cell flows. Cheng and Maden (1996) formed a model to minimize intercellular moves using distance as a measure. A truncated tree search algorithm has been presented by them.

Lee and Garcia-Diaz (1996) and Wu (1998) used a machine-machine relation matrix to calculate the inter-machine flows. ACCORD by Nair and Narendran (1999) considered intercell moves and within cell load variation as objective function Anita (2000) proposed a new part family identification using a simple genetic algorithm to determine a set of part family differentiating attributes and to guide the formation of part families. Adil Baykasoglu and Gindy (2000) made a study on multiple objective capability based approach to form part machine groups for cellular manufacturing application and called as MOCACEF. Hiroshi Ohta and Masateru Nakamura (2002) developed cell formation with the objective of reduction in setup times.

2.4 DISCUSSIONS

The majority of studies, particularly earlier studies, on cell formation focus on proposing efficient methods in terms of reducing exceptional elements and computational burden using zero-one MPIM. The major limitations of these approaches lie in the fact that real life production factors like operational time, sequence of operations, lot size of the parts etc. are not considered resulting in inefficient cells. However, some studies propose the methods considering production factors based on similarity coefficient (Seifoddini and Wolfe 1986, Vakharia and Wemmerlov 1990, Seifoddini and Hsu, 1994, Choobineh 1988, Nair and Narendran 1998), heuristics (Nair and Narendran, 1999, George et al. 2003, Vitanov et al. 2007, Iraj Mahdavi and Mahadevan 2008), metaheuristics (Boctor 1991, Venugopal and Narendran 1992a, b, Logendran et al. 1994, Jayaswal and Adil 2004). However, extremely limited number of studies has reported on cell formation using soft computing techniques when production factors have been considered. In order to consider two important production factors like operational time and sequence, ratio level and ordinal level data are used respectively as input to the cell formation algorithm. For the ratio-level data, workload information is commonly used and a modified incidence matrix is formed with this data. The total processing time of a part is computed by multiplying the production quantity of the part with its unit processing time. The workload (or ratio) value replaces '1's in the incidence matrix. The resultant

workload values can take any value in the ratio scale, and they constitute the ratio level data (George et al. 2003). If the resultant values can take any value in the ordinal scale, they constitute the ordinal level data. For example, operation sequence of the parts and batch size of the parts are ordinal level data. As soft-computing tools are efficient in cluster formation, investigation needs to be carried out for cell formation using production factors. The effects of parameters of soft-computing tools need to be established to provide guidelines to the users. Further, the performance measures to judge the goodness of cluster formation need to be redefined when production factors are to be considered. The performance measures existing for zero-one binary matrix are not appropriate in this case.

Therefore, it is felt that avenue exist for exhaustive research on application of soft-computing techniques for cell formation considering production factors. In chapter 3 and 4 the cell formation problem with operational time of parts is discussed and suitable methodology is proposed to solve the problem. Similarly, chapter 5 deal with cell formation with operational sequence and chapter 6 focuses on cell formation with operational time and sequence along with batch size. The appropriate performance measures are proposed to check the goodness of cluster formation.

Chapter – 3

Machine Cell Formation using Operational Time: A Neural Network Approach

3.1 INTRODUCTION

In light of the literature survey made in previous chapter, it is well understood that very few studies focus on cell formation considering production factors such as operational time, operational sequence, batch size etc. That too they are considered independently. In this work, it is attempted to consider some of the real time production factors individually and also by combining them. In this chapter, the zero-one MPIM of CF problem is converted into real valued workload data. The workload represents the operational time required by the parts in the machines. Soft computing technique is found more suitable for such type of problems and capable of producing good results. Soft computing is an innovative approach for constructing computationally intelligent systems (Sinha et al. 2000). Thereby, it is more appropriate to make use of soft computing techniques like neural network, fuzzy sets for cell formation problem with operational time. The ART1 algorithm with necessary modification is developed to form disjoint machine cells to handle the ratio level data. The methodology first allocates the machines to various machine cells and then the parts are assigned to respective machine cells with the aid of degree of belongingness through a membership index (Zolfaghari and Liang, 2003). The method of assignment of parts to various cells thus generated is discussed in section 3.4.1. The proposed algorithm uses a supplementary procedure to take care effectively of the problem of generating cells with a single machine that may be encountered at times.

3.2 THE ART 1 MODEL

The ANNs are massively parallelized computer systems that have the ability to learn from experience and adapt to new situations. The ANN consists of many units that represent neurons. Each unit is a basic unit of information process. The units are interconnected via links that contain weight values that help the neural network to express knowledge. The neural network is divided into three layers viz. input layer, hidden layer, and output layer as shown in Figure.3.1. The input layer transfers input signals into the hidden layer or the output layer. The hidden layer transfers signals from the input layer into the output layer and the output layer

gives the useful transformed signals. The units can be regarded as nodes that constitute the neural network. One node can receive signals from other nodes and transfer specific signal into other nodes. Neural networks theory adopts two types of training methods, viz. supervised and unsupervised learning.

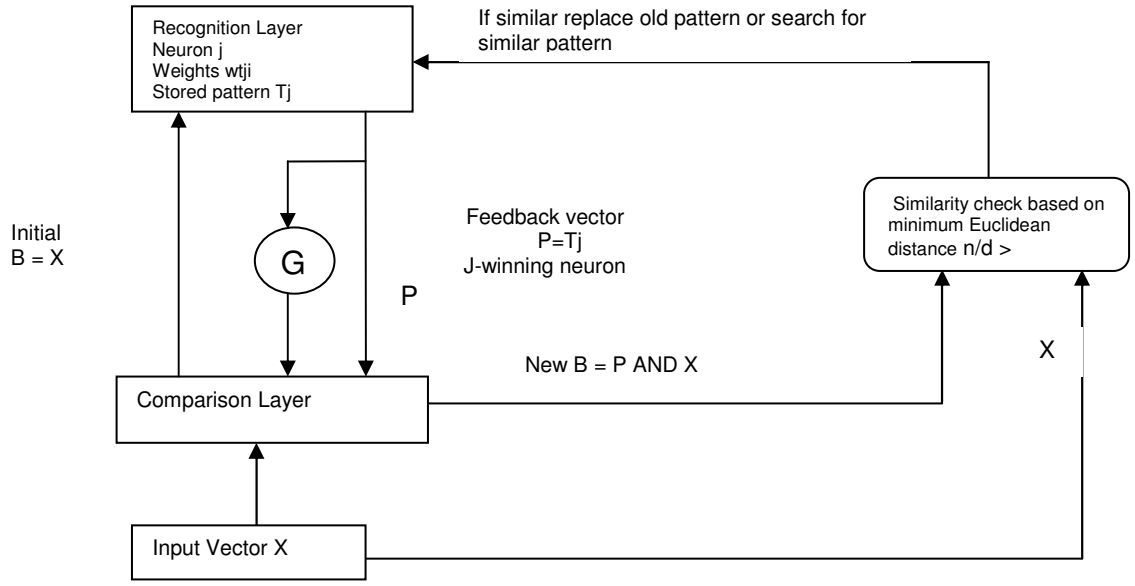


Figure 3.1 The ART1 Model

Supervised learning requires paring each input vector to a target vector and iteratively tunes the network. Unsupervised learning has no predetermined outputs. If an input vector is detected to be similar to a stored pattern, the weights representing the stored pattern are adjusted such that the stored pattern is more like the input vector (Jun wang and Yoshiyasu Takefuji 1993) The adaptive resonance theory (ART1), proposed by Carpenter and Grossberg (2002), is an example of unsupervised learning. Principles derived from an analysis of experimental literatures in vision, speech, cortical development, and reinforcement learning, including attentional blocking and cognitive-emotional interactions, led to the introduction of adaptive resonance as a theory of human cognitive information processing.

The foundation of ART1's stability is based on its matching criterion. The ART1matching criterion is determined by parameter ρ (vigilance parameter) that

specifies the minimum fraction of the input that must remain in the matched pattern for resonance to occur. Low vigilance allows broad generalization, coarse categories, and abstract memories where as high vigilance leads to narrow generalization, fine categories, and detailed memories (Carpenter and Grossberg 1987). If the network has learnt previously to recognize an input vector then a resonant state will be achieved quickly if similar input vector is presented. During resonance, the adaptation process will reinforce the memory of the stored pattern. If the input vector is not immediately recognized, the network will rapidly search through its stored patterns looking for a match. If no match is found, the network will enter a resonant state whereupon the new pattern will be stored for the first time. In this way, the network tries to respond quickly to previously learnt data. Kao and Moon (1991a) introduced back propagation neural network model for GT whereas Kaparthi and Suresh (1992) made an attempt to introduce adaptive resonance theory (ART1). Venkumar and Haq (2005) modified the ART 1 to apply for CF problem and come out with improved results. However, there are certain disadvantages of ART1 network viz. (i) it will recognize only the binary input data, and (ii) the resulting solution is highly influenced by the order of presentation of input vectors representing operation time T_{ij} which indicates that part j takes T_{ij} units of time to complete its operation in machine i. It is assumed that the lot size for all the parts is equal to one to characterize the behaviour of the sample problems considered in this chapter although it is not restrictive to one. If different lot sizes are considered then the processing times are multiplied with lot size to obtain the input workload matrix.

3.3 NEED FOR MODIFICATION IN ART1 APPROACH TO MACHINE CELL FORMATION

The basic idea in cellular manufacturing is to group the machines into machine cells and the parts into part families. The past research work reveals that the cell formation problems are addressed with zero-one binary incidence matrices in most cases. These approaches can hardly incorporate the real life production factors. The production data such as lot size of the parts, machine capacity, operational time and operation sequence need to be considered in order to generalize cell

formation problem. In the present work, an attempt is made to address the cell formation problem with operational time of the parts. Meta-heuristics like GA, SAA, TS and ACS seem to be prominent algorithms in giving good solutions to cell formation problems. Some models in neural networks are quite useful in clustering of both machines and parts with less computational time. In this chapter ART1 which is one among the neural network models is dealt with necessary modifications to consider real valued workload matrix to form machine cells and part families. When the input matrix is presented to the algorithm, the result is obtained in the form of a block diagonal structure where each block represents a cell. The rows and columns of the cell are machines and parts respectively assigned to the cell. The elements outside the block diagonal structure are termed as exceptional elements that represent inter-cell moves. In ART1 approach the formation of machine cells are natural, meaning that there is no constraint or objective function involved in the algorithm during clustering process. (Chapter 4 deals with the machine cell formation considering objective function).

The basic purpose of ART1 approach is to develop a simple and efficient methodology to provide quick solutions for shop floor managers with least computational efforts. However, there are certain disadvantages of ART1 network – (i) it will recognize only the binary input data. (ii) the resulting solution is highly influenced by the order of presentation of input vectors.

Chen and Cheng (1995) have successfully overcome the second disadvantage using some supplementary procedures. ART2 and self-organizing maps (Carpenter and Grossberg 1987) can overcome the first drawback of ART1. But simple network architecture, fast computation and outstanding ability to handle large scale industrial problems favour the choice of ART1 over other methods. The present work uses a modified ART1 to address the first disadvantage of the basic adaptive resonance theory.

3.4 THE MODIFIED ART1 ALGORITHM TO MACHINE CELL FORMATION WITH OPERATIONAL TIME

The algorithm given in section 3.4.1 is a modified version of ART1, adapted from the method proposed by Yoh-Han Pao (1989) that accommodates analogue patterns (matrix with ratio level data) instead of binary form of input vectors (conventional MPIM) for machine cell formation problem. The input to the algorithm is the workload matrix in which cell entries indicate the processing times. Let M be the total number of machines and N be the total number of parts then workload matrix size becomes M x N (for example 6 x 8 matrix as shown in Table 3.1a.

Table 3.1a MPIM of size 6x8 matrix (data set 4)

i/j	P1	P2	P3	P4	P5	P6	P7	P8
M1	0	1	0	1	0	0	1	0
M2	1	1	1	0	1	1	1	1
M3	0	0	1	0	0	1	0	1
M4	0	0	0	1	0	0	1	0
M5	1	0	1	0	1	1	0	1
M6	0	0	0	1	0	0	1	0

Table 3.1b MPIM matrix with real values

i/j	P1	P2	P3	P4	P5	P6	P7	P8
M1	0	0.53	0	0.99	0	0	0.83	0
M2	0.91	0.82	0.83	0	0.91	0.92	0.86	0.97
M3	0	0	0.79	0	0	0.56	0	0.88
M4	0	0	0	0.53	0	0	0.51	0
M5	0.98	0	0.83	0	0.71	0.58	0	0.54
M6	0	0	0	0.54	0	0	0.74	0

Table 3.1c Row sorted matrix

i/j	P1	P2	P3	P4	P5	P6	P7	P8
M1	0	0.53	0	0.99	0	0	0.83	0
M4	0	0	0	0.53	0	0	0.51	0
M6	0	0	0	0.54	0	0	0.74	0
M2	0.91	0.82	0.83	0	0.91	0.92	0.86	0.97
M3	0	0	0.79	0	0.00	0.56	0	0.88
M5	0.98	0	0.83	0	0.71	0.58	0	0.54

Table 3.1d Output matrix

i/j	P4	P7	P1	P2	P3	P5	P6	P8
M1	0.99	0.83	0	0.53	0	0	0	0
M4	0.53	0.51	0	0	0	0	0	0
M6	0.54	0.74	0	0	0	0	0	0
M2	0	0.86	0.91	0.82	0.83	0.91	0.92	0.97
M3	0	0	0	0	0.79	0	0.56	0.88
M5	0	0	0.98	0	0.83	0.71	0.58	0.54

3.4.1 Modified ART 1 algorithm

Step1: Initialize: Set nodes in the input layer equal to N (number of parts) and nodes in output layer equal to M (number of machines). Set vigilance threshold (ρ).

Step 2: Initialize top-down connection weights as given in equation (3.1)

$$wt_{ji}(0)=0 \qquad \textbf{(3.1)}$$

for i = 1, 2,...M. and j = 1,2,...N.

Step 3:Let q =1. The first input vector X_1 (first row of the workload matrix) is presented to the input layer and assigned to the first cluster. Then, first node in the output layer is activated.

Step 4:The top-down connection weights for the present active node are set equal to the input vector.

Step 5: Let $q = q + 1$. Apply new input vector X_q. (input vectors are the rows of the workload matrix).

Step 6: Compute Euclidean distance between X_q and the exemplar stored in the top-down weights (wt_{ji}) for all active nodes i as given in the equation (3.2). This distance function is used to calculate similarity between the stored pattern and the present input pattern. If the similarity value is less than or equal to ρ (vigilance threshold), the present input is categorized under the same cluster as that of stored pattern.

$$e_i = \sqrt{\sum_{j=1}^{N} (x_{qj} - wt_{ji})^2} \qquad \textbf{(3.2)}$$

Step 7: Perform vigilance test: Find out minimum Euclidean distance.

Step8: If min $e_i \leq \rho$ (threshold value), select output node for which Euclidean distance is minimum. If tie occurs, select the output node with lowest index number. Suppose output node k is selected then allocate the vector X_q to the node k (cell) and activate node k. Make increment to the number of machines in the active node k by one. If e_i s for all active nodes are greater than ρ, then go to step 9.

Step 9: Start a new cell by activating a new output node.

Step10:Update top-down weights of active node k using equation (3.3).

When a new vector is presented to the algorithm, its belongingness to existing nodes is judged by matching with respective top-down weights. The matching criterion is based on minimizing dissimilarity between existing exemplar stored as top-down weights and new input vector. Therefore, top-down weight updating principle warrants for storing combined information of previously stored exemplar and the present input pattern. Usually, higher weights are emphasised on stored exemplar than that of the new input vector. When a vector is selected (to be allocated to an output node), its top-down weights are updated using more

information of the previously stored exemplar and a relatively less information of the input vector (pattern) as shown in equation (3.3).

$$wt_{jk} = (\frac{n}{m} . wt_{jk}) + (\frac{1}{m} . x_{qj}) \quad \textbf{(3.3)}$$

Step 11: Go to step 5 and repeat till all the rows are assigned in the output nodes (cells).

Step 12: Check for single machine cells. If a single machine is found in any cell, perform the following operations to merge the single machine cells into any other cells.

1. **Determine average workload of each cell.**
2. **Calculate the Euclidean distance between the cells.**
3. **Merge a cell containing single machine with another in such a way that Euclidean distance between them is minimum.**

Step 13: Assign parts to cells using the membership index given in equation (3.4).

$$P_{kj} = \frac{f_{kj}}{f_k} \cdot \frac{f_{kj}}{f_j} \cdot \frac{T_{kj}}{T_j} \quad \textbf{(3.4)}$$

The membership index P_{kj} represents the belongingness of part j to cell k. Membership index consists of three components as shown in equation (3.4). First component (f_{kj}/f_k) denotes the proportion of machines of cell k required by part j. The second component (f_{kj}/f_j) is a ratio between the number of machines in cell k required by part j and the total number of machines required by part j. The third component (T_{kj}/T_j) is the proportion of processing time of part type j that can be accommodated in cell k. The belongingness of the part *j*, P_{kj} is calculated for all the cells, k = 1, 2, 3…C. Part j is assigned to cell k based on its maximum belongingness to cell k. The maximum belongingness can be calculated using equation (3.5).

$$P_m = max \quad \{P_{kj}\} \qquad k=1, 2, 3\ldots C. \tag{3.5}$$

The value of P_m lies between 0 to 1 where P_m = 1 indicates that the part *j* perfectly belongs to cell *k*.

3.5 MODIFIED GROUPING EFFICIENCY- A NEW PERFORMANCE MEASURE

Literature suggests that two popular measures viz. grouping efficiency and grouping efficacy are used to check the performance of block diagonal structure generated by a cell formation technique. Grouping efficiency is given by the equation (3.6). Chandrasekharan and Rajagopalan. (1986a) defined grouping efficiency as a weighted average of two functions η_1 and η_2 as shown in equations (3.7 and 3.8)

$$\eta = (r\, x\, \eta_1) + (1 - r)\eta_2 \tag{3.6}$$

where,

$$\eta_1 = \frac{Number\ \ of\ ones\ \ in\ the\ diagonal\ \ blocks}{Total\ \ number\ \ of\ elements\ \ in\ the\ diagonal\ \ blocks} \tag{3.7}$$

$$\eta_2 = \frac{Number\ \ of\ zeroes\ \ in\ the\ off\ -\ diagonal\ \ blocks}{Total\ \ number\ \ of\ elements\ \ in\ the\ off\ -\ diagonal\ \ blocks} \tag{3.8}$$

r is a weighting factor that lies between zero to one ($0 \leq r \leq 1$) and its value is decided depending on the size of the matrix. Grouping efficiency considers two functions - packing density inside the cells (η_1) and inter-cell moves (η_2). Weighting factor is used to achieve a trade off between two functions depending on desirability of the decision maker. A higher value of η is supposed to indicate better clustering.

The first drawback of grouping efficiency is its low discriminating capability i.e. less ability to distinguish a good quality solution from a bad quality solution. A solution with many 1's (ones) in the off-diagonal blocks shows higher efficiency (ranges from 75% to 100%), which intuitively must show lower efficiency. Secondly, emphasis on number of zeros in the off-diagonal blocks rather than number of 1's in equation (3.7) invariably leads to calculate a higher efficiency.

This phenomenon is more closely observed when number of exceptional elements decreases with increase in size of the matrix. Therefore, it can be ascertained that grouping efficiency is highly sensitive to the size of the matrix. To overcome these shortcomings, grouping efficacy was proposed by Kumar and Chandrasekharan (1990) as given in equation (3.9). The emphasis on number of ones outside the diagonal blocks and the number of zeros inside the blocks are given in equations (3.10 and 3.11)

$$\tau = \frac{(1-\psi)}{(1-\varphi)} \qquad \textbf{(3.9)}$$

where,

$$\psi = \frac{Number\ of\ exceptional\ elements}{Total\ number\ of\ operations} \qquad \textbf{(3.10)}$$

$$\varphi = \frac{Number\ of\ voids\ in\ the\ diagonal\ blocks}{Total\ number\ of\ operations} \qquad \textbf{(3.11)}$$

Unlike grouping efficiency, grouping efficacy is not affected by the size of the matrix. However, both measures - grouping efficiency and grouping efficacy treat all operations equally and suitable only for the zero-one incidence matrix. These measures cannot be adopted for cell formation problem where information regarding operational times is of importance. Therefore, generalized grouping efficacy introduced by Zolfaghari and Liang (2003) can be conveniently used to measure the performance considering operational times of the parts. But in contrast to grouping efficiency and grouping efficacy measures, generalized grouping efficacy ignores the effect of voids inside cells, which predominantly affects the goodness of the block diagonal structure. Hence, a new measure for grouping efficiency termed as modified grouping efficiency (MGE) has been introduced in this chapter to find out the performance of the cell formation method that deal with workload matrix with due consideration of voids inside the cells.

For the cell formation problems using workload (operational time) information, the grouping efficiency has to be found out from the ratio of total workload inside the cells denoted as T_{pti}, and total workload of the matrix. When total workload is being calculated, the number of voids present inside the cells are taken into account and the proportionate value of voids with the number of elements present inside the cells are calculated using the weighting factor to the voids ratio as shown in the equation (3.12). The elements outside the cells represent exceptional elements, denoted as T_{pto}. The MGE is calculated using equation (3.13).

$$w_v = N_{vk} \; / \; N_{ek} \qquad \textbf{(3.12)}$$

$$MGE = \frac{T_{pti}}{T_{pto} + \sum_{k=1}^{c} T_{ptk} + \sum_{k=1}^{c} T_{ptk} \; .w_v} \qquad \textbf{(3.13)}$$

Unlike grouping efficiency, modified grouping efficiency does not treat all the operations equally. Moreover a weighting factor for voids is considered to reflect the packing density of the cells. It produces 100% efficiency when the cells are perfectly packed without any voids and exceptional elements.

3.6 AN ILLUSTRATIVE EXAMPLE

The binary matrix of size 6×8, shown in Table 3.1(a), is converted into real valued workload matrix by replacing the ones with uniform random numbers in the range of 0.5 to 1 and zeros remain unchanged in the same position. The resultant matrix shown in Table 3.1 (b) is presented as input to the ART1 algorithm. Initially the algorithm assigns the machines (rows) to the cells and the row sorted matrix is given in Table 3.1 (c). After rows are sorted out, parts (columns) are assigned to the cells using the membership index given in the equation (3.4) and equation (3.5) to form the part families. Thus final solution matrix obtained after sorting the columns has two cells as shown in Table 3.1(d). The first cell does not have any voids and hence weighting factor for the voids (w_v) is zero. The number of voids in second cell (N_{vk}) is four and total number of elements (N_{ek}) is eighteen. Therefore,

weighting factor for voids (w_v) for the second cell equals to 0.2222. The total processing time in second cell (T_{ptk}) is 11.234 and it is multiplied by its weighting factor for voids to produce T_{ptk} x w_v equal to 2.496. The total processing time inside both the cells (T_{pti}) is 15.379. As total number of exceptional elements is two, sum of their value (T_{pto}) is 1.385. The summation of (T_{pto}), (T_{pti}) and (T_{ptk} x w_v) is calculated as 19.2604. Finally, the value of MGE can be expressed as the ratio of 15.379 and 19.2604 and it is found to be 79.85%.

3.6.1 Eliminating cells with single machine

A cell with a single machine is not desirable in cellular manufacturing because its expected advantages, in particular flexibility, will be lost. The algorithm sometimes generates cells with single machine. A procedure has been illustrated to deal with single machine cell with the help of an example problem of size 30 x 50 (data set 24). The matrix is presented to the algorithm already described in section 3.4.1 to produce row sorted and column sorted matrices and the output matrix are given in Appendix IV. Initially the configuration has 10 cells comprising of 19, 1, 1, 2, 1, 2, 1, 1, 1, and 1 number of machines in respective cells (from 1 to 10). There are seven cells with single machine in this configuration. The algorithm works in such a way that the machines numbered 2, 3, 10, 17, 19, 20, and 21, each representing a single machine cell, are combined with other cells. The process of combining cells is based on Euclidean distance of average workloads in the cells. The Euclidean distance for average workload between the cell with single machine and all other cells is found out. The cell that shows minimum value of Euclidean distance as shown in Table 3.2 with the cell containing single machine can be combined together to form one cell.

Initially, there were ten cells and after eliminating single machine cells, the final output has only three cells.

1. Machines numbered 2, 10, 17, 19, 20 and 21 are merged with cell-1 having 19 machines.
2. Machine numbered 3 is merged with machines numbered 8 and 25.
3. There is no change in the cell with machines numbered 15 and 28.

   The new configuration has 3 cells as shown in Appendix IV.

Table 3.2 Comparison vectors (the highlighted numbers show minimum Euclidean distance in the cells having singleton machine) (data set 24)

Cell No	1	2	3	4	5	6	7
1	0.00	2.45	4.02	4.44	2.94	5.67	4.11
2	2.45	0.00	4.51	4.58	4.14	5.81	4.24
3	4.02	4.51	0.00	6.50	2.36	7.74	6.17
4	4.44	4.58	6.50	0.00	6.14	3.20	3.49
5	2.94	4.14	**2.36**	6.14	0.00	5.64	5.80
6	5.67	5.81	7.74	**3.20**	5.64	0.00	3.23
7	4.11	4.24	6.17	**3.49**	5.80	3.23	0.00

3.7 RESULTS AND DISCUSSIONS

The ART 1 is adapted in this chapter since in recent years researchers prove that soft computing techniques made successful contributions in the CF problems (Venugopal 1999). Besides that there are very few methods developed so far which clusters simultaneously machines and parts while considering important real life production factors such as operational time and operational sequence (George et al. 2003). Considering these issues, in this chapter the ART 1 is suitably modified to handle such production factors and form both clusters concurrently thereby identifying machine cells and parts families for constructing CMS. The following major contributions are made in this chapter.

- Suitable modification is made in the ART1 to handle operation time (ratio level data) to form machine cells and part families concurrently.
- Procedure for eliminating cells containing single machine and merging them into the cells with minimum threshold value is given.
- Appropriate performance measures are developed to consider operational time of the parts.

. The proposed algorithm is tested with a wide variety of benchmark problems from open literature and is found to be consistent in producing quality solutions.

Table 3.3 shows the problems of different sizes selected from open literature with their sources for testing the algorithm. For each sample problem, the workload

Table 3.3 Performance of the proposed modified ART1

Data set No.	No. of Cells	K-means algorithm		C-linkage algorithm		Proposed modified ART1	
		EE	MGE %	EE	MGE %	EE	MGE %
1	2	2	77.25	2	77.25	2	77.25
2	2	2	78.34	2	78.34	2	78.34
3	2	7	81.87	7	81.87	7	81.87
4	2	2	79.85	2	79.85	2	79.85
5	2	3	61.77	3	61.77	3	61.77
6	2	1	65.48	1	65.48	1	65.48
7	2	6	57.00	6	57.00	4	69.70
8	2	28	60.00	28	60.00	25	61.30
9	3	9	83.40	9	83.40	9	83.40
10	3	0	77.14	0	77.14	0	77.14
11	3	0	93.28	0	93.28	0	93.28
12	2	2	59.43	2	59.43	2	60.59
13	4	7	68.13	9	65.23	2	76.13
14	3	15	64.81	15	64.81	15	64.81
15	2	42	49.13	42	49.13	19	60.10
16	3	1	71.00	1	71.00	1	71.15
17	4	31	61.50	31	61.50	28	61.71
18	3	38	51.70	38	51.70	42	50.50
19	4	34	46.70	30	51.39	30	51.39
20	6	0	90.28	0	90.28	0	90.28
21	5	7	71.60	7	71.60	9	73.89
22	3	12	56.65	17	53.98	17	53.98
23	6	20	61.84	20	61.84	26	55.51
24	3	33	50.51	33	50.51	17	53.19

(input) matrix is generated by replacing the ones in the incidence matrix with uniformly distributed random numbers in the range of 0.5 to 1 and zeros to remain in same positions. Different sizes of the problem range from 5 x 7 to 30 x 50 have been considered. It is assumed that the lot size for all the parts equal to one to characterize the behaviour of the sample problems considered in this chapter although it is not restrictive to one**.** In order to evaluate the performance of the ART1 algorithm, the sample problems are tested with two more algorithms viz. K-means clustering (Cheung 2003) algorithm and C-linkage algorithm (Defays 1977). The algorithm is coded in C^{++} and run on an IBM Pentium IV PC with 2.4 GHz Processor.

K-means clustering and C-linkage clustering algorithms are used for comparison since literatures suggest these algorithms are comparable in clustering. Moreover Euclidean distance can be used in both K-means and C-linkage clustering algorithms for finding out the nearness between clusters. They assign machines to different clusters using Euclidean distance. The number of clusters and number of iterations are varied depending on the size of the problem until no further improvement is possible in the solution.

It is observed that the number of iterations lies in the range of 20-35 for the sample problems considered in this work. The standard software SYSTAT.11.0 is used to form clusters using K-means and C-linkage algorithms. The solutions obtained by the proposed modified ART1 algorithm are compared with the solutions of K-means and C-linkage clustering algorithms.

The computational time required to obtain solution in modified ART1 is reported for few sample problems in Table 3.4.

Table 3.4 CPU Time for the proposed method

S.N	Problem Size	CPU Time (Sec) Modified ART 1
1	5 x 7	0.060213
2	16 x 30	0.396320
3	30 x 50	1.854945

Careful observation of Table 3.3 reveals that number of exceptional elements reduces with modified ART1 algorithm for most of the sample problems as compared to solutions obtained by two other methods. As far as MGE is concerned, solutions obtained by the modified ART1 algorithm outperform other two methods in most of the tested problems. K-means and C-linkage clustering methods produces comparatively inferior results in respect to performance measures like MGE and number of exceptional elements. However, all the three methods are equally good when the problem size is small. Modified ART1 provides desired solution in a single iteration whereas both K-means and C-linkage methods require multiple numbers of iterations for any size of the problem. The advantage of modified ART1 lies in its ability to generate quality solution for large size problems.

In modified ART1, the vigilance threshold value greatly influences the number of cells obtained. It is used to tune the algorithm and also to create the desired number of cells as required by the decision maker. For example the threshold value of 2 makes 5 cells whereas the threshold value of 2.5 creates only 3 cells for the problem of size 16 x 30.

The threshold value for each problem is varied from 1.5 to 2.5. It is observed that the number of cells equals to the total number of machines if the threshold value is set at zero. As the threshold value increases, the number of cells can be reduced as shown in Table 3.5.

The threshold value for illustrated example of size 6 x 8 is 2.00 and the modified grouping efficiency (MGE) obtained by the modified ART1 algorithm is 79.85%. Both K-means algorithm and C-linkage algorithms also produce the same value of MGE for the illustrated example. The supplementary procedure described in step 12, section 3.4.1 can be used to avoid cells with single machine that is encountered at times. The algorithm is flexible in the sense that maximum number of machines to be accommodated in a cell can be limited. The modified grouping efficiency given in this chapter is evidently suitable to measure the performance of cell formation algorithm taking into account workloads on machines, weighting factor for voids, and exceptional elements.

The ART1 algorithm is tested with different size problems from open literature and resulting solutions are compared with the solutions obtained from K-means and C-linkage clustering methods. In most of the problems, it is observed that the solutions obtained by the ART1 algorithm either outperform existing methods or

Table 3.5 Effect of threshold value on number of cells for the problem size 16 x 30

S.N	Threshold Value (ρ)	Number of cells	Machines allocated	Parts allocated
1	2	Cell –1	0 3 6 7 10 11	1 3 6 8 11 17 21 29
		Cell –2	1	0 9 15
		Cell –3	2 5 8 14	4 18 22 24 26 27 28
		Cell –4	4 9 13 15	5 7 10 13 14 16 20 23 25
		Cell –5	12	2 12 19
2	2.3	Cell –1	0 3 6 7 10 11	1 3 6 8 11 17 21 29
		Cell –2	1	0 9 15 19
		Cell –3	2 5 8 12 14	2 4 12 18 22 24 26 27 28
		Cell –4	4 9 13 15	5 7 10 13 14 16 20 23 25
3	2.5	Cell –1	0 3 6 7 10 11	1 3 6 8 11 17 21 29
		Cell –2	1 2 5 8 12 14	0 4 9 12 15 18 19 22 24 26 27 28
		Cell –3	4 9 13 15	2 5 7 10 13 14 16 20 23 25

remain the same as far as number of exceptional elements and MGE are concerned. The effect of vigilance threshold to the number of cells is shown in Figure 3.2. Since the algorithm uses simple network architecture, it helps to reduce computational burden compared to K-means and C-linkage clustering methods. Therefore, the modified ART1 is found to be computationally efficient for generating quick solutions for industrial applications.

It has been found from extensive experimentation that the modified ART1 algorithm is sensitive to the order of presentation of the input vectors due to decaying of the stored template leading to unsystematic weight updating. Therefore, it may lead to produce different solution if the order of presentation of the input vectors is changed. Few methods have been suggested by Suresh and Kaparthi (1994) are available to address this limitation when zero-one incidence matrix is used as input. However, no effective method exists to address this limitation if workload matrix is dealt (Chen and Cheng, 1995).

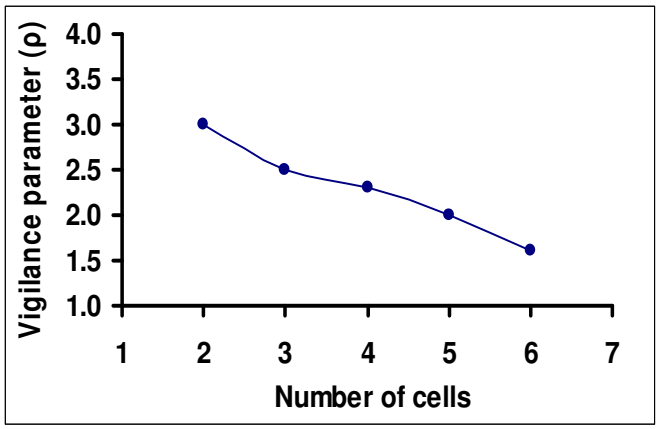


Figure 3.2 Vigilance parameter vs Number of cells (operational time)

Chapter – 4

Genetic Cell Formation with Operational Time

4.1 INTRODUCTION

In the previous chapter, it has been discussed about natural clustering using ART1 approach that the formation of machine cells is done without considering any objective function or constraints for grouping. In this chapter, the following objectives are considered for the formation of machine cells: (i) Total cell load variation, (ii) exceptional elements and (iii) combination of total cell load variation and exceptional elements. Uniform workload distribution among the cells is an important aspect of CMS because non-uniform workload distribution among the cells may give rise to increase in work-in-process inventories and lead time. However, this vital aspect is not adequately addressed in the literature. Venugopal and Narendran (1992a) have proposed a minimization multi-objective function, which is a combination of cell load variation and number of inter-cell moves, where the rationalization of two objectives is highly desirable to bring both the objectives into same scale.

Cell load variation and number of exceptional elements are heavily dependent on number of cells. Increasing number of cells leads to increase in number of exceptional elements and reduce cell load variation whereas decreasing number of cells causes to lower the number of exceptional elements and increase cell load variation. Because of the conflicting nature of two objectives, in this research a combined objective function is considered with different weights assigned to each of the normalized objectives. The efficiency of the layout is also affected unless a systematic procedure for part allocation is adopted. Therefore, structured method of part allocation into cells is followed in this study.

In this chapter, an attempt has been made to address the following issues:

(i) Minimization of total cell load variation so that smooth flow of parts, reduction of work-in-process inventories and lead time can be achieved.

(ii) Minimization of exceptional elements.

(iii) Minimization of the combined objective (i.e.) total cell load variation and exceptional elements bringing them to a normalized scale with weighted sum approach.

(iv) Uniform allocation of parts following a patterned procedure, which also aids in achieving distribution of loads uniformly to machine cells and reduce the exceptional elements.

Modified grouping efficiency (equation (3.13)), is used to find out the grouping performance with operational time.

Genetic Algorithm (GA), a popular evolutionary technique, is suggested for machine cell formation with the objective of minimizing both the total cell load variation and the exceptional elements. The cell load variation is calculated as the difference between workload on the machine and average load on the cell. The exceptional elements are found out by counting the number of non-zero values in off diagonal blocks. It has been demonstrated that GA provides an efficient search to find quality solutions.

Since cell formation problems are known as NP complete optimization problems, it is difficult to obtain generalized solutions. Modern meta-heuristics like GA and SA seem to be prominent algorithms to be tested for solution quality when applied to cell formation problems. Nowadays multi-objective models are considered taking all benefits of CMS into account, for example, ACCORD by Nair and Narendran (1999) GGA by Yasuda et al. (2005) As GA has been tested successfully in cell formation problems, the methodology uses GA in a wide range of problem sizes.

4.2 SOFT COMPUTING TECHNIQUES FOR MULTI OBJECTIVE CELL FORMATION

Soft computing techniques seem particularly suitable to solve multi-objective optimization problems, because they are less susceptible to the shape or continuity of the Pareto front (e.g., they can easily deal with discontinuous or concave Pareto fronts), whereas this is a real concern for mathematical programming techniques. Additionally, many current soft computing techniques (e.g., GA, SA, particle swarm optimization, fuzzy set theory etc.) are population-based, that several elements of the Pareto optimal set in a single run can be

generated. Goncalves and Resende (2002) provided hybrid GA for CF problem. Dimopoulos (2006) has developed a multi-objective genetic programming, an evolutionary computation methodology for the solution of the multi-objective CF problem.

4.3 PROBLEM FORMULATION

Several objectives like inter-cell and intra-cell moves, grouping efficiency and exceptional elements are associated with machine grouping problem as found in literature. But all these objectives hardly reflect smooth flow of material leading to high work-in-process inventories. In order to achieve smooth flow of materials leading, less work-in-process inventories and increased productivity, cell load variation must be considered. In section 4.4.1 and 4.4.2 total cell load variation and exceptional elements are considered respectively. In section 4.4.3 the two objectives both total cell load variation and exceptional elements are combined and formulated as a multi objective minimization problem. In reality both the objectives are conflicting to each other in giving good solutions. The number of cells greatly influences the combined objective function since increase in the number of cells increases exceptional elements, where cell load variation decreases and vice versa. Because of this conflicting nature of both the objectives, a combined objective function is considered giving different weightings q_1 and q_2.

4.3.1 Total cell load variation

The visit of the parts to the machines has been denoted in terms of their workload on the machines for the computation of cell load variation. The cell load variation is calculated as the difference between the workload on the machine and the average load on the cell (Venugopal and Narendran, 1992a). It is expressed in equation (4.1)

$$Z_1 = \sum_{i=1}^{M} \sum_{k=1}^{C} X_{ik} \sum_{j=1}^{N} (W_{ij} - m_{kj})^2 \qquad \textbf{(4.1)}$$

4.3.2 Exceptional elements

The exceptional elements are found out by counting the number of non-zero values in off diagonal blocks as given in equation (4.2).

$$Z_2 = \frac{1}{2} \sum_{k=1}^{C} \sum_{j=1}^{N} \sum_{i=1}^{M} |X_{ik} - Y_{jk}| a_{ij} \quad \textbf{(4.2)}$$

4.3.3 Combination of total cell load variation and exceptional elements

In the objective of our work, two functions are combined giving equal weightage. The first function is total cell load variation and the second is exceptional elements. The total cell load variation is expressed in real numbers whereas exceptional elements in integers. Normally, the numerical value of cell load variation is higher compared to number of exceptional elements. Therefore, extend of impact of the exceptional elements is not reflected adequately in the result. Therefore, it is essential to normalize both functions in a uniform way and both functions must be measured in the same scale. In our work, both cell load variation and exceptional elements are normalized to give values between zero and one so that the combined function can be easily interpreted reflecting effect of both the objectives. Thus the objective function of the cell formation problem is formulated as shown in equation (4.3). The first component expanded represents the ratio between square root of the total cell load variation and the total workload of the matrix. The second component indicates the ratio of number of exceptional elements and the total elements of the matrix. In view of equal importance for both the objectives, the values of the weights are assigned to 0.5 to both q_1 and q_2.

Minimize

$$Z = q_1 \left(\frac{\sqrt{Z_1}}{T_w} \right) + q_2 \left(\frac{Z_2}{T_o} \right) \quad \textbf{(4.3)}$$

To characterize the behaviour of the algorithm, it is mandatory to fix the weightages. However, equal weightage is normally preferred to demonstrate the effectiveness of the algorithm. But it can be varied depending on choice of

decision makers. However, limiting condition is that the sum of the weightages must be equal to one $q_1+q_2=1$.

4.4. THE PROPOSED METHODOLOGY

Genetic Algorithm (GA) is adopted to find out the machine clusters to form cells. In GA a candidate solution represented by sequence of genes called chromosome. A chromosome potential is called its fitness function, which is evaluated by the objective function. A set of selected chromosomes is called population and the population is subjected to generations (number of iterations). In each generation crossover and mutation operators are performed to get new population. A brief introduction about GA is given in Appendix V.

4.4.1 Representation

Representation is made in the form of solution string (t). In this problem considered, each gene represents cell number and its position gives machine number. As shown in Figure 4.1, the machine number 1, 3 and 5 are in cell number 2, the machine number 2 and 4 are in cell number 1.

4.4.2 Reproduction

A fitness function value is computed for each string in the population and the objective is to find a string with the maximum fitness function value. Since objective is minimization it is required to map it inversely and then maximize the resultant. Goldberg (1989) suggested a mapping function given as

F (t) - fitness function of t^{th} string [$F(t) = Z_{max} - Z(t)$]

Z_{max} - max [Z (t)] of all strings

The advantage is that the worst string gets a fitness function value of zero and there is no chance of the worst string getting reproduced into the next generation.

4.4.3 Crossover and Mutation

The crossover operator is carried out with a probability known as crossover probability. Crossover is exchange of a portion of strings at a point called crossover site (S). The two strings, which take part in the crossover operation, are

also selected at random. Here partial mapped crossover given by Michalewicz (1996) is performed i.e., crossover site is selected and the genes of one string

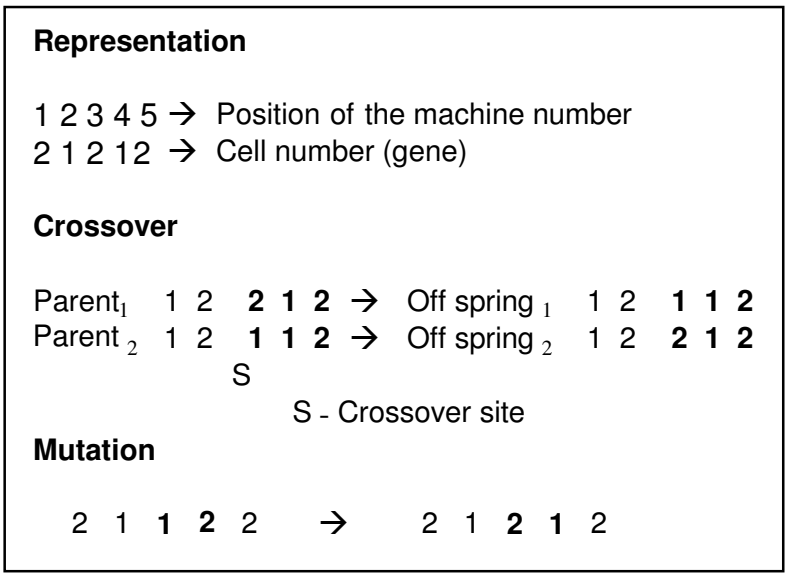


Figure 4.1 Genetic operators

between the sites are swapped with genes of another string as shown in Figure 4.1. Mutation is also done randomly for each gene and it depends upon another parameter called mutation probability. In this method *inversion mutation* is adopted where one gene is selected at random, comes out from one cell and goes to another cell, while a machine from latter cell comes to the former cell as shown in Figure 4.1.

4.4.4 Part Assignment

The following procedure given by Chen and Cheng (1995) is used to assign parts into the machine cells. A machine cell, which processes the part for a larger number of operations than any other machine cell, is found out and the corresponding part is assigned into that cell. Ties are broken by choosing the machine cell that has the largest percentage of machines visited by the part. In the case of tie again the machine cell with the smallest identification number is selected. Thus all the parts are assigned to all the cells, which form part families.

4.5 PROPOSED GA BASED ALGORITHM

I. Initialization

Step 1: Set the values of P_s, gen, P_c, P_m.

Step 2: Read the workload given in terms of processing time W_{ij} of part j on machine i.

Step 3: Create an initial population of size Ps and call it old population (P_{old}).

Step 4: Calculate the objective function using equation (4.3).

Step 5: Sort string in the increasing order of objective function value.

Step 6: Set gen = 0.

II. Reproduction

Step 1: Compute F(t) for $P_{old.}$

Step 2: Compute P_t of each string.

Step 3: Find the cumulative of $P_{t.}$

Step 4: Generate 'r_a' and select the string from P_{old} according to r and reproduce it in $P_{new.}$

Step 5: Repeat step 4 for P_s time.

Step 6: End.

III. Crossover

Step 1: Generate 'r' if ($r<P_c$) go to step 2 else go to step 4.

Step 2: Select two strings t1 and t2 and swap genes between them by selecting crossover site S randomly.

Step 3: Repeat step 2 for $P_s/2$ times.

Step 4: End.

IV. Mutation

Step 1: Generate 'r_a'.

Step 2: If ($r_a < P_m$) go to step 3 else go to step 1.

Step 3: Select two machines randomly in t and interchange its positions.

Step 4: Repeat step 1 for all genes in $P_{new.}$

Step 5: End.

V. Part Assignment

Step 1: **Find a machine cell that processes the part for a larger number of operations than any other machine cell and assign the part in that machine cell.**

Step 2: **If tie occurs, choose the machine cell that has the largest percentage of machines visited by the part and assign in that cell.**

Step 3: **If again tie occurs, select the machine cell with the smallest identification number and assign the part in that machine cell.**

Step 4: **End.**

VI. Main Algorithm

Step 0: Define the number of cells c = k. (k = 2,3,…,m)

Step 1: Initialize the values and evaluate the objective function as given in section I.

Step 2: Do Reproduction as given in section II.

Step 3: Do Crossover as given in section III.

Step 4: Do Mutation as given in section IV.

Step 5: Do Part Assignment as given in section V.

Step 6: Increment counter.

Step 7: If (counter < gen) go to step 2 else step 11.

Step 8: Store the objective value in Z. Go to step 0. k=k+1.

Step 9: Print the best value of Z.

Step 12: Stop.

4.6 CONVERGENCE

The data set no.25 of size 5 x 8 is taken as an example to illustrate the convergence curve during iterations. For the first iteration the objective value (Z × 100) is to be 51.77. It gets reduced when the number of generation increases. At 10^{th} generation it reached to the value of 30.35, a reduction of 41.37%. During 25^{th} generation the Z value (× 100) is 5.6, a reduction of 81.67%. The value remains same for further increase in the iterations. So it is terminated at this point of time. The convergence curve is shown in Figure 4.2.

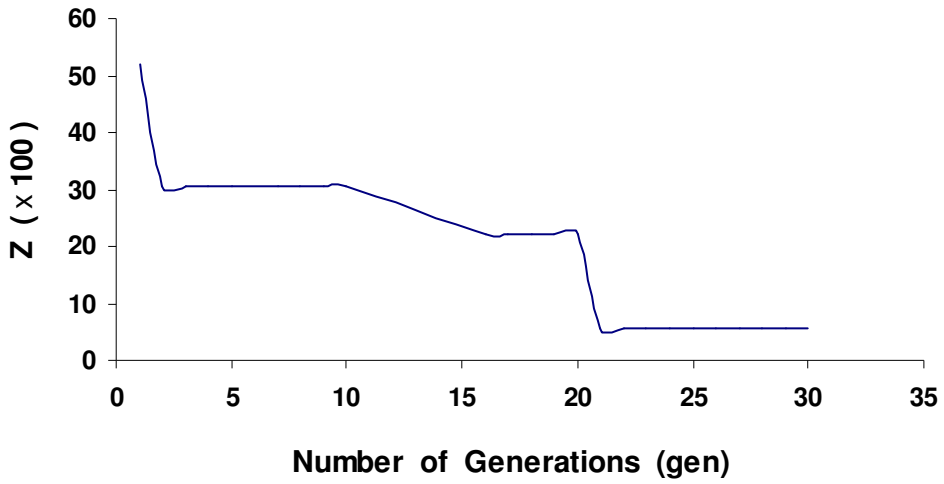


Figure 4.2 Convergence curve

4.7 RESULTS AND DISCUSSIONS

The algorithm is coded in C++ and run on Pentium IV PC, 2.4GHz processor. The real valued matrix is produced by assigning random numbers in the range of 0.5 to 1 as uniformly distributed values by replacing the ones in the incidence matrix and zeros to remain in its same positions. The model developed using GA has been tested with 25 benchmark problems of varied sizes ranging from 5 x 8 to 24 x 40 from open literature and the results are compared with K-means clustering and C-link clustering algorithms, given in Table 4.1, confirm that GA is an appropriate solution methodology to such type of optimization problems. Based on exhaustive experiments, the crossover and mutation probabilities are fixed to be 0.5 and 0.1 respectively. This probability can be varied depending upon the decision maker to tune the algorithm. The chromosome representation used in this study may result in the formation of an empty cell or violates some constraints. Particularly, crossover may result in the formation of a chromosome like 113331 when predefined number of cells is three. The above chromosome contains an empty cell where cell number 2 is missing. In such cases, the respective chromosomes are rejected. Crossover and mutation steps are repeated with other pairs of chromosomes till a useful chromosome is obtained.

Table 4.1 Comparison of results of proposed GA with K-means and C-link methods.

Data Set No.	Problem Size	Number of Cells	K-means		C-link		GA	
			Z (×100)	MGE %	Z (×100)	MGE %	Z (×100)	MGE %
25	5 x 8	2	5.60	100.00	5.60	100.00	5.60	100.00
26	7 x 11	2	15.52	63.42	15.52	63.42	15.52	63.42
27	8 x 20	2	18.38	59.74	18.38	59.74	18.38	59.74
28	8 x 20	2	7.86	72.11	7.86	72.11	7.86	72.11
29	9 x 9	2	11.82	73.25	11.82	73.25	11.82	73.25
30	10 x 15	2	4.42	72.19	4.42	72.19	4.42	72.19
31	8 x 14	3	1.65	100.00	1.65	100.00	1.65	100.00
32	9 x 10	3	1.72	100.00	1.72	100.00	1.72	100.00
33	12 x 31	3	17.94	53.61	17.12	55.06	17.12	55.06
34	16 x 30	3	14.25	57.75	12.16	59.84	12.16	59.84
35	16 x 30	3	6.33	68.55	6.33	68.55	6.33	68.55
36	16 x 30	3	7.60	67.89	7.60	67.89	7.60	67.89
37	16 x 30	3	16.50	53.38	16.03	52.71	15.98	54.69
38	16 x 30	3	7.15	70.05	7.63	68.99	7.15	70.05
39	16 x 30	3	5.51	69.73	5.51	69.73	5.86	70.91
40	16 x 30	3	6.31	71.50	6.31	71.50	6.31	71.50
41	16 x 30	3	9.91	65.18	14.25	59.71	9.91	65.18
42	16 x 30	3	7.76	71.78	7.76	69.49	7.76	71.78
43	16 x 30	3	6.71	68.87	9.75	65.70	7.82	67.26
44	16 x 43	3	17.09	53.47	18.99	44.62	17.09	53.47
45	10 x 20	4	3.74	96.40	3.74	96.40	3.74	96.40
46	11 x 16	4	3.81	98.11	3.81	98.11	3.81	98.11
47	16 x 43	4	19.45	53.41	19.45	53.41	19.45	53.41
48	24 x 40	4	21.11	46.37	16.82	48.67	17.93	49.15
49	24 x 40	5	9.20	67.29	6.97	67.55	9.54	67.62

The number of cells greatly influences the objective function Z since increase in the number of cells decreases cell load variation and increases exceptional elements. Hence it rests with the decision maker to trade off between the objectives and choose the value of the weighting factors accordingly. In this work the weighting factor is assumed to be 0.5 to provide equal importance to both the objectives. The number of generations is varied from problem to problem in the range of 50 to 1000. Similarly, the population size is varied in the range of 10 to 40 depending on the size of problem. For instance, the optimal solution has been arrived with the population size of 20 in 103 iterations for the problem of size 10x15 where as the problem of size 30x41 is solved with the population size of 15 with 593 iterations to reach optimal solution. The convergence property of GA is given in Figure 4.2.

The proposed algorithm gives same results as that of K-means and C-linkage algorithms when the problem size is small, if the problem size increases the GA outperforms other methods. The time taken for the problems varies with the size of the problem, population size and number of generations, which is given in Table 4.2. MGE (equation (3.13)) is used to assess the performance of the grouping with real values. This measure is capable of judging the goodness of block diagonal structure as it takes care of both voids and exceptional elements into consideration. The results of proposed algorithm are compared with the results obtained from K-means clustering algorithm and C-link algorithm based on MGE. Problems have been tested by varying the number of cells from 2 to 5 depending on the total number of machines. Since the objective function is a combination of exceptional elements and cell load variation, it depends upon the decision maker to make a trade off between two conflicting objectives by preferring weight for each objective. In some cases, particularly in data set numbers 39 and 49, the value of objective function is more in case of GA compared to other two algorithms but block diagonal structure is always better in case of GA. Finally the proposed GA is compared with modified ART1 as given in chapter 3. In Genetic cell formation, GA attempts to allocate the machines based on minimizing total cell load variation. In the process of cell load distribution, sometimes number of

exceptional elements may be more as compared with the solutions obtained from other methodologies.

This is the reason that in few problems GA produces comparatively inferior results. For instance the solutions obtained in data set 13 (Table 4.3) the modified ART1 gives better results. The reason is that the modified ART1 attempts to form cells based on the attributes. If similar attributes are found within machines, they are grouped in one cluster. Besides that the methodology of clustering exclusively differs from GA. As far as MGE is concerned GA performs better than modified ART 1, but for faster computational effort modified ART 1 seems better than that of GA which is inferred from Table 4.4

Table 4.2. CPU Time for the proposed GA

S.N	Problem Size	Population Size (nos)	No. of Generations	CPU Time (sec)
1	10 x 15	20	103	0.21978
2	12 x 31	25	248	0.54945
3	16 x 43	20	494	1.26373
4	24 x 40	25	853	3.62637
5	30 x 41	15	593	2.19780

Table 4.3 Comparison of results of GA with modified ART 1

Data Set No.	No of cells	Modified ART1		GA	
		No of EE	MGE %	No of EE	MGE %
1	2	2	77.25	2	77.25
2	2	2	78.34	2	78.34
3	2	7	81.87	7	81.87
4	2	2	79.85	2	79.85
5	2	3	61.77	3	61.77
6	2	1	65.48	1	65.48
7	2	4	69.70	6	69.70
8	2	25	61.30	28	61.30
9	3	9	83.40	9	83.40
10	3	0	77.14	0	77.14
11	3	0	93.28	0	93.28
12	2	2	60.59	0	62.42
13	4	2	76.13	3	73.19
14	3	15	64.81	20	64.81
15	2	19	60.10	19	60.10
16	3	1	71.15	1	71.15
17	4	28	61.70	32	61.70
18	3	42	50.50	42	51.92
19	4	30	51.39	29	52.02
20	6	0	90.28	0	94.58
21	5	9	73.89	9	73.89
22	3	17	53.98	15	56.14
23	6	26	55.51	22	62.23
24	3	17	53.19	25	55.35

Table 4.4 CPU Time of modified ART1 and GA

S.N	Problem Size	CPU Time (Sec)	
		Modified ART 1	GA
1	5 x 7	0.060213	0.109890
2	16 x 30	0.396320	1.043956
3	30 x 50	1.854945	6.043956

Chapter – 5

Cell Formation with Operational Sequence

5.1 INTRODUCTION

Chapter 3 and chapter 4 deal with cell formation problem using operational time where workload on the machines by parts is considered. In this chapter, cell formation with operational sequence is considered where the machine cells and part families are identified such that movement of parts from one cell to another cell is minimised. The routing information is converted in the form of PMIM with sequence data. This PMIM is taken as the input for the formation of part families. The part families and machine cells are identified from the diagonal blocks of the output matrix. If any value exists in the off diagonal blocks, it indicates the inter-cell movements of the respective parts. Several methods to solve cell formation problem, as described in chapter 2, are based on iterative procedure.

The neural network applications proposed by Malave and Ramachandran (1991), Dagli and Huggahalli (1991) and Kaparthi and Suresh (1992) have demonstrated the ability of a neural network in solving cell formation problem. In all these methods, the input is the Part-Machine Incidence Matrix. The demerits of CF problem with such methods have already been discussed in chapter 2. However, some of the popular algorithms viz. the Clustering Algorithm (CASE) (Nair and Narendran 1998) and Fuzzy ART algorithm (Suresh et al. 1999) found in the literature have been proved to produce satisfactory results for the CF problem with sequence data otherwise known as *ordinal level data*.

5.2 CELL FORMATION WITH OPERATIONAL SEQUENCE

The ordinal level data or operational sequence of the parts is the commonly used information in real time manufacturing environment where MPIM is formed by replacing the '1's in the incidence matrix. The resultant values can take any value in the ordinal scale, and they constitute the ordinal level data (George et al. 2003). In this chapter, an algorithm is proposed to make use of the operational sequence of the parts, obtained through their route sheets to group the parts and machines, with an idea to minimise the number of inter cell movements of the parts. The

proposed algorithm employs the principle of ART1 network found in the literature (Venkumar and Haq 2005).

Basically the ART1 network classifies a set of binary vectors into groups based on their similarity. The ART1 recognizes patterns and clusters of the binary vectors with the recognized pattern based on the devised comparison mechanism. The proposed algorithm first converts the given non-binary sequence data into a zero-one binary matrix known as PMPM and feed the ART1 network with PMPM as the input matrix. The performance of the proposed ART1 based algorithm is compared with that of the existing Clustering Algorithm (CASE) and Fuzzy ART algorithm found in the literature.

5.3 APPLICATION OF ART1 IN THE PROPOSED ALGORITHM

In the proposed algorithm the second phase rests on the application of ART1 which consists of three processes. The first one is cluster search process, in which the network computes a matching score to reflect the degree of similarity of the present row-wise input vector (X_i) to the existing stored neurons. The initial t_{ji} and b_{ij} weights are initialized by using the following equations (5.1, 5.2 and 5.3).

$$b_{ij} = \frac{1}{(1+N)} \quad \textbf{(5.1)}$$

$$t_{ji} = 1 \quad \textbf{(5.2)}$$

The matching score for neuron *j*, is defined by

$$\mu_j = \sum_i b_{ij}(t)\, x_i \quad \textbf{(5.3)}$$

N is the number of input neurons.

The largest (μ_j) implies that the most like group and the associated group *J* is the candidate of the group.

The next process in the first phase is cluster verification process. Even though *J* is the most like group, it does not guarantee that the X_i will pass the vigilance test. The vigilance threshold $(\rho), 0 \le \rho \le 1$, determines the degree of the required similarity between the current input and a neuron already stored. The

similarity check is done to verify whether the neuron belongs to the same previously stored pattern. Otherwise, the process returns to the cluster search process and tries the next largest (μ_j).

The last process of the first phase is the cluster learning process. If the similarity between the X_i and the group *J* is good enough, then the vector X_i is accepted as a member of group *J*. The learning process updates b_{ij} and t_{ji}. For the new group the t_{ji} is identical to the X_i. But for the already stored neuron the "logical AND" is applied between X_i and the t_{ji}.

In the proposed algorithm, PMIM with sequence data is converted into a PMPM with zeros and ones. Then PMIM is fed into the ART1 algorithm to get the part families. In the next stage of the proposed algorithm, a subsequent mechanism to group the machines is incorporated.

5.4 THE PROPOSED ALGORITHM FOR CELL FORMATION WITH OPERATIONAL SEQUENCE

The input to the algorithm is the sequence based part -machine incidence matrix (PMIM) of size $N \times M$ for the *M* machines and *N* jobs cell formation problem.

Phase 1. Formulation of part-machine precedence matrix

Step 1. Using the given PMIM with the sequence data, for every part, a Machine–Machine Precedence Matrix (MMPM) of size $M \times M$ is constructed. Each row of a MMPM represents a machine and the '1's in the row indicate the machines which are required for the part *j* subsequently. The row corresponding to the first machine to be visited by the part, the '1's are given to all the machines required by the part, thus it holds the maximum number of ones in the MMPM of the particular part. The number of '1's is decreased by '1' to the subsequent machines required by the part. For the rows corresponding to the machine which are not required by the part, all the elements are assigned with zero.

Step 2. Using the '*N*' number of MMPMs single part-machine precedence matrix (PMPM) of size '$N \times M^2$' is constructed. Each row of the PMPM corresponds to a part and the element of the row is obtained by placing all the rows of the MMPM in a linear sequence.

Phase 2. Grouping of parts into part families using ART1

The PMPM obtained from the phase1 is given as input to the ART1 network.

Step 1. Before starting the network training process, the top-down weights t_{ji} and the bottom-up weights b_{ij} are set to initial values by using the Equations (5.2) and (5.4) respectively.

$$b_{ij} = \frac{1}{(1 + M^2)} \quad \text{for all i and j} \qquad \textbf{(5.4)}$$

The vigilance threshold ρ is suitably selected such that $0 < \rho < 1$

Step 2. Apply new input vector Xi

Step 3. Compute matching scores using equation (5.5)

The output μj of every output node j equals

$$\mu_j = \sum_i b_{ij}(t)\, x_i \quad \text{for j = 0,1,…, } (M^2 - 1) \qquad \textbf{(5.5)}$$

Step 4. Select best matching exemplar i.e. node (θ) with maximum output $\mu_\theta = max(\mu_j)$. Outputs of other neurons are suppressed. In case of tie choose the neuron with lower j.

Step 5. Vigilance test i.e. test of similarity with best matching exemplars

Compute number of 1's in the input vector using equation. (5.6)

$$\|X\| = \sum_i x_i \qquad \textbf{(5.6)}$$

Compute number of perfectly matching 1's between input vector and best matching exemplar using equation (5.7)

$$\|T \cdot X\| = \sum_i t_{i\theta} \cdot x_i \qquad \textbf{(5.7)}$$

Step 6. Similarity check is done using equation (5.8).

$$\frac{\|T \cdot X\|}{\|X\|} > \rho \qquad \textbf{(5.8)}$$

If the similarity is greater than ρ go to Step 7.

Step 7. Disable the best exemplar temporarily output of the best matching node selected in the step 4 is temporarily set to zero; other outputs have already been suppressed. Then go to step 3. In step 3, a new neuron in the output layer gets selected to represent the new class.

Step 8. Update best matching exemplar using equations. (5.9 and 5.10).

$$t_{i\theta}(t+1) = t_{i\theta}(t).x_i \qquad \textbf{(5.9)}$$

$$b_{i\theta}(t+1) = \frac{t_{i\theta}(t).x_i}{0.5 + \sum_i t_{i\theta}(t)x_i} \qquad \textbf{(5.10)}$$

Step 9. Repeat the step 2 after enabling any nodes disabled in step 6

The output of this phase will be the optimal number of part families and the list of parts within each part family.

Phase 3. Grouping of machines into machine cells

Step 1. Each machine is allocated to a cell corresponding to a particular part family where the total number of operations required by all the parts in the family put together is maximum.

Step 2. The columns of the output are rearranged into block diagonal form such that the number of inter cell movements are kept minimum.

5.5 MEASURE OF PERFORMANCE

Since the data considered in this chapter is operational sequence of parts, the MGE as used in chapter 3 to calculate the measure of performance, can not be used for this purpose. Hence the Group Technology Efficiency (GTE) given by Harhalakis et al. (1990) can be conveniently used to measure the performance considering operational sequence of parts. GTE is defined as the ratio of the difference between the maximum numbers of inter-cell travels possible and the numbers of inter-cell travels actually required by the system to the maximum numbers of inter-cell travels possible as given in equations. (5.11 and 5.12).

The maximum numbers of inter-cell travels possible in the system is

$$I_p = \sum_{j=1}^{N} (n_o - 1) \qquad \textbf{(5.11)}$$

The number of inter-cell travels required by the system is

$$I_r = \sum_{j=1}^{N} \sum_{w=1}^{n_o - 1} t_{njw} \qquad \textbf{(5.12)}$$

The GTE is calculated using equation (5.13)

$$GTE = \frac{I_p - I_r}{I_p} \qquad \textbf{(5.13)}$$

I_p Maximum number of inter-cell travel possible in the system.

I_r Number of inter-cell travel actually required by the system.

n_o Number of operations (w=1,2,3,…,n_o)

t_{njw} = 0 if the operations w, w+1 are performed in the same cell

= 1 otherwise.

5.6 NUMERICAL ILLUSTRATION

Table 5.1 shows the sequence based PMIM of an example problem wherein seven parts are to be processed using five machines. For every part, an MMPM is constructed. Table 5.2 shows the MMPM for the part 1and and Part 2. Table 5.3 shows the PMPM constructed as per step 2 of phase 1 of the algorithm. The output of phase 2 is shown in Table 5.4. There are two part families. The parts 2,3,4 and 6 are in one family and the parts 1,5 and 7 are in another family. Table 5.5 shows the output of the phase 3 of the algorithm, the optimal solution to the sample CF problem. It is observed from the output matrix that the parts p2, p4, p6 and p7 have one exceptional element each and one inter-cell move. The part p3 has two exceptional elements and one inter-cell move. Hence there are six exceptional elements and five inter-cell moves. The group technology efficiency (Nair and Narendran 1998) is calculated using equation (5.13). The value of GTE is 64.3 %.

Table 5.1 Part machine incidence matrix with sequence data (7 × 5)

	m1	m2	m3	m4	m5
p1	1	2	0	3	0
p2	0	1	2	0	3
p3	2	0	0	1	3
p4	0	1	2	0	3
p5	1	2	0	3	0
p6	3	0	1	0	2
p7	0	3	0	2	1

Table 5.2 Machine – machine precedence matrix for parts

	(for part-1)						(for part-2)				
	m1	**m2**	**m3**	**m4**	**m5**		**m1**	**m2**	**m3**	**m4**	**m5**
m1	1	1	0	1	0	**m1**	0	0	0	0	0
m2	0	1	0	1	0	**m2**	0	1	1	0	1
m3	0	0	0	0	0	**m3**	0	0	1	0	1
m4	0	0	0	1	0	**m4**	0	0	0	0	0
m5	0	0	0	0	0	**m5**	0	0	0	0	1

Table5.3 Part machine precedence matrix for the problem size 7 × 5

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**
p1	1	1	0	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
p2	0	0	0	0	0	0	1	1	0	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0	1
p3	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	1
p4	0	0	0	0	0	0	1	1	0	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0	1
p5	1	1	0	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
p6	1	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	1	0	0	0	1
p7	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	1	1

Table 5.4 Output of phase 2 – Part families

Cell No.	Parts
1	**p2**
1	**p3**
1	**p4**
1	**p6**
2	**p1**
2	**p5**
2	**p7**

Table 5.5 Final output matrix (7 × 5)

	m3	m5	m1	m2	m4
p2	2	3	0	1	0
p3	0	3	2	0	1
p4	2	3	0	1	0
p6	1	2	3	0	0
p1	0	0	1	2	3
p5	0	0	1	2	3
p7	0	1	0	3	2

5.7 RESULTS AND DISCUSSION

In previous chapters, operational time is taken into account to solve CF problem. In this chapter, operational sequence of the parts is considered to form machine cells and part families in CF. In light of the literature (Nair and Narendran 1998) it hardly finds methods that consider the issue of operational sequence while constructing CF. Since the basic ART1 consider only zero-one binary data as input, it is required to change the input pattern which reflects operational sequence information. Therefore a suitable transformation method to convert the sequence data into binary data is developed in this chapter and hence ART 1 is successfully used to handle this problem and form clusters to identify machine cells and parts families. The following major contributions are made in this chapter.

- A transformation method is developed to convert the operational sequence (ordinal level data) into binary data so that it can be fed into the basic ART1 algorithm.
- Appropriate performance measure is considered to incorporate operational sequence of the parts.

The ART1 based algorithm for the cell formation problem with operational sequence (operation sequence data) has been coded in C^{++} and executed in a Pentium IV, 2.4 GHz system. The performance of the proposed algorithm is tested with example problems and the results are compared with the existing methods found in the literature. Table 5.6 shows the problems of different sizes selected from open literature (Nair and Narendran 1998) for testing the proposed algorithm. It is found that the proposed algorithm for machine cell formation with sequence data gives satisfactory results which are either superior or same as the existing methods found in literature. The results are compared with the results produced by CASE algorithm (Nair and Narendran 1998) as shown in Table 5.6.

Table 5.6 Comparison of results of the proposed algorithm with CASE

Data set No.	Problem Size	No. of Cells	Clustering Algorithm (CASE)			Proposed ART1 based Algorithm		
			EE	IM	GTE	EE	IM	GTE
50	7 x 7	2	2	4	69.25	2	4	69.25
50	7 x 7	3	3	6	53.85	3	6	53.85
51	20 x 8	3	10	17	58.54	10	17	58.54
52	20 x 20	4	-	-	-	12	15	74.58
52	20 x 20	5	15	19	67.80	16	18	69.49
53	40 x 25	5	-	-	-	26	22	72.04
53	40 x 25	8	35	31	66.67	35	31	66.67

The output of the example problem of size 8×20 obtained by the proposed algorithm is compared with the other two methods namely CASE (Nair and Narendran 1998) and Analytical Iterative Approach (George et al. 2003) as shown in Table 5.7. In addition to the group technology efficiency and the number of exceptional elements, performance measures used in the literature, the number of inter cell movements is also used for evaluating and comparing the performance of the algorithms.

Table5.7 Comparison of results of the proposed method over existing methods for the problem of size 20×8 (data set 51)

CASE (Nair & Narendran 1998)		Analytical-iterative clustering algorithm (George et al. 2003)	Proposed Algorithm
Cell No.	Parts allocated in the cells	Parts allocated in the cells	Parts allocated in the cells
1	1 5 10 12 15	1 5 10 12 15	1 5 10 12
2	2 8 9 11 13 14 16 17 19	2 8 9 11 13 14 16 17 19	2 8 9 11 13 14 16 17 19
3	3 4 6 7 18 20	3 4 6 7 18 20	3 4 6 7 15 18 20

As for as sequence matrix is concerned the inter-cell moves are calculated using the equations (5.10 and 5.11). If the operation of a part is allotted in the

same cell where the pervious operation of the part has taken place, then the inter-cell move is considered as zero. The total possible inter-cell moves are calculated just by taking summation of the difference between one and maximum operation of each part as given in equation (5.10). The output of problem of sizes 19×12, 20×20 and 40×25 are compared with existing approaches as shown in Tables 5.8, 5.9 and 5.10 respectively. In ART1, the vigilance threshold (ρ) value greatly influences the number of cells obtained. The vigilance threshold value for each problem is varied from 1 to 9.

Table 5.8 Comparison of results for the example problem of size 19 × 12 (data set 59)

	Kiang et al. (1995)	**Fuzzy ART (Suresh et al. 1999)**	**Flexible beta (Park and Suresh 2003)**	**Proposed Algorithm**
Cell No.	Parts allocated in the cells	Parts allocated in the cells	Parts allocated in the cells	Parts allocated in the cells
1	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
2	5 6 11 18	5 6 18	5 6 18 19	5 6 18 19
3	7 8 9 10	7 8 9 10 11	7 8 9 10 11	7 8 9 10 11
4	12 13 14 15 16 17 19	12 13 14 15 16 17 19	12 13 14 15 16 17	12 13 14 15 16 17

Table 5.9 Comparison of results for the example problem of size 20 × 20 (data set 61)

	Harhalakis et al. (1990)	**CASE (Nair and Narendran 1998)**	**Fuzzy ART (Suresh et al. 1999)**	**Flexible beta (Park and Suresh 2003)**	**Proposed Algorithm**
Cell No.	Parts allocated in the cells	Parts allocated in the cells	Parts allocated in the cells	Parts allocated in the cells	Parts allocated in the cells
1	1 9 12 17 20	1 9 12 14 17 20	1 9 12 17 20	1 9 12 17 20	1 9 12 17 20
2	2 4 11 19	2 4 11 19	2 4 11 19	2 4 11 19	2 4 11 19
3	3 10 14 18	3 10 18	3 10 14 18	3 10 14 18	3 10 14 18
4	5 8 13 16	5 8 13 16	5 8 13 16	5 8 13 16	5 8 13 16
5	6 7 15	6 7 15	6 7 15	6 7 15	6 7 15

Table 5.10 Comparison of results for the example problem of size 40 × 25 (data set 53)

CASE (Nair and Narendran 1998)		Fuzzy ART (Suresh et al. 1999)		Flexible Beta (Park and Suresh 2003)	Proposed Algorithm
Cell No.	Parts allocated	Cell No.	Parts allocated	Parts allocated	Parts allocated
1	18, 32	1a	18	11, 18, 27, 29, 32	18, 32
		1b	32		
2	1, 5, 7, 16, 17, 30	2	1, 5, 7, 15, 16, 17, 30	1, 5, 7, 16, 17, 30	1, 5, 7, 16, 17, 30
3	8, 15, 23, 24, 31	3	8, 23, 24, 21	8, 15, 23, 24	8, 15, 23, 24, 31
4	3, 9, 13, 14, 33	4	3, 9, 13, 14, 33	3, 9, 13, 14, 33	3, 9, 13, 14, 33
5	11, 25, 27, 29,5,40	5a	25, 35, 40	25, 35,40	11, 25, 27, 29, 35,40
		5b	11		
		5c	27		
		5d	29		
6	4, 6, 20, 26, 34, 37, 39	6a	6, 20, 26, 34, 37, 39	4, 6, 20, 26, 34, 37, 39	4, 6, 20, 26, 34, 37, 39
		6b	4		
7	2, 12, 36	7	2, 12, 36	2, 12, 31, 36	2, 12, 36
8	10, 19, 21, 22, 28,38	8a	10, 22	10, 19, 21, 22, 28,38	10, 19, 21, 22, 28,38
		8b	19, 21, 28, 38		

It is found that the number of cells equals to the total number of parts if the vigilance threshold value is set at zero. Figure 5.1 shows the effect of vigilance threshold with the number of cells and it is inferred from the Figure 5.1 that the number of cells is inversely proportional to the value of vigilance threshold. As the vigilance threshold value increases, the number of cells is reduced. If the vigilance threshold value is further relaxed, the algorithm produces only one cell.

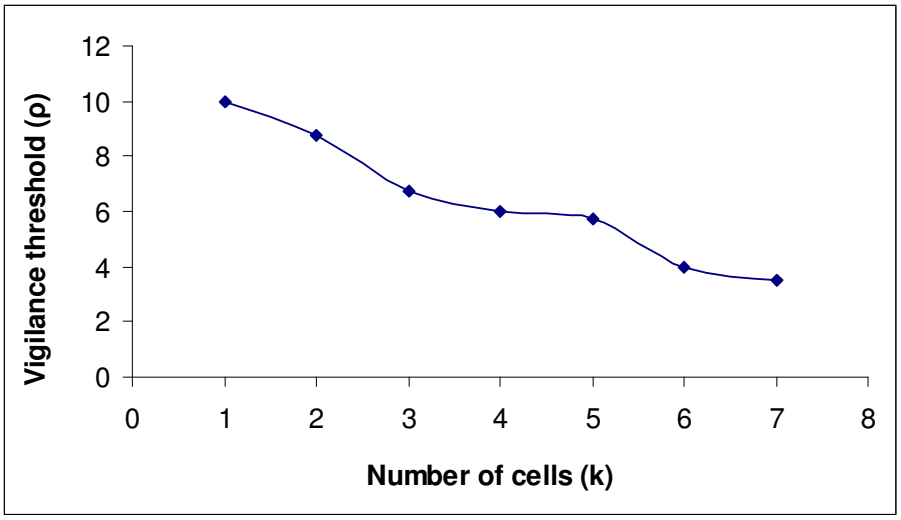


Figure 5.1 Effect of vigilance threshold (Sequence data)

Therefore, vigilance threshold value plays a vital role for obtaining quality solution. For each sample problem, the vigilance threshold has been varied to tune the algorithm and it is incremented in the step of 0.5 starting from zero till desired solution is obtained. Fifteen trial data sets are tested using the proposed algorithm and the results are given in Table 5.11. The output matrix of the problem of size 40×25 is given in Table 5.12.

Table 5.11 Performance of the proposed algorithm on test data sets (Sequnce data)

Data set No.	Problem Size	Exceptional Elements	Inter-cell Moves	Group Technology Efficiency
55	5 x 4	0	0	100.00
56	5 x 5	1	1	85.71
57	7 x 5	6	5	64.30
58	8 x 6	2	2	84.61
59	19 x 12	8	9	83.93
60	20 x 12	11	10	78.00
61	20 x 20	3	3	94.00
62	30 x 15	21	17	76.71
63	37 x 20	25	25	71.59
64	50 x 25	49	46	69.13
65	55 x 20	15	19	81.20
66	60 x 28	39	38	70.50
67	65 x 30	58	52	76.68
68	80 x 32	53	59	74.57
69	90 x 35	54	56	77.69

Table 5.12 Output matrix by the proposed algorithm for example problem of size 40 × 25 (data set 53)

j\i	4	5	7	12	16	18	19	23	1	2	17	24	3	11	20	25	8	9	10	6	13	14	15	21	22
1	5		3		4	2													**1**						**6**
4				1				2																	
5	3			2		1																			
6				3	2			1																	
7	3		2		4	1													**5**						
8		1			3		2																		
15		3			1		2			**4**									**5**						
16	1		3		2	4																			
17			1			2													**3**						
20								1																	
23		2			3		1																		
24		1			2																				
26	2			3				1															**4**		
29		3																		**2**				**1**	
30	4		2		3	1																			
34				2				1				**3**													
37			3	2				1																	
39				1																					
40					1				**2**										**3**						
2									2	3	4					**1**									
12									1		3	2	**4**			**5**									
31		**2**					**3**				1														
36									2	3	1			**4**											
3													2	3	1										
9													3	4	1	2									
13													3	2	1										
14		**1**											4	2	3										
22											**1**		3				**4**	**2**							
33														1	3	2									
10																3	2	1							
11																	2				**3**			**1**	
19																	1	3	2						
21																	1	3	2						
28																	2	1	3						
38																	2	3							**1**
18																						3	2		1
25																				1			3	2	
27				**1**																				3	2
32																					2	1	3		4
35																				2			4	1	3

The proposed algorithm provides solution in a single iteration only. The advantage of the proposed algorithm lies in its ability to generate quality solution for large size problems. The algorithm is flexible in such a way that the maximum number of parts to be accommodated in a family can be limited. Most of the existing methods use similarity measures to solve CF problem with sequence input. The disadvantage is that those measures forget about the important measure called exceptional elements which is the core factor in building CMS model. The draw back is avoided in this work. Considering the example problem of size 40 x 25, the results shows 35 exceptional elements and 31 inter-cell moves in the existing CASE algorithm, for which the proposed algorithm produces the solution with 26 exceptional elements and 22 inter-cell moves. It shows that the proposed algorithm is superior over existing methods as for as CF problem with operational sequence or parts is concerned.

Chapter – 6

Cell Formation with Operational Time and Sequence

6.1 INTRODUCTION

In previous chapters the cell formation problems are solved using the information of workload data and operation sequence data individually which are taken from the route sheet. But in practice, for constructing the cells, individual solutions are least preferred. Taking this into consideration in this chapter, an attempt has been made to make use of the operation sequence of the parts (known as ordinal level data) and the operation time of the parts (known as operational time or workload data) which are obtained through the route sheets to group the parts into part families and machines into machine cells. The proposed algorithm employs the principle of modified ART1 network which has been used in the chapter 3. The proposed algorithm first converts the given non-binary data into a zero-one binary matrix known as part machine precedence matrix (PMPM). The workload data is then multiplied with this PMPM to get Matrix of Combine Data (MCD) which reflects the combined information of operation sequence and time. This information is fed in to the modified ART1 network with MCD as the input matrix.

6.2 THE ART BASED ALGORITHM FOR CELL FORMATION WITH COMBINED OPERATION TIME AND SEQUENCE

The input to the algorithm is the operational time and sequence based part - Machine incidence matrix (PMIM) of size $N \times M$.

Step 1: Using the given PMIM with the sequence data, for every part, a machine-machine precedence matrix (MMPM) of size $M \times M$ is constructed. Each row of a MMPM represents a machine and the ones in the row indicate all other machines which are required subsequently for the part considered. However for the row corresponding to the first machine to be visited by the part, the '1's will appear in all the machines required by the part, thus it has the maximum number of '1's in the MMPM of the particular part. For the rows corresponding to the machine which are not required by the part, all the elements are assigned with zero.

Step 2: using the n number of MMPMs, a single Part-Machine precedence matrix of size $N \times M^2$ is constructed. Each row of the PMIM corresponds to a part and the element of the row is obtained by placing all the rows of the MMPM in a linear sequence.

Step 3: Multiply all the ones present in the PMPM by the respective workload data from the work load matrix which represents operational time of the parts. The new matrix is a combination of operational time and operational sequence of the parts which is named as matrix of combined data (MCD)

Step 4: The MCD will be the input to the modified ART1 which is given in the steps (5-15).

Step 5: Set nodes in the input layer equal to *N* (number of parts) and nodes in output layer equal to (M^2). Set vigilance threshold (ρ).

Step 6: Initialize top-down connection weights.

Top-down weights are initialized using equation (6.1).

$$wt_{ji}(0) = 0 \qquad \textbf{(6.1)}$$

for i = 1, 2,...*N*. and j = 1,2,... (M^2).

Step 7: Let q =1. The first input vector X_1 (first row of the workload matrix) is presented to the input layer and assigned to the first cluster. Then, first node in the output layer is activated.

Step 8: The top-down connection weights for the present active node are set equal to the input vector.

Step 9: Let q = q+1. Apply new input vector X_q. (input vectors are the rows of the PMPM).

Step10: Compute Euclidean distance between X_q and the exemplar stored in the top-down weights (wt_{ji}) for all active nodes i as given in the equation (6.2). This distance function is used to calculate similarity between the stored pattern and the present input pattern. If the similarity value is less than or equal to ρ (vigilance threshold), the present input is categorized under the same cluster as that of stored pattern.

$$e_i = \sqrt{\sum_{j=1}^{M} (x_{qj} - wt_{ji})^2} \qquad \textbf{(6.2)}$$

Step11: Perform vigilance test: Find out minimum Euclidean distance.

Step12: If min $e_i \leq \rho$ (threshold value), select output node for which Euclidean distance is minimum. If tie occurs, select the output node with lowest index number. Suppose output node k is selected then allocate the vector X_q to the node k (cell) and activate node k. Make increment to the number of parts in the active node k by one. If e_i's for all active nodes are greater than ρ, then go to step 13.

Step13: Start a new cell by activating a new output node.

Step14: Update top-down weights of active node k.

The decision for belongingness of an input vector to a node (cluster) is determined using similarity between previously stored exemplar with present input pattern. In other words, top-down weights play the role of storing exemplars (for active nodes) for comparison purpose. Therefore, top-down weights must contain relevant information of all the input vectors already classified under an active node (cluster) in aggregate nature. The top down weights are updated each time when a new input vector is presented and clustered to an active node. When a vector is selected (to be allocated to an output node), its top-down weights are updated using more information of the previously stored exemplar and a relatively less information of the input vector (pattern)

as shown in equation (6.3).

$$wt_{jk} = (\frac{n}{m} . wt_{jk}) + (\frac{1}{m} . x_{qj}) \qquad \textbf{(6.3)}$$

Step15: Go to step 5 and repeat till all the rows are assigned in the output nodes (cells).

Step16: Check for single ton part family. If a single ton is found in any part family, perform the following operations to merge the part family with one part into any other part families.

- **Determine average of processing time in each part family.**
- **Calculate the Euclidean distance between the part families.**
- **Merge the part family containing single part with another part family in such a way that Euclidean distance between them is minimum than other part families.**

Step17: Allocate machines to the part families using following supplementary procedure

- **The number of operations of a part in a particular machine is computed. If the part has maximum number of operations in machine i then the machine i is allocated to that part family where the part exists.**
- **In case tie occurs the machine is allocated with the part family where minimum inter-cell moves are possible.**
- **In case tie occurs again the machine is allocated with the small identification number of part family.**

6.3 MEASURE OF PERFORMANCE

Group technology efficiency (GTE) given by Harhalakis et al. (1990) and grouping efficiency for ratio level data (as proposed in this chapter) can be combined to measure the performance considering sequence of parts and operational times of the parts combinedly. Group technology efficiency is defined in chapter 5.

In this work the grouping efficiency for ratio level data (GER) is defined as the ratio between the total processing time inside the cells to the summation of exceptional elements, total processing time of the cells and total value of the voids present in the cells. Voids factor for cell *k* is calculated by multiplying the number of voids in cell *k* to the average time of machine *i* in cell k. The void factor of cell *k* is multiplied by the total processing time of cell *k* to give the value of voids present in cell *k*. GER is calculated using equation (6.4).

$$GER = \frac{\sum_{k=1}^{c} T_{ptk}}{N_e + \sum_{k=1}^{c} T_{ptk} + \sum_{k=1}^{C} \left(N_{vi} \times \frac{\sum T_{ik}}{\sum N_p}\right)} \quad \textbf{(6.4)}$$

Hence, a new measure of grouping efficiency termed as Ratio Ordinal Combined Efficiency (ROCE) has been proposed in this work to find out the goodness of the grouping in the cell formation problem that deals with both operation sequence and time in the input matrix with due consideration of equal weightages to both the data. Ratio-Ordinal Combined Efficiency is calculated using the equation (6.5).

Ratio Ordinal Combined Efficiency (ROCE) is defined as the weighted average of GER and GTE.

$$ROCE = q\,(GER) + (1 - q)\,(GTE) \quad \textbf{(6.5)}$$

The problem of size 12 × 10 from (George et al. 2003) is considered for illustrating the proposed performance measure ROCE. Initially the input matrix (12 × 10) is fed into the algorithm. The output generated is given in Table 6.1 and 6.2. Table 6.1 is the output matrix with operation sequence data. Table 6.2 represents the output matrix in terms of operation time. The total number of exceptional elements of the matrix (N_e = 4). The total processing time inside each cell has been calculated and found to be (T_{ptk}) => (T_{pt1}=11.4), (T_{pt2}=10.05), (T_{pt3}=4.01). The number of voids in each column is a count of zeros present in the respective columns inside the cells (N_{vi}) => (N_{v1}=1, N_{v4}=1, N_{v5}=1, N_{v8}=2, N_{v9}=1, N_{v10}=1). The average of the operational time of parts in each column is calculated which is multiplied by the respective N_{vi} values to get the void factor. The maximum number of inter-cell travel possible in the system (I_p) is found to be 25 and the number of inter-cell travel actually required (I_r) by the system is 4. The maximum number of operations (n_o) is given as 5. The values of GTE, GER and

Table 6.1 Output matrix with operation sequence for the problem of size 12 x 10 (data set 54)

	1	3	6	2	5	8	10	4	7	9
1		1	3					2		
5	3	5	1			2		4		
9	4	1	2			3				
10	3	1	2							
2				1	3	4	2			
3				2	4	1	3			
8				1			2			
12				3	2	1				
7				1	3		2			
4								1	3	2
6								1	3	2
11									1	

Table 6.2 Output matrix with operation time for the problem of size 12 x 10

	m1	m3	m6	m2	m5	m8	m10	m4	m7	m9
p1	0	0.96	0.63					0.95		
p5	0.63	0.97	0.61			0.94		0.89		
p9	0.54	0.92	0.72			0.92				
p10	0.39	0.61	0.72							
p2				0.86	0.54	0.04	0.67			
p3				0.88	0.49	0.08	0.73			
p7				1.2	0.81	0	0.83			
p8				1	0	0	0.62			
p12				0.7	0.72	0.02	0			
p4								0.07	0.83	0.72
p6								0.11	0.99	0.76
p11								0	0.71	0

ROCE are calculated using equations (5.13, 6.4 and 6.5) by substituting these values. Hence the value of GTE = 0.84 GER = 0.7645, and ROCE = 0.8023.
These values are calculated for the output of the existing methods also and presented in Table 6.3

It is observed that the results of the proposed performance measure outperforms the existing methods ACCORD (Nair and Narendran, 1999), Analytical iterative approach (George et al, 2003)

Table 6.3 Comparison of the results of the proposed method over existing methods for the problem of size 12 × 10 (data set 54)

Factors considered	ACCORD	Analytical Iterative Approach	Proposed method
Exceptional elements	5	5	4
Grouping efficiency	0.881	0.881	0.897
Grouping efficacy	1.026	1.026	1.026
Grouping efficiency for ratio level data (GER) %	69.24	69.24	76.45
Group technology efficiency (GTE) %	80.00	80.00	84.00
Ratio ordinal combined efficiency (ROCE) (%)	74.62	74.62	80.23

6.4 RESULTS AND DISCUSSIONS

In chapters 3, 4 and 5, important real life production factors such as operational time and operational sequence of the parts are considered separately to solve CF problem. In this chapter, both operational sequence and operational time of the parts are considered combinedly to form machine cells and part families while dealing with CF. From the literature it is understood that methods that consider combination of production factors are very small in number (Nair and Narendran 1999). The proposed algorithm make use of modified ART 1 presented in chapter 3 after the combined matrix is formed which reflects the ratio level data. The matrix with operational sequence of parts is converted into binary matrix initially and the operational time is combined with this matrix by multiplying the respective values. The resultant matrix (MCD) reflects the combination data which is a new contribution adapted in this chapter. Besides that the following major contributions are made in this study.

- The proposed methodology is open for considering any real time production factors which is in the form of either ratio level data or ordinal level data and combination of these data in CF problems.
- Appropriate performance measures are developed to consider the production factors such as operational time and operational sequence of the parts combinedly.

The algorithm is coded in C^{++} and run on an IBM Pentium IV PC with 2.4 GHz Processor. Table 6.4 shows the problems of different sizes selected from open literature (Nair and Narendran 1998; George et al. 2003) for testing the proposed algorithm. For all trial data sets shown in Table 6.4, the input matrix is generated with uniformly distributed random numbers in the range of 0.5 to 5 for operational time and 1 to 9 for operational sequence. The problem sizes considered in this work range from 5×4 to 90×35.

Table 6.4 Performance of the proposed algorithm on test data sets (combined data)

Data set. No	Problem Size	Exceptional Elements	Inter-cell Moves	GTE	GER	ROCE
55	5 x 4	0	0	100.00	83.48	91.74
56	5 x 5	1	1	85.71	81.15	83.43
57	7 x 5	6	5	64.30	72.01	68.16
58	8 x 6	2	2	84.61	70.15	77.38
59	19 x 12	8	9	83.93	65.08	74.51
60	20 x 12	11	10	78.00	59.56	68.78
61	20 x 20	3	3	94.00	84.25	89.13
62	30 x 15	21	17	76.71	60.02	68.37
63	37 x 20	25	25	71.59	60.99	66.29
64	50 x 25	49	46	69.13	58.39	63.76
65	55 x 20	15	19	81.20	66.03	73.62
66	60 x 28	39	38	70.50	57.20	63.85
67	65 x 30	58	52	76.68	59.59	68.14
68	80 x 32	53	59	74.57	62.28	68.43
69	90 x 35	54	56	77.69	62.26	69.98

The results are compared with the results produced by ACCORD and Analytical Iterative Approach as shown in Table 6.3. In addition a new weighted average performance measure ROCE is proposed that measures the performance of the algorithm proposed and tested with fifteen trial data sets. The results are found to be consistent for all the data sets tested which are shown in Table 6.4. The result of the example problem of size 12 × 10 obtained by the proposed algorithm outperforms other two methods as shown in Table 6.3. The weights for the exceptional elements are given as one. Since in reality, the voids are not influencing the system as much as that of the exceptional elements, the weights for the voids are proportionally taken to the average values of the respective columns where the voids exist. If the operation of a part is allotted in the same cell where the pervious operation of the part has taken place, then the inter-cell move is considered as zero. The total possible inter-cell moves are calculated just by taking summation of the difference between one and maximum operation of each part. It is the decision maker's choice to fix the value of the weighting factor q while calculating the proposed performance measure ROCE. Here, the value of q is considered as 0.5 for illustrating the performance measure by giving equal weightage to both GER and GTE. Since both operational time and operational sequence of parts are given equal priority in production flow analysis, the weightage factor is assumed to be 0.5 each but it is not restrictive to 0.5. It depends on the problem where either operational time or operational sequence is more prioritized, can be given more weightage. The proposed algorithm provides solution in a single iteration only. The advantage of the proposed algorithm lies in its ability to generate quality solution for large size problems.

The vigilance threshold value for each problem is varied from 1 to 9. It is found that the number of cells equals to the total number of machines if the vigilance threshold value is set at zero. As the vigilance threshold value increases, the number of cells is reduced as shown in Figure 6.1. If the vigilance threshold value is further relaxed, the algorithm produces only one cell. Therefore, vigilance threshold value plays a vital role for obtaining quality solution. The threshold value of 6.5 makes 5 cells and the threshold value of 7.0 produces 3 cells for the

problem of size 12 x 19. It is observed that the influence of the threshold value on clustering cells varies depending on the type of problems.

For each sample problem, it is incremented in steps of 0.5 starting from zero until desired solution is obtained. The algorithm also takes care of avoiding cells with singleton part family that is encountered at times. The algorithm is flexible in such a way that the maximum number of parts to be accommodated in a family can be limited. From the Table 6.3, it is observed that the grouping efficiency and grouping efficacy measures found in the literature produce the values are almost same in case of all the three methods compared. But as for as number of exceptional elements and the proposed measures like GER, GTE and ROCE the modified ART1 methodology shows better performance. Hence the proposed grouping efficiency measures are evidently suitable to measure the performance of cell formation algorithm taking into account operational time and operational sequence of the parts. The proposed algorithm is not limited to considering only the above mentioned production factors. For instance, for the data set 4, the batch size is assumed to all the parts (P1–10, P2–20, P3–10, P4– 5, P5–20, P6–25, P7–15, P8–10). Now the operational time in input matrix will be multiplied by the respective batch size and can be fed into the modified ART1 algorithm. The output is as shown in Table 6.5. This is an entirely different output when it is compared with the output of original problem as shown in Table 3.1d. Hence, it is inferred that based on the input pattern the proposed algorithm acts upon accordingly to get the desired output. This shows that the modified ART1 is flexible enough to consider any type of production factors in the form of either ratio level data or ordinal level data.

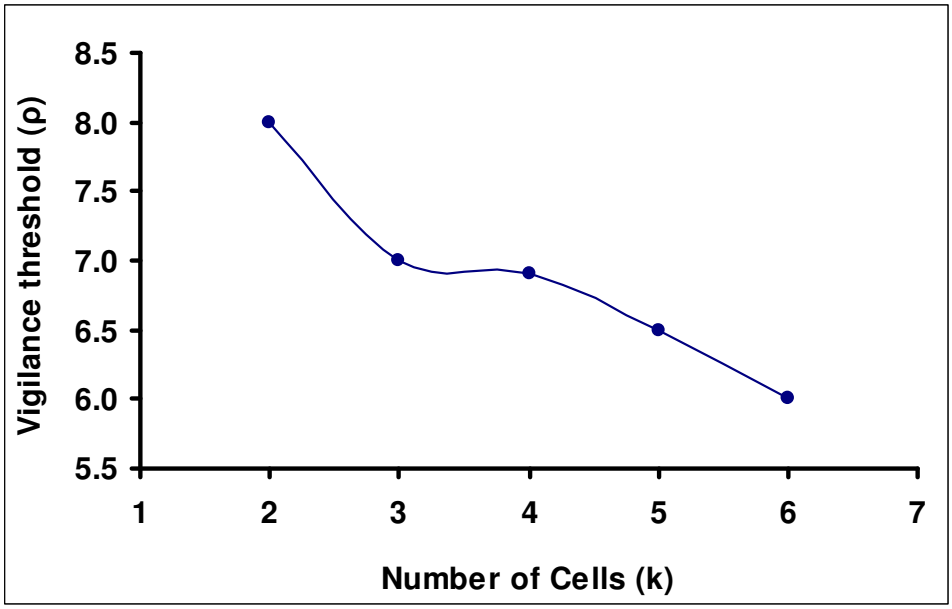


Figure 6.1 Vigilance threshold Vs number of cells (combined data)

Table 6.5 Illustration of problem of size 6 × 8 considering batch size with workload for parts (data set 4)

	P1	P2	P3	P4	P5	P6	P7	P8
M1	0	10.60	0	4.95	0	0	12.50	0
M2	9.10	16.40	8.30	0	18.20	23.00	12.90	9.70
M3	0	0	7.90	0	0	12.90	0	8.80
M4	0	0	0	2.65	0	0	7.65	0
M5	9.80	0	8.30	0	14.20	14.20	0	5.40
M6	0	0	0	2.70	0	0	11.10	0

Input matrix

	P3	P4	P7	P8	P1	P2	P5	P6
M1	0	4.95	12.5	0	0	10.60	0	0
M3	7.90	0	0	8.80	0	0	0	12.90
M4	0	2.65	7.65	0	0	0	0	0
M6	0	2.70	11.10	0	0	0	0	0
M2	8.30	0	12.90	9.70	9.10	16.40	18.20	23.00
M5	8.30	0	0	5.40	9.80	0	14.20	14.50

Output matrix

Chapter – 7

Conclusions and Scope for Future Work

7.1 INTRODUCTION

In this study, an attempt is made to solve the cell formation problem considering production factors such as operational time and operational sequence of the parts using soft computing techniques and metaheuristics. For this purpose, operational time of the parts is first considered in chapter 3 and 4. It denotes that how long a part takes for its particular operation in a particular machine. Secondly, in chapter 5, the operational sequence of the parts is considered to construct machine cells and part families. This represents in which order the operations involved in a part are carried out on various machines. Thirdly, the combination of these data (i.e.) operational time and operational sequence of the parts is considered in chapter 6 and represented in a single matrix to form cells. There are very few studies which reflect these issues in the chosen field of study as referred in the literature. This research work will definitely provide a limelight to the future researcher and industrialists in this field. The soft computing techniques like ART1 and metaheuristics like GA with suitable modifications have been successfully implemented for CF problems with operational time and operational sequence respectively. The modified ART1 also deals with combined form of these data. These approaches are applied to fifty nine benchmark data sets which are found in the literature and ten randomly generated data sets and the results are compared to other algorithms in terms of various performance measures proposed in this research work. Real life large size data sets also tested in the modified ART1 for CF problems.

7.2 CONTRIBUTION OF THE RESEARCH WORK

Modification in ART1 and GA to consider real life production factors such as operational time and operational sequence has been done to solve the cell formation problems. Since the modified ART1 makes use of Euclidean distance, it is computationally simple to arrive at clusters that have machines and parts with similar attributes. Here, the vigilance threshold plays a major role for getting required number of cells. The proposed GA based algorithm is used to find out better solution for combined objective function (i.e. combination of total cell load

variation and exceptional elements). The methodology of converting the non-binary data into a suitable binary data and subsequently by feeding to the ART1 networks to solve the CF problem is an additional contribution which may lead further development in this area. While forming clusters in some cases, cells with single machines are present that is definitely not a desirable output. To avoid such cases, a supplementary procedure is developed that will take care for eliminating single machine cells while producing output which is claimed as an epsilon contribution in this study. In chapter 6, it is notable that initially the sequence matrix is converted into binary matrix (PMPM). Then, combining operational time with operational sequence, all the '1's in PMPM are multiplied with the respective operational time to get the matrix of combined data which is fed as an input to the modified ART1 algorithm. This transformation procedure is a further contribution that can be claimed from chapter 6. The new performance measures proposed in chapter 3 and chapter 6 are able to measure the goodness of the output block diagonal structure with operational time and operational sequence and also combination of these data that is an additional claim towards the contribution out of this research work. The proposed algorithms are tested with wide variety of benchmark problems from open literature (Appendix VI) and resulting solutions are compared with the solutions obtained by K-means and C-linkage algorithms.

7.3 SUMMARY OF FINDINGS

In the study undertaken in chapter 3, the modified ART1 uses Euclidean distance to find out the nearness among the machines. Hence, machine cells are formed based on this distance. The threshold value known as vigilance parameter is used to tune the algorithm and also to create the desired number of cells as required by the decision maker. Similarly in chapter 4, the proposed GA is tuned by the crossover and mutation operators. The population size can be varied from 10 to 40 depending upon the problem size. The iteration number is also varied depending on the problem size as well as the population size chosen for the particular problem. The crossover and mutation probability are maintained at 0.5 and 0.1 respectively which is not restrictive. The combined objective function used in chapter 4 is normalized to bring both the objectives to the same scale since they

are measured in different scales originally and both are contradictory in nature too. In most of the problems, it is observed that the solutions obtained by proposed algorithms either outperform the other two methods or remain the same. In genetic cell formation as discussed in chapter 4, the proposed GA considers to uniformly distributing the cell load. Hence the number of exceptional elements may be more some times as compared with the solutions obtained from other methodologies. For instance the solutions obtained in data set 13 (Table 4.3) the modified ART1 gives better results. The modified ART1 attempts to form cells based on the attributes. The machines with similar attributes are grouped in one cluster. This is the reason for GA to produce inferior solutions in few cases as compared with other methodologies considered in this study.

For each sample problem, the input matrix is generated by replacing the ones in the incidence matrix with uniformly distributed random numbers in the range of 0.5 to 1 for operational time and 1 to 9 for operational sequence. The zeros remained in the same positions. The problem sizes considered in this work ranges from 5 x 4 to 90 x 35. It is assumed that the lot size for all the parts equal to one to characterize the behaviour of the sample problems considered in this work although it is not restrictive to one**.** Since the algorithms proposed in this study uses simple network architecture, it helps to reduce computational burden for generating quick solutions for industrial applications.

7.4 LIMITATIONS OF THE STUDY

In spite of advantages obtained through proposed algorithms, the followings may be treated as limitations of the study since they have not been addressed in this study.

- Since ART1 recognizes the entire input pattern keeping the first input pattern as reference, it may not be appropriate in cases where the input matrix (pattern) is sparse.
- Revisit of the parts to a particular machine is not addressed in this work.
- Demand from customers has not been considered in this study.

7.5 SCOPE FOR FUTURE WORK

The revisit of parts to the machines and the demand for parts from customer are not considered in this work which is one of the real time production factors. The work can be further extended in future incorporating production data like revisit of parts to machines, demand for parts from customer, machine capacity, total moves of the parts, production volume, layout and tools considerations, and material handling systems enhancing it to a more generalized CMS. Different types of reproduction, crossover and mutation greatly influence the performance of GA. Therefore, various types of GA operators may be tested in cell formation problem. In multi-objective GA formulation, Pareto optimality may be tested instead of using additive objective function. The ART1 can be modified in an adaptive way so that important parameter like vigilance threshold can be fixed depending on type and size of the problem.

Bibliography

1. Adil Baykasoglu and Gindy, N.N.Z., 2000. MOCACEF: multiple objective capability based approach to form part machine groups for cellular manufacturing application. *International Journal of Production Research*, **38** (5), pp.1133-1161.

2. Anita, L-P., 2000. Part family identification using a simple genetic algorithm. *International Journal of Production Research*, **38** (4), pp.793-810.

3. Askin, R.G. and Subramanian, S.P., 1987. A cost based heuristic for grouping technology configuration. *International Journal of Production Research*, **25**(1), pp.101-113.

4. Askin, R.G. and Chiu, K.S., 1990. A graph partitioning procedure for machine assignment and cell formation in group technology. *International Journal of Production Research*, 28, pp. 1555-1572.

5. Atmani, A., Lashkari, R. S., and Caron R. J. 1995. A mathematical programming approach to joint cell formation and operation allocation in cellular manufacturing. International Journal of Production Research, 33 (1), 1-15.

6. Bahrami, A., Dagli, C.H. and Modarress, B., 1991. Fuzzy associative memory in conceptual design. *Proceedings of the International Joint Conference on Neural Networks*, Seattle, **1**, pp. 183-188.

7. Ballakur, A. and Steudel, H.J., 1987. A with-in cell utilization based heuristic for designing cellular manufacturing systems. *International Journal of Production Research*, **25** (5), 639-655.

8. Beaulieu, A., Gharbi, A. and Ait-Kadi., 1997. An algorithm for the cell formation and the machine selection problems in the design of a cellular manufacturing system. *International Journal of Production Research*, **35** (7), pp.1857-1874.

9. Ben-Arieh, D. and Triantaphyllou, E., 1992. Quantifiying data for group technology with weighted fuzzy features. *International Journal of Production Research*, **30**(6), pp.1285-1299.

10. Boctor, F.F., 1991. A linear formulation of the machine-part cell formation problem. *International Journal of Production Research*, **29** (2), pp.343-356.

11. Boe, W.J. and Cheng, C.H., 1991. A close neighbour algorithm for designing cellular manufacturing systems. *International Journal of Production Research*, **29**(10), 2097-2116.

12. Burbidge, J.L., 1963. Production flow Analysis. *Production Engineer*, December, pp.742-752.

13. Burbidge, J.L., 1971. Production flow Analysis. *Production Engineer*, **50**, (4/5), 139-152.

14. Burbidge, J.L., 1975. *The introduction of group technology*, London: Heinemann Press.

15. Burbidge, J.L., 1977. A manual method of production flow Analysis. *Production Engineer*, **56**, pp.34-38.

16. Burke, L. and Kamal, S., 1992. Fuzzy ART for cellular manufacturing. *Proceedings of the Conference on Artificial Neural Networks in Engineering*, **1**, pp.779-784.

17. Burke, L. and Kamal, S., 1995. Neural networks and the part family/machine group formation problem in cellular manufacturing: a framework using fuzzy ART. *Journal of Manufacturing Systems*, **14** (3), pp.148-159.

18. Carpenter, G.A. and Grossberg, S., 1987. ART 2: Self-organization of stable category recognition codes for analog input patterns. *Applied Optics*, **26**, pp. 4919-4930.

19. Carpenter, G.A. and Grossberg, S., 1987. A massively parallel architecture for a self-organizing neural pattern recognition machine. *Computer Vision, Graphics, and Image Processing*, **37**(1), pp. 54-115.

20. Carpenter, G.A. and Grossberg, S., 1988. The ART of adaptive pattern recognition by a self-organizing neural network, *Computer*, **21**(3), pp.77-88.

21. Carpenter, G.A. and Grossberg, S., 2002. *A self-organizing neural network for supervised learning, recognition, and prediction*. In Thad A. Polk and Colleen M. Seifert (Eds.), *Cognitive Modeling*, Cambridge, MA: MIT Press, pp.288-314.

22. Carrie, A.S., 1973. Numerical taxonomy applied to group technology and plant layout. *International Journal of Production Research*, **11**(4), pp.399-416.

23. Chan, H.M. and Milner, D.A., 1982. Direct Clustering algorithm for group formation in Cellular Manufacturing. *Journal of Manufacturing Systems*, **1**, pp.65-75.

24. Chandrasekharan, M.P. and Rajagopalan, R., 1986a An ideal seed non-hierarchical clustering algorithm for cellular manufacturing. *International Journal of Production Research*, **24** (2), pp. 451-464.

25. Chandrasekharan, M.P. and Rajagopalan, R., 1986b. MODROC: An extension of rank order clustering for group technology. *International Journal of Production Research*, **24**(5), pp.1221-1233.

26. Chandrasekharan, M.P. and Rajagopalan, R., 1987. ZODIAC: An algorithm for concurrent formation of part families and machine cells. *International Journal of Production Research,* **25** (6), pp.451-464.

27. Chen, S.J. and Cheng, C.S., 1995. A neural network-based cell formation algorithm in cellular manufacturing, *International Journal of Production Research*, **33** (2), pp.293-318.

28. Chen. D.S., Chen. H.C. and Park J.M., 1996. An improved ART neural net for machine cell formation. *Journal of Material Processing Technology*, **61**, pp.1-6.

29. Cheng, C H., Maden, M S. and Motwani, J., 1996. Designing of cellular manufacturing systems by a truncated tree search. *International Journal of Production Research*, **34**(2), pp. 349-361.

30. Cheung, Y.M., 2003. K-means: A new generalized K-means clustering algorithm. *Pattern Recognition Letters*, **24**, pp.2883–2893

31. Choobineh, F., 1988. A frame work for the design of manufacturing systems", *International Journal of Production Research*, **26**, pp.1161-1172.

32. Chu, C.H. and Tsai, M., 1990. A Comparison of three array-based clustering techniques for manufacturing cell formation", *International Journal of Production Research*, **28**(8), pp.1417-1433.

33. Chu, C.H. and Hayya, J.C., 1991. A Fuzzy clustering approach to manufacturing cell formation. *International Journal of Production Research*, **29**(7), pp.1475-1488.

34. Chu, C.H., 1993. Manufacturing cell formation by competitive learning. *International Journal of Production Research*, 31(4), pp. 829-843

35. Chung, Y. and Kusiak, A., 1994. Grouping parts with a neural network. *Journal of Manufacturing Systems*, **134**, pp.262-275.

36. Co, H.C. & Araar, A., 1988, "Configuring cellular manufacturing system" *International Journal of Production Research*, **26**(9),1511-1522.

37. Crama, Y. and Oosten, M., 1996. Models for machine part grouping in cellular manufacturing. *International Journal of Production Research*, **34**(6), pp.1693-1713.

38. Dagli, C. and Huggahalli, R., 1991. Neural network approach to group technology. *Knowledge Based Systems and Neural Networks – Techniques and Applications*, R.Sarada, J.Y. Cheung, and N.J. Cochran (Eds) (New York: Elsevier), pp. 213-228.
39. Dagli, C. and Huggahalli, R., 1995. Machine-part family formation with adaptive resonance paradigm. *International Journal of Production Research*, **33**(4), pp.893-913.

40. Defays.D. 1977. An Efficient Algorithm for a Complete Link Method. *The Computer Journal*. 20(4), 364-366.

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

42. Dimopoulos, C., 2006. Multi-objective optimization of manufacturing cell design. *International Journal of Production Research*, **44**(22), pp.4855-4875.

43. Dobado, D., Lozano, S., Buno, J.M. and Larraneta, J., 2002. Cell formation using a Fuzzy Min-Max neural network. *International Journal of Production Research*, **40** (1), pp.93-107.

44. Enke, D., Ratanapan, K. & Dagli, C., 1998. Machine-part family formation utilizing an ART1 neural network implemented on a parallel neuro-computer. *Computers and Industrial Engineering*, **341**, pp.189-205.

45. Enke, D., Ratanapan, K. and Dagli, C., 2000. Large machine-part family formation utilizing a parallel ART1 neural network. *Journal of Intelligent Manufacturing*, **11**, pp.591-604.

46. Gallagher, C.C. and Knight, W.A., 1973. *Group Technology*. London: Butterworth Press.

47. George, A.P., Rajendran, C. and Ghosh, S., 2003. An analytical-iterative clustering algorithm for cell formation in cellular manufacturing systems with ordinal-level and ratio-level data. *International Journal of Advanced Manufacturing Technology*, **2**, pp.125–133.

48. Godfrey C. Onwabolu., 1999a. Manufacturing features recognition using back propagation neural networks. *Journal of Intelligent Manufacturing*, **10** (3/4), pp.289-299.

49. Godfrey C. Onwabolu., 1999b. Design of parts for cellular manufacturing using neural network-based approach. *Journal of intelligent manufacturing*, **10**(3/4), pp.251-265.

50. Goldberg, D.E., 1989. Genetic Algorithms in Search Optimization, and Machine Learning, Reading, MA: Addison-Wesley Press.

51. Goncalves, F.J. and Resende, M.G.C., 2002. A hybrid genetic algorithm in cell formation problems. AT&T Labs Research Technical Report TD-5FE6RN, AT&T Labs Research, Florham Park, NJ 07932 USA.

52. Goncalves.J.F and Resende M.G.C. 2004. An Evolutionary Algorithm for Manufacturing Cell Formation, *Computers and Industrial Engineering*, **47**(2), 3247-273

53. Gupta, T., 1991. Clustering algorithms for the design of a cellular manufacturing system – an analysis of their performance. *Computers and Industrial Engineering*, **20** (4), pp.461-468.

54. Gupta, Y., Kumar, A., Gupta, M. and Sundaram, C., 1996. A Genetic algorithm based approach to cell composition and layout design. *International Journal of Production Research*, 34 (2), pp. 447-482.

55. Ham, I., Hitomi, K. and Yoshida, T., 1985. *Group Technology: Applications to Production Management*. Boston: Kluwer-Nijhoff Press.

56. Han, C. and Ham, I., 1986. Multi-objective cluster analysis for part family formations. *Journal of Manufacturing System*, **5** (4), pp.223-229

57. Harhalakis, G., Nagi, R. and Proth, J.M., 1990. An efficient heuristic in manufacturing cell formation for group technology applications. *International Journal of Production Research*, **28** (1), pp.185-198.

58. Heragu, S. and Kakuturi, R.S., 1997. Grouping and placement of cells. *IIE transaction*, **29**, pp. 561-571.

59. Hiroshi ohta and Masateru Nakamura (2002). Cell Formation with reduction in setup times, *Computers and Industrial Engineering,* **42**, pp.317-327.

60. Holland, J.H., 1975. *Adaptation in Natural and Artificial Systems*. Ann Arbor, Michigan: The University of Michigan Press.

61. Hsu, C.M. and Su, T.C., 1998. Multi-Objective machine component grouping in cellular manufacturing-A Genetic algorithm approach. *Production Planning Control*, **9**, pp.155-166.

62. Iraj Mahdavi and Mahadevan.B 2008. CLASS: An algorithm for cellular manufacturing system and layout design using sequence data. *Robotics and Computer-Integrated Manufacturing*, **24**, 488–497

63. Iri, M., 1968. On the synthesis of loop and cutest matrices and the related SAAC memories, 4 A-XIII 376.

64. Jang.J-S.R., Sun.C-T. and Mizutani.E., 2002. *Neuro-Fuzzy and Soft Computing*, New Delhi: Prentice Hall Press.

65. Jayaswal, S. and 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.

66. Jeffrey Schaller, E., 2005. Tabu search procedures for the cell formation problem with intra-cell transfer costs as a function of cell size. *Computers & Industrial Engineering,* **49** (3), pp.449-462.

67. Jun Wang and Yoshiyasu Takefuji. 1993. Neural networks in design and manufacturing. Singapore: World Scientific Publishers.

68. Kamal, S. and Burke, L., 1996. FACT: A new neural network-based clustering algorithm for group technology. *International Journal of Production Research*, **34** (4), pp.919-946

69. Kandiller, L., 1994, A Comparative study of cell formation in cellular manufacturing system. *International Journal of Production Research*, **32**, pp. 2395-2429

70. Kao, Y. and Moon, Y.B., 1990. Learning part families by the back-propagation rule of neural networks. *Proceedings of the First International Conference on Automation Technology*, National Chiao Tung University, Hsinchu, Taiwan, **1** pp.819-824.

71. Kao, Y. and Moon, Y. B., 1991a. A unified group technology implementation using the backpropagation learning rule of neural networks. *Computers and Industrial Engineering*, **20** (4), pp.425-437.

72. Kao, Y. and Moon, Y. B., 1991b. A unified group technology implementation using ART-1 neural network model. *Proceedings of the International Conference on Computer Integrated Manufacturing*, GINTIC Institute of CIM, Nanyang Technological University, Singapore, pp.279- 282.

73. Kao, Y. and Moon Y.B., 1998. Feature-based memory association for group technology. *International Journal of Production Research*, **36** (6), pp.1653-1677

74. Kaparthi, S. and Suresh, N.C., 1991. A neural network system for shape-based classification and coding of rotational parts. *International Journal of Production Research*, **29** (9), pp.1771-1784.

75. Kaparthi, S. and Suresh, N.C., 1992. Machine-component cell formation in group technology: a neural network approach. *International Journal of Production Research*, **30** (6), pp.1353-1367.

76. Kiang, M.Y., Kulkarni, U.R. and Tam, K.Y., 1995. Self-organising map network as an interactive clustering tool – an application to group technology. *Decision Support Systems*, **15**, pp.351-374

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

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

79. Kumar, S.C. and Chandrasekharan, M.P., 1990. Grouping efficacy: a quantitative criterion for goodness of block diagonal forms of binary matrices in group technology. *International Journal of Production Research*, **28**(2), pp.233-243.

80. Kusiak, A., 1985. The part families problem in flexible manufacturing systems. *Annals of Operations Research*, **3**, pp.279-300.

81. Kusiak, A., 1987. The generalized group technology concept. *International Journal of Production Research*, **25**(4), pp.561-569.

82. Kusiak, A. and Ibrahim, W.I., 1988. Knowledge-Based system for group technology (KBGT). *IEEE*, 0-8186-0888-9/88/0000/0184.

83. Kusiak, A. and Chung, Y., 1991. GT/ART: Using neural networks to form machine cells. *Manufacturing Review*, **44**, pp.293-301.

84. Kusiak, A. and Lee, H., 1996. Neural computing-based design of components for cellular manufacturing. *International Journal of Production Research*, **34**(7), pp.1777-1790.

85. Kyung-Mi Lee, Takeshi Yamakawa and Keon-Myung Lee, 1997. Machine-part grouping for cellular manufacturing systems: a neural network approach. *First International Conference of Knowledge-Based Intelligent Electronic Systems*, Adelaide, Australia.

86. Lee, H. and Garcia-Diaz, A., 1996. Network flow procedures for the analysis of cellular manufacturing systems. *IIE Transactiions*, **28**, pp.333-345.

87. Liao, T.W. and Chen, L.J., 1993. An evaluation of ART1 neural models for GT part family and machine cell forming. *Journal of Manufacturing Systems*, **124**, pp.282-290.

88. Liao, T.W. and Lee, K.S., 1994. Integration of a feature-based CAD system and an ART1 neural model for GT coding and part family forming. *Computers and Industrial Engineering*, **26**(1), pp.93-104.

89. Logendran, R., Ramakrishnan, P. and Sriskandarajah, C., 1994. Tabu search based heuristics for cellular manufacturing system in the presence of alternate process plan. *International Journal of Production Research*, **33**, pp.675-695.

90. Lozano, S., Onieva, L., Larraneta, J. and Teba, J., 1993. A neural network approach to part-machine grouping in GT manufacturing. *Robotics, Mechatronics and Manufacturing Systems*, Eds.North-Holland, Amsterdam, (ISBN: 0-444-89700-3). pp.619-624.

91. Malakooti, B. and Yang, Z., 1995. A variable-parameter unsupervised learning clustering neural network approach with application to machine-part group formation. *International Journal of Production Research*, **33**(9), pp. 2395-2413

92. Malave, C.O. and Ramachandran, S., 1991. Neural network-based design of cellular manufacturing systems. *Journal of Intelligent Manufacturing*, **25**, pp. 305-314.

93. McAuley, J., 1972. Machine grouping for efficient production. *Production Engineer*, **51** (2), 53-57.

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

95. Melody Y. K., 2001. Extending the Kohonen self-organizing map networks for clustering analysis. Computational Statistics and Data Analysis, **38**, pp.161-180.

96. Miltenburg, J. and Zhang. W., 1991. A comparative evaluation of nine well-known algorithms for solving the cell formation problem in group technology. Journal of Operations Management, **10** (1), pp.44 -72.

97. Ming-Laing Chen, Chung-Min Wu and Chuen-Lung Chen, 2002. An integrated approach of ART1 search to solve cell formation problems. Journal of the Chinese Institute of Industrial Engineers, **19** (3), pp.62-74.

98. Michalewicz, Z. 1996. Genetic Algorithms + Data Structures = Evolution Programs. New York: Springer-Verlag.

99. Mitrofanov, S.P., 1959. Russian Text; also 1966 Scientific Principles of Group Technology. Part I, Boston: National Lending Library of Science and Technology.

100. Mitsuo Gen and Runwei Cheng, 2000. Genetic Algorithms and Engineering Optimization. New York: John Wiley & sons.

101. Moon, Y.B., 1992. Establishment of neuro computing model for part family/machine group identification. *Journal of Intelligent Manufacturing*, **3**, pp.173-182.

102. Moon, Y.B. and Chi, S.C., 1992 Generalized part family formation using neural network techniques. *Journal of Manufacturing Systems*, **11** (3), pp. 149-159.

103. Moon, Y.B. and Roy, U., 1992. Learning group-technology part families from solid models by parallel distributed processing. *International Journal of Advanced Manufacturing Technology*, **7** (2), pp.109-118

104. Mosier, C.T. and Taube, L., 1985. Weighted similarity measure heuristics for the group technology machine clustering problem. *OMEGA*, **13** (6), pp.577-583.

105. Murthy C.V.R. and Srinivasan G., 1995. Fractional Cell formation in-group technology. *International Journal of Production Research*, **33** (5), pp.1323-1337.

106. Nair, G.J. and Narendran, T.T., 1996. Grouping index: a new quantitative criterion for goodness of block-diagonal structure forms in group technology. *International Journal of Production Research*, **34** (10), pp.2767-2782.

107. Nair, G.J. and Narendran, T.T., 1998. CASE: A clustering algorithm for cell formation with sequence data. *International Journal of Production Research*, **36** (1), pp157-179.

108. Nair G.J and Narendran T.T., 1999. ACCORD: a bicriterion algorithm for cell formation using ordinal and ratio level data. *International Journal of Production Research,* **37**(3),pp.539–556

109. Panneerselvam, R. and Balasubramanian, K.N., 1985. Algorithm grouping of operations sequences. *Engineering costs and production economics*, **9**, pp. 125-135.

110. Park,S. and Suresh, N.C., 2003. Performance of Fuzzy ART neural network and hierarchical clustering for part-machine grouping based on operation sequences. *International Journal of Production Research*, 41(14), pp.3185-3216

111. Peker, A. and Kara, Y., 2004. Parameter setting fuzzy ART neural network to part-machine cell formation problem. *International Journal of Production Research*, **42** (6), pp.1257-1278.

112. Prabhakaran, G., Janakiraman, T.N. and Sachithanandam, M., 2002. Manufacturing data based combined dissimilarity coefficient for machine cell formation. International Journal of Advanced Manufacturing Technology, **19**, pp.889–897.

113. Purcheck, G.F.K., 1975. A mathematical classification as a basis for the design of group-technology production cells. *Production Engineer*, **54**, pp.35-48.

114. Rajagopalan, R. and Batra, J.L., 1975. Design of Cellular production system –A graph theoretic approach. *International Journal of Production Research*, **13** (6), pp.567-579.

115. Rajamani, D., Singh, N. and Aneja, Y.P., 1990. Integrated design of cellular manufacturing systems in the presence of alternate process plans. *International Journal of Production Research.* **28**, pp.1541-1554.

116. Ramabhatta,V. and Nagi,R., 1998. An Integrated formulation of manufacturing cell formation with capacity planning and multiple routings. *Annals of Operations Research*, **77**, pp.79-95.

117. Rao, H.A. and Gu, P., 1995. A multi-constraint neural network for the pragmatic design of cellular manufacturing systems. *International Journal of Production Research*, **33** (4), pp.1049-1070.

118. Sankaran, S. and Kasilingam, R.G., 1993. On cell size and machine requirements planning in group technology systems. *European Journal of Operations Research*, **69**, pp.373-383.

119. SaravanaSankar, S., 2005. *Evolutionary algorithms for multi-objective scheduling optimization of flexible manufacturing system*. An unpublished thesis, submitted at M.S.University, Tirunelveli.

120. Schonberger, R., 1996. World Class Manufacturing: The Next Decade. NewYork, NY: The Free Press.

121. Seifoddini, H. and Wolfe, P.M., 1986. Application of the similarity coefficient method in group technology. *IIE Transactions*, **183**, pp.271-277.

122. Seifoddini, H., 1989. Duplication process in machine cells formation in group technology. *IIE Transactions*, pp.382-388.

123. Seifoddini, H. and Hsu, C.P., 1994. Comparative study of similarity coefficients and clustering algorithms in cellular manufacturing. *Journal of Manufacturing Systems*, **13** (2), 119-127.

124. Seifoddini, H. and Djassemi, M., 1996. A new grouping measure for evaluation of machine-component matrices. *International Journal of Production Research*, **34**(5), pp.1179-1193.

125. Shafer, S.M. and Rogers, D.F., 1993a, Similarity and distance measures for cellular manufacturing, *International Journal of Production Research*, **31** (5), 1131-1142.

126. Stanfel, L., 1985. Machine clustering for economic production. *Engineering Costs and Production Economic*, **9**, pp.73-81

127. Singh, N., 1993. Design of cellular manufacturing systems: An invited review. *European Journal of Operational Research*, **69**, pp.284-291

128. Singh, N. and Rajamani, D., 1996. *Cellular Manufacturing Systems: Design, Planning and Control*, London: Chapman and Hall Press. pp.73-81.

129. Sinha, N.K., Gupta, M.M. and Zadeh, L.A., 2000. *Outline of Computational Theory of Perceptions Based on Computing with Words, Soft Computing & Intelligent Systems*. New York: Academic Press

130. Sofianopoulou, S., 1999. Manufacturing cells design with alternative process plans and / or replicate machines. *International Journal of Production Research*, **37**(3), pp.707-720.

131. Solimanpur, M., Vrat, P. and Shankar, R., 2004, Ant colony optimization algorithm to the inter-cell layout problem in cellular manufacturing, *European Journal of Operational Research*, **157** (3), pp.592-606.

132. Sridhar, J. and Rajendran, C., 1994. A Genetic Algorithm for Family and Job Scheduling in a Flowline based Manufacturing Cell. *Computers and Industrial Engineering*, **27**(1-4), pp.469-472

133. Srinivasan, G., Narendran, T. T., and Mahedevan, M., 1990. An assignment model for the part-families problem in group technology. *International Journal of Production Research*, **29**, pp.463-478.

134. Srinivasan, G. and Narendran, T.T., 1991. GRAFICS: a non-hierarchical clustering algorithm for group technology. *International Journal of Production Research*, **29** (3), pp.463-478.

135. Srinivasan, G., 1994. A clustering algorithm for machine cell formation in group technology using minimum spanning trees. *International Journal of Production Research*, 32 (9), 2149-2158

136. Sung-Lyong and Wemmerlov, U., 1993. A work load oriented heuristic approach cell formation allowing reallocation of operations. *European Journal of Operation Research*, **69**, pp.292-311.

137. Sung Youl Lee and Fischer, G.W., 1999. Grouping parts based on geometrical shapes and manufacturing attributes using a neural network. *Journal of Intelligent Manufacturing*, **10**(2), pp.199-209.

138. Suresh, N.C. and Kaparthi, S., 1994. Performance of fuzzy ART neural network for group technology cell formation. *International Journal of Production Research*, **32** (7), pp.1693-1713

139. Suresh, N.C., Slomp, J. and Kaparthi, S., 1999. Sequence-dependent clustering of parts and machines a fuzzy ART neural network approach. *International Journal of Production Research*, **37** (12), pp.2793-2816.

140. Taylor, D.G. and Taha, A., 1993. A comparison of buffering strategies for the CMS routing /sequencing problem. *European Journal of Operation Research*, **693**, pp.450-466.

141. Vakharia, A.J. and Wemmerlov, U., 1990. Designing a Cellular manufacturing system a material flow approach based on the operation sequences. *IIE Transactions*, **22** (1), pp.84-97.

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

143. Vannelli, A. and Kumar, K.R., 1986. A method for finding minimal bottleneck cells for grouping part machine families. *International Journal of Production Research*, **24** (2), pp.387-400.

144. Veeramani, D. and Mani, K., 1996. A polynomial-time algorithm for optimal clustering in a special class of {0, 1}-matrices. *International journal of production research*, **34**(9), pp.2587-2611.

145. Venkumar, P. and Haq, A.N., 2005. Manufacturing cell formation using modified art1 networks. *International Journal of Advanced Manufacturing Technology*, **26**, pp.906-916.

146. Venkumar, P. and Haq, A.N., 2006. Complete and fractional cell formation using Kohonen self organizing map networks in a cellular manufacturing system. *International Journal of Production Research*, **44**(20), pp.4257-4271.

147. Venugopal, V. and Narendran, T.T., 1992a. Cell formation in manufacturing systems through simulated annealing. *European Journal of Operations Research*, **63**, pp.409-422.

148. Venugopal, V. and Narendran, T.T., 1992b. A Genetic algorithm approach to the machine component and grouping problem with multiple objectives. *Computers and Industrial Engineering*, **224**, pp.469-480.

149. Venugopal, V. and Narendran, T.T., 1992c. Neural network model for design retrieval in manufacturing systems. *Computers in Industry*, **20**, pp.11-23.

150. Venugopal, V. and Narendran, T.T., 1994. Machine-cell formation through neural network models. *International Journal of Production Research*, **32** (9), pp.2105-2116

151. Venugopal, V. 1999. Soft-computing-based approaches to the group technology problem: a state-of-the-art review. *International Journal of Production Research*, **37**(14), pp.3335-3357.

152. Verma,P. and Ding,F.Y., 1995. A sequence-based materials flow procedure for designing manufacturing cell. *International Journal of Production Research*, **33**, pp.3267-3281.

153. Vitanov, V., Tjahjonol, B. and Marghalany, 2007. A decision support tool to facilitate the design of cellular manufacturing layouts. *Computers and Industrial Engineering*, **52**(4), pp.380-403.

154. Vohra, T., Cheng, D. and Chang, J.L., 1990. A network model for cell formation in CMS. *International Journal of Production Research*, **28**(11), pp.2075-2084.

155. Waghodekar, P.H. and Sahu, S., 1984. Machine-component cell formation in group technology. *MACE*. *International Journal of Production Research*, **22** (6), pp.937-948.

156. Wei, J.C. and Kern, G.M., 1991. Discussion: reply to "A note on a linear cell clustering algorithm. *International Journal of Production Research*, **29**(1), pp.217-218.

157. Wemmerlov, U. and Hyer, N.L., 1986. The part family machine grouping identification problem in cellular manufacturing. *Journal of Operations Management*, **6**, pp.125-134.

158. Wemmerlov and Hyer. 1987. Research issues in cellular manufacturing, *International Journal of Production Research*, **25**, 413–431

159. Won,Y. and Lee, K.C., 2001. Group technology cell formation considering operation sequence and production volumes. *International Journal of Production Research*, **39**(13), pp.2755-2768.

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

161. Wu, Low and Wu, 2004, A tabu search approach to the cell formation problem, *International Journal of Advanced Manufacturing Technology*, **23**, pp.916-924.

162. Xu, H. and Wang, H.P. 1989. Part family formation for GT applications based on fuzzy mathematics, *International Journal of Production Research,* **27** (9), 637–1651.

163. Yasuda, K., Hu, L. and Yin.Y., 2005. A grouping genetic algorithm for the multi-objective cell formation problem. *International Journal of Production Research*, **43** (4) pp.829-853.

164. Yin.Y and Yasuda.K. 2002. Manufacturing cells' design in consideration of various production factors, **40** (4), pp.885-906.

165. Yoh-Han Pao, 1989. *Adaptive Pattern Recognition and Neural Networks*, New York: Addison –Wesley Publishing Company, Inc.

166. Zahir Albadawi, Hamdi Bashir, A. and Mingyuan Chen, 2005. A mathematical approach for the formation of manufacturing cells. *Computers & Industrial Engineering*, **48** (1), pp.3–21.

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

168. Zolfaghari, S. and Liang, M., 1997. An objective-guided ortho-synapse Hopfield network approach to machine grouping problems. *International Journal of Production Research*, **35** (10), pp.2773-2792

169. Zolfaghari, S. and Liang, M., 2003. A new genetic algorithm for the machine/part grouping problem involving processing times and lot sizes. *Computers and Industrial Engineering*, **45**, pp.713-731.

APPENDIX I

ADVANTAGES OF CELLULAR MANUFACTURING SYSTEMS

The traditional type of organization for manufacturing is process organization in which each of the organizational units specializes in a particular process. This is gradually being replaced by CMS in which the organizational units (groups) complete all the parts they make at their particular processing stage and are equipped with all the machines and other facilities that they need to do so. CMS addresses many benefits like,

(i) Setting time reduction
(ii) Reduced throughput time due to reduced lead time.
(iii) Improved ability to follow market changes
(iv) Reduced stocks and WIP
(v) Centralization of responsibilities
(vi) Reduced handling and setting costs
(vii) Simplification of paper work
(viii) Reduced indirect labour-better cost analysis
(ix) Improved human relation and better communication
(x) Reduced investment per unit output.
(xi) Improved Quality of work
(xii) Reduced material obsolescence
(xiii) Reduced material costs
(xiv) Reduced indirect labour
(xv) Elimination of inter-departmental stores.

LIMITATIONS OF CELLULAR MANUFACTURING SYSTEMS

CMS has the following limitations:

(i) Difficult load balancing
(ii) Difficulty in batch size selection
(iii) Bottleneck machines are allotted to the remainder shop

(iv) Hardly any factory product range can be divided into clear cut component families.
(v) Job satisfaction may fail due to reduced variety of job processed in cells.
(vi) Stocks and WIP are high, as machines require their own cell rather than the common shop as in functional layout.
(vii) Production control coordinating the various cells is difficult.

Implementing cellular manufacturing is not as easy as it looks. It involves a series of vital steps so that it can be ensured whether the parts are actually produced by the system. The scale of the cellular manufacturing implementation varies from manufacture to manufacture depending on the scale of the business and on the objectives of the firm.

The following points have to be considered by the manufacturer to implement cellular manufacturing successfully.

A manufacturer has to organize parts that have similar characteristics into part families. Each family should produce higher volumes because the higher the production volume, the more efficient the production process within a cell. After parts are assigned to certain cells, a firm has to know what types of machines are required for each cell to produce parts. Some machines may need to be purchased where parts from different cells require the same machine.

In this step, workers must be trained and educated. This is the most important step in cellular manufacturing implementation. In cells, workers are required to operate multiple machines and take more responsibilities. Therefore, continuous training and education programs are necessary for improving manufacturing productivity. Moreover, trained workers and supervisors have to be given more responsibility. Rules, management policies, management techniques, or compensation system may have to be changed before cellular manufacturing actually starts working. These changes are critical for cellular manufacturing to be fully beneficial. The final step is to relocate the machine to begin production in the cells. Although the amount of time and

cost depend on the rearrangement, the rearrangement should avoid conflicts with any other production lines.

APPLICATIONS OF CELLULAR MANUFACTURING SYSTEMS

A cellular manufacturing system is slowly gaining hold in the industries all over the world. Gallagher and Knight (1973) observes that GT principles have been applied in many fields including machining, welded fabrications, foundry work, presswork, forging, plastic injection moulding etc. The reasons for GT's popularity lie in various achievements about its implementation by the industries. Applications include many types of industries like metal processing industries, equipment industries, electrical and electronic products oriented companies and automobile part manufactures.

Japan makes extensive use of cellular manufacturing in order to achieve Just-In-Time (JIT) Manufacturing (Schonberger 1996). In the last few decades the US and European companies have also learnt and implemented successfully the Japanese strategy.

This grouping philosophy has been widely used in Flexible Manufacturing System (FMS) (Kusiak, 1985) and in JIT production. Schonberger (1996) has identified five sub problems to be solved before production begins in the FMS. They are part type selection problem, machine grouping problem, part mix ratio problem, resource allocation problem and machine loading problem.

Another important conclusion was that consultants should be part of the process to introduce new concepts, establish goals, and prepare and execute plans. At a certain point however, they should be phased out so the internal employee team can finish developing and implementing the plans, fostering ownership, reliance on textbook solutions should be avoided. A blend of textbook concepts and real-life experience will produce a plan that is realistic, achievable, and sustainable.

The implementation of the factory of the future was a major undertaking for the industry (Sridhar and Rajendran 1994). New manufacturing strategies were introduced — cells, JIT receipt of materials, production per customer order, DNC—and involved a division wide team effort with full corporate support

Cellular manufacturing is gaining increasingly popularity as a way to improve productivity and competitiveness quickly. It becomes necessary to address the issues related to CMS.

APPENDIX II

SOFT COMPUTING TECHNIQUES FOR CELL FORMATION

Since cell formation problems are non-polynomially complete in nature, it is difficult to obtain solutions that satisfy all constraints. Therefore, it is expected to make use of simple but efficient computing techniques. Soft computing technique is found more suitable for such type of problems and capable of producing good results. Soft computing is an emerging approach to computing which parallels the remarkable ability of the human mind to reason and learn in an environment of uncertainty and imprecision (Jang et al. 2002). Soft computing is an innovative approach for constructing computationally intelligent systems. It is realized that complex real world problems require intelligent systems that combine knowledge, techniques and methodologies from various sources. Thereby it is more appropriate to make use of soft computing techniques like neural network, fuzzy sets for cell formation problem

ADAPTIVE RESONANCE THEORY (ART) NETWORKS

ART is an unsupervised Artificial Neural Network (ANN) similar to a real brain. The applications of ART mainly are for classification and pattern recognition. ART is better regarding its speed and accuracy compared with other ANN. Another advantage is its plasticity. ART is able to remember new input patterns without forgetting the previous/old input patterns. Nevertheless, ART is not so popular due to its high complexity and unpredictability to tune the learning parameter. ART consists of two layers: recognition layer and comparison layer. The input patterns will be saved at recognition layer (short-term memory) then the patterns at this recognition layer will be compared with the patterns at comparison layer (long-term memory). If a matching pattern was not found, this input pattern will be classified as a new pattern. The development of ART model leads to ART-2, ART-3, Fuzzy ART, ARTMAP (supervised) and Multi channel Fuzzy ART.

APPENDIX III

PERFORMANCE MEASURES

There are several performance measures proposed by the researchers in last two decades. The following are some of the grouping measures found in the benchmark literatures. Each of them has its own advantages and drawbacks depending on the data considered for CF problem. However, no grouping efficiency can be considered for the generalized cell formation with maximum available information. In this work, a new grouping measure is proposed for measuring the goodness of the block diagonal output matrix with ratio level data, ordinal level data and both combined data.

a) Grouping Efficiency (η)

Chandrasekharan and Rajagopalan (1986a) have developed grouping efficiency - a very first performance measure in CF. The higher grouping efficiency will result in better grouping. The efficiency was proposed as a weighted average of two efficiencies .

b) Grouping Efficacy (τ)

Kumar and Chandrasekharan (1990) have introduced grouping efficacy as a new performance measure, which has been proposed to overcome the drawbacks of grouping efficiency. High grouping efficacy will result as good CF.

c) Grouping Capability Index (GCI)

Seifoddini and Hsu (1994) introduced a new performance measure: grouping capability index (GCI), which is defined based on exceptional operations (Seifoddini and Djassemi 1996)). They considered the GCI as follows:

$$GCI = 1 - \frac{E}{o}$$

Unlike group efficiency and group efficacy, *GCI* exclude zero entries from the calculation of grouping efficacy

d) Machine utilization

Machine utilization (MU) indicates the percentage of times the machines within the clusters are used in production. MU is defined as

$$MU = \frac{N_o}{\sum_{k=1}^{k} M_k X N_k}$$

Where

N_o – total number of ones in the k_{th} cell

M_k – number of machines in the k^{th} cell

N_k – number of jobs in the k^{th} cell.

e) Total Bond energy

Measures of effectiveness (ME) is defined by

$$ME = \frac{1}{2} \sum_{i=1}^{M} \sum_{j=1}^{N} A_{ij} \left[A_{i(j+1)} + A_{i(j-1)} + A_{i(i+1)j} + A_{(i-1)j} \right]$$

where M – number of rows in binary matrix

N–number of columns in binary matrix

A_{ij} –1 if i^{th} machine is required by j^{th} part, 0 otherwise

f) Global efficiency

It is the ratio of the total number of operations that are performed within the suggested cells to total number of operations in the systems.

g) Group efficiency

It is the ratio of difference between total number of maximum external cells that could be visited and total number of external cells actually visited by all parts to total number of maximum external cells that could be visited.

h) Group technology efficiency

It is the ratio of difference between maximum number of inter-cell travels possible and number of inter-cell travels actually required by the system to the maximum number of inter-cell travels possible.

i) Grouping index

Nair and Narendran (1996) incorporated, in addition to diagonal space and a weighing factor (A) and derived a new measure called Grouping index.

$$\gamma = \frac{1 - \frac{qe_v + (1-q)(e_o - A)}{B}}{1 + \frac{qe_v + (1-q)(e_o - A)}{B}}$$

$$\gamma = \frac{1-\alpha}{1+\alpha_k}$$

Where

$$\alpha = \frac{qe_v + (1-q)(e_o - A)}{B}$$

e_0– number of ones in the off-diagonal block

e_v– number of voids in the diagonal block

q – weighing factor, $(0 \le q \le 1)$

B – block diagonal space (total number of elements in the diagonal block)

A = 0 for $e_0 < B$

A = e_0–B for e_o>B

j) Weighted Grouping efficiency

A weighting factor may be considered for each machine within the cell to get a better distribution of workload and by varying the weights on machines

$$\gamma = 1 - \frac{qe_v + (1-q)e_o}{qd_1 + (1-q)e_o}$$

where

e_0= number of ones in the off-diagonal block

e_v= number of voids in the diagonal block

d_1=total number of elements in the diagonal block

q= weighing factor, $(0 \le q \le 1)$

k) Quality Index

Quality Index (QI) is defined as the ratio of the intercellular workload (ICW) to the total Plant's Workload. (PW)

$$ICW = \sum_{i=1}^{k}\left[\sum_{i=1}^{M} X_{il}\left(\sum_{j=1}^{N}(1-Y_{jl})Z_{ij} \times V_j \times T_{ij}\right)\right]$$

X_{il} -1 if machine i is assigned to machine cell *I*, 0 otherwise

Y_{jl} -1 if part j is assigned to machine cell *I*, 0 otherwise

Z_{ij} -1 if part j has operations on machine i, 0 otherwise

V_j- production volume for part j

T_{ij}- Processing time of part j on machine i

$$PW = \sum_{i=1}^{M} \sum_{j=1}^{N} Z_{ij} \times V_j \times T_{ij}$$

The Quality Index (QI) for a block diagonal machine component matrix is calculated as

$$QI = 1 - \frac{ICW}{PW}$$

Performance measures and their source

Performance measure	Citation
Grouping Efficiency (η)	Chandrasekharan and Rajagopalan (1986a)
Number of Bottleneck Parts	Seifoddini and Wolfe (1986)
Machine utilization	Chandrasekharan and Rajagopalan (1986a)
Number of Bottleneck Machines	Kumar and Vannelli, (1987)
Grouping Efficacy (τ)	Kumar et al, (1990)
Global efficiency, Group efficiency, Group technology efficiency (GTE)	Harhalakis et al, (1990)
Number of Exceptional Elements (EE)	Boctor (1991)
Total Cell Load Variation (CLV)	Venugopal and Narendran (1992)
Grouping Capability Index	Seifoddini and Hsu (1994)
Total Moves (TM)	Gupta, et. al. (1996)
Bond efficiency	Nair and Narendran, (1998)
Grouping index	Nair and Narendran (1996)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50
1	0	0	0	0	0	0	0	.52	.910	0	0	0	0	0	0	0	0	0	0	0	.830	0	0	0	0	0	0	0	0	0	0	0	.910	0	0	0	0	0	0	0	.820	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.700	0	0	0	.580	0	0	0	0	0	0	0	0	0	.540	.540	0	0	0	0	0	0	.740	0	0	0	0	0	0	0	0	0	0	0
5	0	.63	.620	0	0	0	0	0	0	0	.520	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	.690	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.620	0	0	0	.680	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.500	0	0	0	0	0	0	0	.610	0		
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.940	0	0	0	0	0	0	0	.67	.67	.700	0	0	0	0	0	0	0	0	0	.840	0	0	0	0	0	0	0	0	0		
9	0	0	0	0	0	0	0	.920	0	0	0	0	0	0	0	0	0	0	0	0	.73	.910	0	0	0	0	0	0	0	0	0	0	0	0	.920	0	0	0	.920	0	0	0	0	0	0	.690				
11	0	0	0	0	0	0	0	.61	.540	0	0	0	0	0	0	0	0	0	0	0	0	0	.660	.690	0	0	0	0	0	0	0	0	0	.840	0	0	0	.910	0	0	0	0	0	0	0	0	0			
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.86	.670	0	.62	.741	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.780	0	0	0	0	0		
13	0	.55	.800	0	0	0	.62	.910	.530	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.520	0	0	.850	0	0	0	0	0			
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.54	.910	0	0	0	0	0	0	0	0	0	0	.92	.930	0	0	0	0	0	0	0	0	0	0	.800	0	0	.680	0			
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.520	0	0	0	0	0	0	0	0	0	.68	.900	0	.520	0	0	.750	0	0	0	0	0	0	0	0	0	0			
18	0	0	0	0	.91	.850	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.79	.670	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
22	0	0	0	0	0	0	0	0	0	0	0	0	.880	.650	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
23	0	0	0	0	0	.570	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.910	0	0	0	0	0	0	0	0	0	.880	0	.790	0	0	0	.710	0	0	0	0	0	.700	0	0	0				
24	0	0	0	0	0	0	0	0	0	0	0	0	.770	.520	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.910	0	0	0	0	0	.910	0	0	0	0	0	0	0				
26	0	0	0	0	0	0	0	0	0	.550	0	.620	0	0	.790	0	.810	0	0	0	0	0	0	0	0	0	0	0	0	0	.910	0	.72	.840	0	0	0	0	0	0	0	0	0	0	0	0	0			
27	0	0	0	0	0	0	0	0	0	0	0	0	.910	.550	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.550	0	0	0	0	0	0	0	.560	0	0	0	0	0	0		
29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.680	0	0	0	0	.89	.64	.570	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.510	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.540	0	0	.800	0	0	0	0	0	0	0	0	0	0			
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.830	.90	.910	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.850	0	0	0	0	0	0	.910	0	0	0			
3	.780	0	.560	0	0	0	0	0	0	0	0	0	0	0	0	.880	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.52	.510	0	0	0	.910	0	0	0	0	0	0	0	0	.830			
8	.780	0	.910	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.930	0	0	0	0	0	0	0	.840	0	.770	0	0	0	0			
25	.540	0	.840	0	0	0	0	0	0	0	0	0	0	0	0	.660	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.920	0	.850	0	0	.510	0	0	0	.600	0	0	0	0	0			
10	0	.89	.520	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.51	.530	0	0	0	0	0	0	0	0	0	0	0	0	.760	.760	0	0	0	0	0	.91	.590	0	0				
15	0	0	0	0	0	0	0	0	0	.590	.630	0	0	0	0	0	.690	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.890	0	0	0	0	.700	0	0	.98			
28	0	0	0	0	0	0	0	0	0	.810	0	0	.700	0	0	.560	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.820	0	0	0	.740	0	0	0	0			
17	0	0	0	0	.880	.790	0	0	0	.520	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.930	.780	0	0	0	0	0	.520	0	0	0	0	0	.720	0	0	0	0	.91				
19	.530	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.680	0	0	0	.590	.530	.650	0	0	0	0	0	0	0	0	0	0	0	0	0	.860	0	.74	.7	0	0	.77			
20	0	0	0	0	.84	.81	.630	0	0	0	.590	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.91	.520	.900	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
21	0	0	0	0	.910	.820	0	0	0	0	0	.770	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.910	0	0	.500	0	0	0	0	.820	0	0	0	0				

Row sorted matrix of size (30 x 50) Data set 24

	8	9	11	13	15	17	19	20	21	22	23	25	31	32	34	39	41	43	44	48	38	46	16	49	1	4	33	36	42	45	2	3	24	37	10	18	40	28	30	50	47	6	12	26	27	29	5	7	14	35
1	.52	.98	0	0	0	0	0	.83	0	0	0	0	.91	0	.82	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	.71	0	0	.58	0	0	.54	.54	0	.74	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	.52	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.63	.62	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	.62	0	0	.68	0	0	0	0	0	.5	0	0	0	.61	0	0	0	0	0	0	0	0	0	0	0	.69	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	.94	0	0	0	0	0	0	0	0	0	.84	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.67	.7	0	0	0	0	0	.67	0	0	0	0	0
9	.92	0	0	0	0	0	0	.73	.98	0	0	0	0	0	0	.92	0	0	0	0	0	0	0	.69	0	0	.92	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	.61	.54	0	0	0	0	0	0	0	.66	.69	0	0	0	.99	0	0	0	0	0	0	0	0	0	0	0	.84	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	.86	.67	0	.62	.85	0	0	0	0	0	0	.78	0	0	0	0	0	0	0	0	0	0	0	0	0	.74	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	.62	.97	.53	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.85	0	0	0	0	0	0	0	0	0	0	0	.55	.80	0	0	0	.52	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	.54	.98	0	0	0	.92	.93	0	0	0	.8	.68	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	.52	0	0	.68	.9	0	.75	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.52
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.85	0	.79	.67	0	.91	0	0	0
22	0	0	0	.88	.65	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0	.97	0	0	.88	0	.79	.71	0	0	0	0	0	0	.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.57	0	0	0	0	0	0	0	0
24	0	0	0	.77	.52	0	0	0	0	0	0	0	0	0	.95	0	.95	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
26	0	0	0	.62	0	.79	0	.81	0	0	0	.97	0	.72	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.55	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.84
27	0	0	0	.98	.55	0	0	0	0	0	0	0	0	.55	0	.56	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
29	0	0	0	0	0	.68	0	0	0	.89	.57	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.64	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.81	0	0	0	0	0	0	0	.51	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.54
2	0	0	0	0	0	.83	.9	.91	0	0	0	0	0	0	0	0	0	0	0	0	.85	.97	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.51	.97	0	0	0	0	0	0	.88	.83	.78	.56	.52	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.78	.99	.93	0	.84	.77	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.66	0	.54	.84	.92	.85	0	.61	0	0	0	0	0	.51	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	.51	0	0	0	.76	0	0	0	0	0	.96	0	0	0	0	0	0	0	0	.89	.52	.53	.76	0	0	0	0	0	.59	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.71	0	0	0	0	0	0	0	0	0	0	0	0	.59	.69	.89	0	0	.98	0	.63	0	0	0	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.74	0	0	0	.81	.56	.82	0	0	0	0	0	0	0	0	0	0	0	.7	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.72	0	0	0	0	0	0	0	0	0	0	0	0	0	.52	0	0	.93	.78	.91	0	.52	0	0	.88	.79	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	.68	0	0	0	0	0	0	0	0	.86	0	0	.74	0	.53	0	0	0	0	0	0	0	0	0	0	0	.53	.65	.77	.7	0	0	.59	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.81	.59	.95	.52	.9	.84	.63	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.51	0	0	0	0	0	0	0	0	0	0	0	0	0	.82	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.91	.82	.77	.99

k

Output matrix of size (30 x 50) with ten cells. (7 cells with single machine) Data set 24

	2	3	5	6	7	8	9	11	13	15	17	19	20	21	22	23	24	25	26	27	28	29	30	31	32	34	35	37	38	39	41	43	44	46	47	48	1	4	16	33	36	42	45	49	10	12	14	18	40	50
1	0	0	0	0	0	.53	.99	0	0	0	0	0	0	.83	0	0	0	0	0	0	0	0	0	0	0	.91	0	0	0	0	0	.82	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	.83	.91	.92	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.86	0	0	0	0	.97	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	.71	0	0	0	.58	0	0	0	0	0	0	0	0	0	.54	.54	0	0	0	0	0	.74	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	.63	.63	0	0	0	0	0	.53	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	.69	0	0	0	0	0	0	0	0	0	.63	0	0	0	.68	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.51	0	0	0	0	0	.61	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	.94	0	0	0	0	0	0	0	0	.68	.67	.7	0	0	0	0	0	0	0	.84	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	.93	0	0	0	0	0	0	0	.73	.98	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.92	0	0	0	0	0	0	0	0	0	.92	0	0	.7	0	0	0	0	0	0
10	.89	.52	0	0	0	0	0	0	0	0	0	0	0	0	0	.52	.54	0	0	0	0	0	0	0	0	0	0	.77	0	.76	0	0	0	.96	.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	.61	.54	0	0	0	0	0	0	0	0	.67	0	.7	0	0	0	0	0	0	0	0	0	0	0	0	.99	0	0	0	0	0	0	0	0	.85	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	.87	.67	0	0	.63	.74	.85	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.78	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	.55	.81	0	0	0	.63	.97	.54	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.85	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.52	0	
14	0	0	0	0	0	0	0	0	0	0	0	.55	.99	0	0	0	0	0	0	0	0	0	0	.93	.94	0	0	0	0	0	0	.8	0	0	.68	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	.53	0	0	0	0	0	0	0	0	0	.68	.91	0	.53	0	.76	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	.88	0	.79	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.94	0	.78	0	0	0	.52	0	0	0	0	.72	0	0	0	0	0	0	0	0	0	0	0	0	0	.52	0	0	.92	
18	0	0	.92	.86	0	0	0	0	0	0	0	0	0	0	0	0	0	.8	.67	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	.69	0	0	0	0	.59	0	.54	0	.66	0	0	0	0	0	0	0	0	.87	0	.74	.7	0	.53	0	0	0	0	0	0	0	0	0	0	0	.77	
20	0	0	.85	.81	.63	0	0	0	0	0	0	0	0	0	0	0	0	0	.96	.53	0	.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.6	0	0	0	0	
21	0	0	.92	0	.83	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.99	0	0	.51	0	0	0	0	0	0	0	0	0	0	0	.82	0	0	0	.78	0	0		
22	0	0	0	0	0	0	0	0	.89	.65	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23	0	0	0	.57	0	0	0	0	0	0	0	0	0	.97	0	0	0	0	0	0	0	0	0	.88	0	.8	0	0	0	.72	0	0	0	.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	0	.78	.53	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.95	0	0	0	.95	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0	.62	0	.8	0	.81	0	0	0	0	0	0	0	0	0	0	.98	0	.72	.84	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.55	0	0	0	0	0
27	0	0	0	0	0	0	0	0	.98	.55	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.56	0	0	0	0	0	.56	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
29	0	0	0	0	0	0	0	0	0	0	.69	0	0	0	.89	.64	.58	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.54	0	0	.81	0	0	0	0	0	0	0	0	0	.52	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.51	0	0	.98	0	0	0	0	0	0	0	.79	.56	.88	.53	0	0	.83	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.79	.99	0	.94	0	.84	.78	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.55	.84	.67	.92	.85	0	.61	0	0	0	0	0	.52	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.71	0	0	0	0	0	0	0	0	0	.6	.63	0	.7	.9	.98	
28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	.75	0	.82	0	.71	.57	.83	0	

Final output matrix of size (30 x 50) with three cells. Data set 24

APPENDIX V

GENETIC ALGORITHM - AN OVERVIEW

This is an introduction to genetic algorithm methods for optimization. Genetic algorithms were formally introduced in the United States in the 1970s by Holland (1975) at University of Michigan. The continuing performance improvement of computational systems has made them attractive for some types of optimization. In particular, genetic algorithms work very well on mixed (continuous *and* discrete), combinatorial problems (Mitsuo Gen and Runwei Cheng 2000). They are less susceptible to getting stuck at local optima than gradient search methods. But they tend to be computationally expensive.

In GA, it is must to represent a solution of the problem as a *genome* (or *chromosome*). The genetic algorithm then creates a population of solutions and applies genetic operators such as mutation and crossover to evolve the solutions in order to find the best one(s). This presentation outlines some of the basics of genetic algorithms.

The three most important aspects of using genetic algorithms are:

(i) Definition of the objective function.

(ii) Definition and implementation of the genetic representation.

(iii) Definition and implementation of the genetic operators.

Once these three have been defined, the generic genetic algorithm should work fairly well. Beyond that you can try many different variations to improve performance, find multiple optima (species - if they exist), or parallelize the algorithms.

The basic steps of a canonical GA are as follows.

Step 1. Initialize the population and enter step 2.

Step 2. Select individuals for recombination and enter step 3.

Step 3. Recombine individuals generating new ones and enter step 4.

Step 4. Mutate the new individuals and enter step 5.

Step 5. If the stopping criterion is satisfied, STOP; otherwise, replace old individuals with the new ones, restructure the population tree and return to step 2.

Genetic algorithm pseudo code:

{

Generate initial population P_t

Evaluate population P_t

while stopping criteria not satisfied **repeat**

{

Select elements from P_t to put into P_{t+1}

Crossover elements of P_t and put into P_{t+1}

Mutate elements of P_t and put into P_{t+1}

Evaluate new population P_{t+1}

$P_t = P_{t+1}$

}

}

Genetic Algorithm is a computerized search and optimization algorithm based on the mechanics of natural genetics and natural selection. GA is a search technique for global optimization in a search space. As the name suggests, they employ the concepts of natural selection and genetics using past information for directing the search with expected improved performance to achieve fairly consistent and reliable results. The traditional methods of optimization and search do not work well over a broad spectrum of problem domain. GA attempts to mimic the biological evolution process for discovering good solutions. They are based on a direct analogy to Darwinian natural selection and mutations in biological reproduction and belong to a category of heuristics known as randomized heuristics that employ randomized choice operators in their search strategy and do not depend on complete a priori knowledge of the features of domain. These operators have been conceived through abstractions of natural genetic mechanisms such as crossover and mutation and have been cast into algorithmic forms. Holland (1975) envisaged the concept of these algorithms in the mid-sixties and it has been applied in diverse areas such as music generation, genetic synthesis, strategy planning and also to address business problems such as traveling salesman problem, production planning and scheduling problem, facility location problem and cell design problems. GA is different from traditional optimization and search techniques in the following ways. It works with a coding of parameters; not with parameter themselves. GA searches from population of points; not from a single point. It uses probabilistic rules rather than deterministic rules.

APPENDIX VI

Source and size of the data sets

Data set No.	Source	Problem size
1	King and Nakornchai (1982)	5 x 7
2	Waghodekar and sahu (1984)	5 x 7
3	Seiffodini (1989)	5 x 18
4	Kusiak (1992)	6 x 8
5	Kusiak (1987)	7 x 11
6	Boctor (1991)	7 x 11
7	Seiffodini and wolfe (1986)	8 x 12
8	Chandrasekaran et al. (1986)a	8 x 20
9	Chandrasekaran et al. (1986)b	8 x 20
10	Mosier et al. (1985)	10 x 10
11	Chan et al. (1982)	10 x 15
12	Askin et al. (1987)	14 x 23
13	Stanfel (1985)	14 x 24
14	Srinivasan et al. (1990)	16 x 30
15	Mosier et al. (1985)	20 x 20
16	Carrie (1973)	20 x 35
17	Boe et al. (1991)	20 x 35
18	Kumar et al. (1986)	23 x 20
19	Mccornick et al. (1972)	24 x 16
20	Chandrasekaran et al. (1989)a	24 x 40
21	Chandrasekaran et al. (1989)b	24 x 40
22	Kumar et al. (1987)	30 x 41
23	Stanfel (1985)a	30 x 50
24	Stanfel (1985)b	30 x 50
25	Venugopal & Narendran (1992)	5 x 8
26	Venugopal & Narendran (1992)	7 x 11
27	Venugopal & Narendran (1992)	8 x 20
28	Srinivasan & Narendran (1991)	8 x 20
29	Venugopal & Narendran (1992)	9 x 9
30	Kusiak & Lee (1987)	10 x 15
31	Venugopal & Narendran (1992)	8 x 14
32	Venugopal & Narendran (1992)	9 x 10
33	Burbidge (1971)	12 x 31

Data set No.	Source	Problem size
34	Boctor (1991)-1	16 x 30
35	Boctor (1991)-2	16 x 30
36	Boctor (1991)-3	16 x 30
37	Boctor (1991)-4	16 x 30
38	Boctor (1991)-5	16 x 30
39	Boctor (1991)-6	16 x 30
40	Boctor (1991)-7	16 x 30
41	Boctor (1991)-8	16 x 30
42	Boctor (1991)-9	16 x 30
43	Boctor (1991)-10	16 x 30
44	Venugopal & Narendran (1992)	16 x 43
45	Venugopal & Narendran (1992)	10 x 20
46	Venugopal & Narendran (1992)	11 x 16
47	Venugopal & Narendran (1992)	16 x 43
48	Venugopal & Narendran (1992)	24 x 40
49	Venugopal & Narendran (1992)	24 x 40
50	Nair & Narendran (1998)	7 x 7
51	Nair & Narendran (1998)	20 x 8
52	Nair & Narendran (1998)	20 x 20
53	Nair & Narendran (1998)	40 x 25
54	Nair & Narendran (1999)	12 x 10
55	Sofianopoulou (1999)	5 x 4
56	Won and Lee (2001)	5 x 5
57	Generated-1	7 x 5
58	Generated-2	8 x 6
59	Park and Suresh (2003)	19 x 12
60	Sofianopoulou (1999)	20 x 12
61	Nagi et al. (1990)	20 x 20
62	Generated-3	30 x 15
63	Generated-4	37 x 20
64	Generated-5	50 x 25
65	Generated-6	55 x 20
66	Generated-7	60 x 28
67	Generated-8	65 x 30
68	Generated-9	80 x 32
69	Generated-10	90 x 35

http://www.freewebs.com/sudhakarpandian/169.htm (Data sets can be downloaded from web)

Data set - 1

	P1	P2	P3	P4	P5	P6	P7
M1	0	0.53	0	0.99	0.83	0.91	0
M2	0.82	0	0.83	0	0	0	0
M3	0.91	0	0.92	0	0	0.86	0.97
M4	0	0.79	0	0.56	0	0.88	0
M5	0.53	0	0	0	0.51	0	0.98

Data set - 2

	P1	P2	P3	P4	P5	P6	P7
M1	0.53	0	0	0	0.99	0.83	0.91
M2	0	0.82	0.83	0.91	0.92	0	0
M3	0	0	0.86	0.97	0.79	0.56	0
M4	0.88	0.53	0.51	0.98	0	0	0
M5	0	0.83	0	0.71	0.58	0.54	0

Data set - 3

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18
M1	0.53	0.99	0.83	0	0.91	0.82	0	0.83	0	0	0.91	0.92	0.86	0.97	0	0.79	0.56	0
M2	0.88	0	0.53	0.51	0	0.98	0.83	0.71	0	0.58	0.54	0.54	0.74	0	0.63	0	0	0.63
M3	0	0	0	0.53	0	0	0.69	0	0	0.63	0	0	0	0	0.68	0	0	0.51
M4	0.61	0.94	0.68	0	0.67	0.7	0	0.84	0	0	0.79	0.99	0.94	0.84	0	0.78	0.93	0
M5	0	0	0	0.73	0	0	0	0	0.98	0.92	0	0	0	0	0.92	0	0	0.7

Data set - 4

	P1	P2	P3	P4	P5	P6	P7	P8
M1	0	0.53	0	0.99	0	0	0.83	0
M2	0.91	0.82	0.83	0	0.91	0.92	0.86	0.97
M3	0	0	0.79	0	0	0.56	0	0.88
M4	0	0	0	0.53	0	0	0.51	0
M5	0.98	0	0.83	0	0.71	0.58	0	0.54
M6	0	0	0	0.54	0	0	0.74	0

Data set - 5

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11
M1	0	0.53	0.99	0	0	0	0.83	0	0	0	0
M2	0.91	0	0	0	0.82	0	0	0	0	0	0.83
M3	0	0	0	0	0	0	0	0	0	0.91	0.92
M4	0.86	0	0.97	0	0	0.79	0	0	0	0	0
M5	0	0	0	0	0.56	0	0	0.88	0	0	0
M6	0.53	0	0	0.51	0	0	0	0.98	0.83	0.71	0
M7	0	0	0.58	0.54	0	0.54	0.74	0	0.63	0	0

Data set - 6

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11
M1	0.53	0.99	0	0	0	0.83	0	0	0	0	0
M2	0	0.91	0	0	0	0.82	0	0	0.83	0	0
M3	0.91	0	0.92	0	0	0	0.86	0	0	0	0.97
M4	0	0	0.79	0	0	0	0.56	0	0	0	0
M5	0	0	0.88	0.53	0	0	0	0	0	0	0.51
M6	0	0	0	0.98	0.83	0	0	0	0	0.71	0
M7	0	0	0	0	0.58	0	0	0.54	0	0.54	0

Data set - 7

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12
M1	0.53	0.99	0.83	0.91	0	0	0	0	0	0	0	0
M2	0.82	0	0.83	0.91	0.92	0.86	0.97	0	0	0.79	0	0
M3	0	0	0.56	0.88	0.53	0.51	0.98	0.83	0.71	0	0	0
M4	0	0	0	0	0	0.58	0.54	0.54	0.74	0.63	0	0
M5	0	0	0	0	0	0	0.63	0.53	0.69	0.63	0	0
M6	0	0	0	0	0	0	0.68	0.51	0.61	0	0.94	0
M7	0	0	0	0	0	0	0	0	0	0	0.68	0.67
M8	0	0	0	0	0	0	0	0	0	0	0.7	0.84

Data set - 8

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20
M1	0.53	0	0.99	0.83	0	0	0	0	0.91	0.82	0	0	0	0.83	0.91	0.92	0	0.86	0.97	0
M2	0	0.79	0.56	0.88	0	0.53	0.51	0	0.98	0	0.83	0	0	0	0	0	0	0.71	0	0.58
M3	0	0	0	0	0.54	0.54	0.74	0.63	0	0	0.63	0.53	0.69	0	0	0.63	0.68	0	0.51	0.61
M4	0	0	0.94	0.68	0	0	0.67	0.7	0.84	0.79	0	0	0.99	0.94	0.84	0	0.78	0.93	0.73	0.98
M5	0.92	0.92	0.7	0	0	0.89	0	0	0	0.52	0	0.52	0.54	0	0.77	0.76	0.96	0	0.6	0.61
M6	0.54	0.67	0	0	0.7	0	0.85	0.99	0	0.87	0.67	0.63	0	0.74	0.85	0.78	0	0.55	0.81	0
M7	0	0	0	0	0.63	0.97	0.54	0.52	0	0	0.85	0.55	0.99	0	0	0.93	0.94	0	0.8	0.68
M8	0.6	0.63	0.7	0.9	0.71	0	0	0.98	0.53	0.68	0	0	0.91	0.53	0	0.76	0.88	0	0	0

Data set - 9

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20
M1	0	0.53	0.99	0	0	0	0	0.83	0.91	0	0.82	0	0.83	0.91	0	0.92	0.86	0	0.97	0
M2	0	0	0.79	0.56	0	0.88	0.53	0	0	0	0	0	0	0.51	0	0	0	0.98	0	0.83
M3	0	0.71	0	0	0	0	0	0.58	0.54	0	0.54	0	0.74	0.63	0	0.63	0.53	0	0.69	0
M4	0	0	0.63	0.68	0	0.51	0.61	0	0	0.94	0	0	0	0	0	0	0	0.68	0	0.67
M5	0.7	0	0	0	0.84	0.79	0	0	0	0.99	0	0.94	0	0	0.84	0	0.78	0	0	0
M6	0.93	0	0	0	0.73	0	0	0	0.98	0.92	0	0.92	0	0	0.7	0	0	0	0	0.89
M7	0	0	0.52	0.52	0	0.54	0.77	0	0	0	0.76	0.96	0	0	0	0	0	0.6	0	0.61
M8	0	0	0.54	0.67	0	0.7	0.85	0	0	0	0	0	0	0	0	0	0	0.99	0	0.87

Data set - 10

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
M1	0.53	0	0	0	0	0	0	0	0	0.99
M2	0	0	0.83	0.91	0	0	0	0.82	0	0
M3	0	0	0	0	0.83	0.91	0	0	0	0
M4	0.92	0	0	0	0	0	0	0	0	0
M5	0	0	0	0	0	0	0.86	0	0	0.97
M6	0.79	0	0	0	0	0	0.56	0	0	0.88
M7	0	0	0.53	0	0	0	0	0.51	0	0
M8	0	0	0	0	0	0.98	0	0	0.83	0
M9	0	0.71	0.58	0.54	0	0	0	0	0	0
M10	0	0.54	0.74	0.63	0	0	0	0.63	0	0

Data set - 11

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15
M1	0	0.53	0	0	0	0	0	0	0	0.99	0.83	0.91	0	0	0
M2	0	0	0.82	0	0.83	0	0	0.91	0	0	0	0	0.92	0	0.86
M3	0.97	0	0	0	0	0.79	0	0	0.56	0	0	0	0	0.88	0
M4	0.53	0	0	0.51	0	0	0	0	0.98	0	0	0	0	0.83	0
M5	0	0	0.71	0	0.58	0	0	0.54	0	0	0	0	0.54	0	0.74
M6	0.63	0	0	0.63	0	0.53	0	0	0.69	0	0	0	0	0.63	0
M7	0	0.68	0	0	0	0	0.51	0	0	0.61	0.94	0.68	0	0	0
M8	0	0	0.67	0	0.7	0	0	0.84	0	0	0	0	0.79	0	0.99
M9	0	0	0	0.94	0	0.84	0	0	0.78	0	0	0	0	0.93	0
M10	0	0.73	0	0	0	0	0.98	0	0	0.92	0.92	0.7	0	0	0

Data set - 12

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23
M1	0	0	0	0	0	0	0.53	0.99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M2	0	0	0	0.83	0.91	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M3	0	0	0	0.82	0.83	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.91	0	0
M4	0	0.92	0.86	0	0	0	0	0	0	0	0	0	0	0	0	0	0.97	0	0.79	0.56	0	0.88	0
M5	0	0.53	0.51	0	0	0	0	0	0	0	0	0	0	0	0	0	0.98	0	0	0.83	0	0.71	0
M6	0.58	0	0	0	0	0	0	0	0	0.54	0.54	0.74	0	0.63	0.63	0.53	0	0	0	0	0	0	0
M7	0	0.69	0.63	0	0	0	0	0.68	0	0	0	0	0	0	0	0	0.51	0	0	0.61	0	0	0
M8	0.94	0	0	0	0	0.68	0	0	0	0.67	0	0.7	0	0.84	0.79	0.99	0	0	0	0	0	0	0
M9	0	0	0	0	0	0.94	0	0	0	0.84	0	0.78	0.93	0	0.73	0	0	0	0	0	0	0	0
M10	0	0	0	0.98	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.92
M11	0	0	0	0.92	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.89	0	0
M12	0	0	0	0	0	0	0	0.52	0.52	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M13	0	0	0	0	0	0	0.54	0.77	0.76	0	0	0	0	0	0	0	0	0.96	0	0	0	0.6	0
M14	0	0	0	0	0	0	0	0	0	0.61	0.54	0	0.67	0	0.7	0	0	0	0	0	0	0	0

Data set - 13

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24
M1	0	0	0	0	0	0.53	0.99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M2	0	0	0.83	0.91	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M3	0	0	0.82	0.83	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.91	0	0	0
M4	0.92	0.86	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.97	0	0.79	0.56	0	0	0.88	0
M5	0.53	0.51	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.98	0	0	0.83	0	0	0.71	0
M6	0	0	0	0	0	0	0	0	0.58	0.54	0.54	0.74	0	0.63	0.63	0.53	0	0	0	0	0	0	0	0
M7	0.69	0.63	0	0	0	0	0.68	0	0	0	0	0	0	0	0	0	0.51	0	0	0.61	0	0.94	0	0
M8	0	0	0	0	0.68	0	0	0	0.67	0.7	0	0.84	0	0.79	0.99	0.94	0	0	0	0	0	0	0	0
M9	0	0	0	0	0.84	0	0	0	0.78	0	0	0.93	0.73	0	0.98	0	0	0	0	0	0	0.92	0	0
M10	0	0	0.92	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.7
M11	0	0	0.89	0.52	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.52	0	0	0.54
M12	0	0	0	0	0	0	0.77	0.76	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M13	0	0	0	0	0	0.96	0.6	0.61	0	0	0	0	0	0	0	0	0	0.54	0	0	0	0	0.67	0
M14	0	0	0	0	0	0	0	0	0.7	0	0.85	0	0.99	0	0.87	0	0	0	0	0	0	0	0	0

Data set - 14

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30
M1	0	0	0	0.53	0.99	0	0.83	0	0.91	0	0	0.82	0	0	0	0	0	0.83	0	0	0	0.91	0	0	0	0	0	0	0	0.92
M2	0.86	0	0	0.97	0	0	0	0	0	0.79	0	0	0	0	0	0.56	0	0.88	0	0.53	0	0	0	0	0	0	0	0	0	0
M3	0.51	0	0	0	0	0	0	0	0	0	0	0	0.98	0	0	0	0	0	0	0	0	0	0.83	0	0.71	0	0.58	0.54	0.54	0
M4	0	0.74	0	0.63	0	0	0.63	0	0.53	0	0	0	0	0	0	0	0	0.69	0	0	0	0.63	0	0	0.68	0	0	0	0	0.51
M5	0	0	0.61	0	0	0.94	0	0.68	0	0	0.67	0	0	0.7	0.84	0	0	0	0	0	0.79	0	0	0.99	0	0.94	0	0	0	0
M6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.84	0	0.78	0	0.93	0	0.73	0
M7	0	0.98	0	0.92	0	0	0.92	0	0	0	0	0.7	0	0	0	0	0	0.89	0	0	0	0.52	0	0	0.52	0	0	0	0	0.54
M8	0	0.77	0	0.76	0	0	0.96	0	0.6	0	0	0.61	0	0	0	0	0	0.54	0	0	0	0.67	0	0	0	0	0	0.7	0	0
M9	0	0	0	0	0.85	0	0	0	0	0	0	0	0	0	0	0	0	0	0.99	0	0	0	0.87	0	0.67	0	0.63	0.74	0	0
M10	0	0	0	0	0	0.85	0	0.78	0	0	0.55	0	0	0.81	0.63	0	0	0.97	0	0	0.54	0	0	0	0	0.52	0	0	0	0
M11	0	0.85	0	0	0	0	0.55	0	0.99	0	0.93	0.94	0	0	0	0	0	0.8	0	0	0	0	0	0	0	0	0	0	0	0.68
M12	0	0.6	0	0.63	0	0	0.7	0	0	0	0	0.9	0	0	0	0	0	0.71	0.98	0.53	0	0.68	0	0	0	0	0	0	0	0
M13	0	0	0.91	0	0	0	0	0	0	0.53	0	0	0.76	0	0	0	0	0	0	0.88	0	0	0	0	0	0	0	0	0.79	0.52
M14	0	0	0	0	0	0.94	0	0.78	0	0	0.52	0	0	0.72	0.92	0	0.92	0	0	0	0.86	0	0	0	0	0	0.8	0	0	0
M15	0	0	0	0	0.67	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.53	0	0.69	0.59	0.54	0	0.66	0.87
M16	0	0	0	0	0	0.74	0.7	0.77	0	0	0	0	0	0.85	0	0	0	0	0	0	0	0	0	0.81	0	0.63	0.6	0	0	0

Data set - 15

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20
M1	0.53	0	0	0	0	0	0.99	0	0	0	0	0	0	0	0	0	0	0.83	0	0.91
M2	0	0.82	0	0	0	0	0	0	0	0	0	0	0.83	0	0	0	0	0	0	0
M3	0	0.91	0.92	0	0.86	0	0	0	0	0	0	0	0	0	0.97	0	0	0	0	0.79
M4	0	0	0.56	0	0	0	0	0.88	0	0	0.53	0.51	0	0	0	0	0.98	0	0	0.83
M5	0	0.71	0	0	0	0.58	0	0	0	0	0.54	0	0	0	0.54	0.74	0.63	0	0	0
M6	0.63	0	0	0.53	0	0.69	0.63	0.68	0.51	0	0	0	0	0	0	0	0	0	0	0
M7	0	0	0	0	0	0	0.61	0	0	0	0	0	0	0	0.94	0	0.68	0.67	0	0
M8	0	0.7	0.84	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.79	0	0.99
M9	0.94	0	0	0.84	0	0	0.78	0.93	0.73	0	0	0	0	0	0.98	0	0	0	0	0
M10	0	0	0	0.92	0	0	0	0.92	0	0.7	0	0	0.89	0	0	0	0	0	0.52	0
M11	0.52	0	0.54	0	0	0	0	0	0	0.77	0	0.76	0.96	0	0	0.6	0	0	0	0
M12	0	0	0	0	0	0	0	0	0	0	0	0.61	0	0	0	0	0	0	0	0
M13	0.54	0	0.67	0	0.7	0	0	0	0	0.85	0.99	0	0.87	0.67	0	0	0	0	0	0
M14	0	0	0.63	0	0	0.74	0.85	0.78	0.55	0	0.81	0.63	0	0.97	0	0.54	0.52	0	0.85	0
M15	0	0	0	0	0.55	0	0	0.99	0	0.93	0.94	0	0	0	0	0	0.8	0	0	0
M16	0.68	0	0.6	0	0	0	0.63	0	0.7	0	0	0	0.9	0	0.71	0.98	0.53	0.68	0	0
M17	0	0.91	0	0	0	0	0	0	0	0	0	0.53	0.76	0.88	0	0	0	0	0	0
M18	0	0	0.79	0	0	0	0.52	0.94	0	0	0.78	0.52	0.72	0.92	0	0	0	0	0	0
M19	0	0	0	0	0	0	0.92	0	0.86	0.8	0.67	0	0	0.53	0	0	0	0	0	0.69
M20	0	0	0	0	0	0	0.59	0.54	0	0	0	0	0	0	0	0	0	0	0	0

Data set - 16

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30	P31	P32	P33	P34	P35
M1	0.53	0	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.83	0	0	0.91	0	0.82	0	0	0	0.83	0	0	0	0	0	0
M2	0	0.9	0	0	0	0	0.9	0	0	0.86	0	0.97	0.79	0	0	0	0	0.56	0	0	0	0	0	0.88	0	0	0.53	0	0	0	0.51	0	0	0	0
M3	0.98	0	0.8	0	0.7	0	0	0	0	0	0	0	0	0	0.58	0	0.54	0	0	0	0	0	0	0	0	0	0	0	0.54	0	0.74	0	0	0	0
M4	0	0.6	0	0	0	0	0.6	0	0	0	0	0.53	0.69	0	0	0	0	0	0	0	0	0	0	0.63	0	0	0.68	0	0	0	0	0	0	0	0
M5	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0.61	0	0.94	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.68	0
M6	0	0	0	0	0	0	0	0.6	0	0	0	0	0	0.7	0	0.84	0	0	0.79	0	0	0.99	0	0	0	0.94	0	0	0	0	0	0	0	0.84	0
M7	0.78	0	0.9	0	0.7	0	0	0	0	0	0	0	0	0	0.98	0	0.92	0	0	0.92	0	0	0.7	0	0	0	0	0	0	0	0	0	0	0	0
M8	0.89	0	0.5	0	0.5	0	0	0	0	0	0	0	0	0	0.54	0	0.77	0	0	0.76	0	0	0.96	0	0.6	0	0	0	0.61	0	0	0	0	0	0
M9	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0.67	0	0	0	0	0.7	0	0	0.85	0	0	0	0.99	0	0	0	0	0	0	0	0	0
M10	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0.67	0	0.63	0	0	0.74	0	0	0.85	0	0	0	0.78	0	0	0	0	0	0	0	0	0
M11	0	0	0	0.5	0	0.81	0	0	0.6	0	0.97	0	0	0	0	0	0	0	0	0	0.54	0	0	0	0	0	0	0.52	0	0.85	0	0.55	0	0	0.99
M12	0	0	0	0.9	0	0.94	0	0	0.8	0	0.68	0	0	0	0	0	0	0	0	0	0.6	0	0	0	0	0	0	0	0	0	0	0	0.63	0	0
M13	0	0.7	0	0	0	0	0	0	0	0	0	0.9	0.71	0	0	0	0	0	0	0	0	0	0	0.98	0	0	0	0	0	0	0	0	0	0	0
M14	0	0.5	0	0	0	0	0.68	0	0	0.91	0	0.53	0.76	0	0	0	0	0.88	0	0	0	0	0	0.79	0	0	0.52	0	0	0	0.94	0	0	0	0
M15	0	0	0	0.7	0	0.5	0	0	0.7	0	0.92	0	0	0	0	0	0	0	0	0	0.92	0	0	0	0	0	0	0.86	0	0.8	0	0.67	0	0	0
M16	0	0	0	0.5	0	0.6	0	0	0.5	0	0.54	0	0	0	0	0	0	0	0	0	0.66	0	0	0	0	0	0	0.87	0	0.74	0	0.7	0	0	0
M17	0.77	0	0.8	0	0.8	0	0	0	0	0	0	0	0	0	0.63	0	0.6	0	0	0	0	0	0.96	0	0.53	0	0	0	0.9	0	0	0	0	0	0
M18	0	0.9	0	0	0	0	0	0	0	0.83	0	0.78	0.99	0	0	0	0	0.51	0	0	0	0	0	0.82	0	0	0	0	0	0	0.89	0	0	0	0
M19	0	0	0	0.6	0	0	0	0	0.5	0	0.97	0	0	0	0	0	0	0	0	0	0.88	0	0	0	0	0	0	0.8	0	0.72	0	0.7	0	0	0
M20	0	0	0	0	0	0	0	0.7	0	0	0	0	0	0.53	0	0	0	0	0.95	0	0	0	0	0	0	0.95	0	0	0	0	0	0	0	0	0

Data set - 17

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30	P31	P32	P33	P34	P35
M1	0.5	0	0	0	0	0.9	0	0	0	0	0.83	0	0	0.91	0.82	0	0	0.83	0	0.91	0	0	0.92	0.86	0.97	0	0	0	0	0.79	0	0.56	0	0.88	0.53
M2	0	0.5	0	0	0	0	0	0	0	0.98	0	0.83	0.71	0	0	0	0	0.58	0.54	0	0	0	0	0.54	0	0	0	0	0	0	0.74	0	0	0	0
M3	0.6	0	0.6	0	0.5	0	0	0	0	0	0	0	0	0	0.69	0	0	0	0	0	0	0	0	0	0	0	0	0	0.63	0	0.68	0	0	0	0
M4	0	0.5	0	0	0	0	0.6	0	0	0	0	0.94	0.68	0	0	0	0	0	0	0	0	0	0	0.67	0	0	0.7	0	0	0	0	0	0	0	0
M5	0	0	0	0	0.84	0	0	0	0	0	0	0	0	0.79	0	0	0.99	0	0	0	0	0	0.94	0	0	0.84	0	0	0	0	0	0	0	0	0
M6	0	0	0	0	0	0	0	0.7	0	0	0	0	0	0.93	0	0	0	0	0.73	0	0	0.98	0	0	0.92	0	0	0	0	0.92	0	0	0	0	0
M7	0.7	0	0.8	0	0.5	0	0.5	0	0	0	0	0.54	0	0	0.77	0	0.76	0	0.96	0.6	0	0.61	0.54	0.67	0	0.7	0	0	0.85	0.99	0.87	0.67	0	0	0
M8	0.6	0	0	0	0.7	0	0	0	0.85	0	0	0	0	0	0.78	0	0.55	0	0	0.81	0	0	0.63	0	0	0	0	0	0	0	0	0	0	0	0.97
M9	0	0	0	0	0	0	0	0.54	0	0	0	0	0	0.52	0	0	0	0	0.85	0	0	0	0.55	0	0	0	0	0	0	0	0	0.99	0	0	0.93
M10	0	0	0	0	0	0	0	0.94	0	0	0	0	0	0.8	0	0.68	0	0	0.6	0	0	0.63	0	0	0	0.7	0	0	0	0	0	0	0	0	0
M11	0	0	0	0.9	0	0.71	0	0	0.98	0	0.53	0	0	0	0	0	0	0	0	0	0.68	0	0	0	0	0	0	0	0	0	0	0	0.91	0	0
M12	0	0	0	0.5	0	0.76	0	0	0.88	0	0.79	0	0	0	0	0	0	0	0	0	0.52	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M13	0	0.9	0	0	0	0	0	0	0	0	0	0.78	0.52	0	0	0	0	0	0	0	0	0	0	0.72	0	0	0	0	0	0	0	0	0	0	0
M14	0	0.9	0	0	0	0	0.92	0	0	0.86	0	0.8	0.67	0	0	0	0	0.53	0	0	0	0	0	0.69	0	0	0.59	0	0	0	0.54	0	0	0	0
M15	0	0	0	0.66	0	0.87	0	0	0.74	0	0.7	0	0	0	0	0	0	0	0	0	0.77	0	0	0	0	0	0	0.85	0	0.81	0	0	0	0	0
M16	0	0	0	0.63	0	0.6	0	0	0.96	0	0	0.53	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0.92	0	0	0	0.83	0	0.78	0	0	0
M17	0.9	0	0.5	0	0.82	0	0	0	0	0	0	0	0	0	0.89	0	0.65	0	0	0.57	0	0	0	0	0.97	0	0	0	0	0	0	0	0	0	0.88
M18	0	0.8	0	0	0	0	0.72	0	0	0	0	0.7	0.78	0	0	0	0	0	0	0.53	0	0	0	0.95	0	0	0	0	0	0	0.95	0	0	0	0
M19	0	0	0	0.55	0	0	0	0	0.84	0	0.67	0	0	0	0	0	0	0	0	0	0.92	0	0	0	0	0	0	0.85	0	0.52	0	0.61	0	0	0
M20	0	0	0	0	0	0	0	0.55	0	0	0	0	0	0.62	0	0	0	0	0.8	0	0	0	0.81	0	0	0.98	0	0	0	0	0	0	0	0	0

Data set - 18

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20
M1	0.53	0	0.99	0	0.83	0.91	0	0.82	0	0	0	0.83	0.91	0.92	0.86	0	0	0.97	0.79	0
M2	0	0	0.56	0	0	0.88	0	0	0.53	0.51	0	0.98	0	0	0	0	0	0	0	0
M3	0.83	0	0	0.71	0	0	0	0.58	0	0	0	0	0	0	0	0	0	0.54	0.54	0.74
M4	0	0	0	0	0.63	0	0	0	0	0	0	0	0.63	0	0	0	0	0	0	0
M5	0	0.53	0	0	0	0	0	0	0.69	0	0.63	0	0	0	0	0.68	0	0	0.51	0
M6	0	0	0	0	0	0	0	0	0	0.61	0	0	0	0	0	0	0.94	0	0	0
M7	0	0	0.68	0	0	0	0.67	0	0.7	0.84	0.79	0	0	0	0	0.99	0.94	0	0	0
M8	0	0	0	0	0	0	0.84	0	0	0.78	0	0	0	0	0	0	0	0	0	0
M9	0	0	0	0	0	0	0.93	0	0	0.73	0	0	0	0	0	0	0	0	0	0
M10	0.98	0	0	0	0.92	0.92	0	0	0.7	0	0	0.89	0.52	0	0	0	0	0.52	0	0
M11	0.54	0	0.77	0	0.76	0	0	0	0	0	0	0.96	0	0	0	0	0	0	0	0
M12	0	0	0	0.6	0	0	0	0.61	0	0	0	0	0	0.54	0.67	0	0	0	0.7	0.85
M13	0	0.99	0	0	0	0	0	0.87	0	0	0	0	0	0	0	0	0.67	0	0	0
M14	0	0.63	0	0	0	0	0	0	0	0	0	0	0	0	0	0.74	0	0	0	0
M15	0.85	0.78	0.55	0	0.81	0.63	0	0.97	0	0	0	0.54	0.52	0.85	0	0	0.55	0.99	0	0.93
M16	0	0	0	0	0	0	0	0.94	0	0.8	0	0	0	0.68	0	0	0	0	0	0.6
M17	0	0	0	0.63	0	0	0	0	0	0.7	0	0	0	0	0	0	0	0	0	0
M18	0	0	0.9	0	0	0	0	0	0.71	0	0	0.98	0.53	0	0.68	0.91	0.53	0	0	0
M19	0	0	0	0	0	0	0	0	0.76	0	0	0	0	0	0.88	0	0.79	0	0	0
M20	0	0	0	0	0	0	0	0	0	0	0	0.52	0.94	0	0	0.78	0	0	0	0
M21	0.52	0	0	0.72	0	0	0	0	0.92	0	0	0	0	0	0.92	0	0	0	0	0
M22	0.86	0	0.8	0.67	0	0.53	0	0.69	0.59	0.54	0	0	0	0.66	0.87	0	0	0	0	0.74
M23	0.7	0	0	0.77	0	0	0	0	0.85	0	0	0	0	0	0.81	0	0	0	0	0

Data set - 19

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16
M1	0.53	0	0.99	0	0	0	0.83	0.91	0	0	0	0	0	0.82	0	0
M2	0	0	0	0	0	0.83	0	0	0	0.91	0	0.92	0.86	0	0	0
M3	0.97	0	0	0	0	0.79	0	0.56	0	0	0	0.88	0	0	0	0
M4	0.53	0	0.51	0	0	0	0	0	0	0	0	0	0	0.98	0	0
M5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.83	0
M6	0.71	0	0.58	0	0	0	0	0	0	0	0	0	0	0.54	0.54	0
M7	0.74	0	0	0	0	0	0	0.63	0.63	0	0.53	0	0.69	0	0	0
M8	0.63	0	0	0	0	0	0	0	0	0	0	0	0.68	0	0	0
M9	0	0	0	0	0	0	0	0	0	0.51	0.61	0.94	0	0	0.68	0.67
M10	0.7	0.84	0	0	0	0	0	0.79	0	0.99	0	0	0	0	0	0
M11	0.94	0	0	0	0	0	0	0	0	0	0	0.84	0.78	0	0	0.93
M12	0	0	0	0	0	0	0	0	0	0	0	0	0	0.73	0	0
M13	0	0	0	0	0	0.98	0	0.92	0	0	0	0	0	0	0	0
M14	0	0	0	0	0	0	0	0	0	0.92	0.7	0.89	0	0.52	0	0.52
M15	0.54	0	0	0	0	0.77	0	0	0	0	0	0	0	0	0	0
M16	0	0	0.76	0.96	0	0	0	0	0	0	0	0	0	0	0	0
M17	0	0.6	0.61	0	0	0.54	0	0	0	0.67	0	0	0	0	0	0.7
M18	0.85	0	0	0	0	0.99	0	0	0	0	0	0	0	0	0	0
M19	0.87	0	0.67	0.63	0	0	0.74	0	0	0	0	0	0	0	0	0
M20	0	0	0	0	0	0	0	0	0	0.85	0.78	0.55	0	0	0.81	0.63
M21	0	0	0	0	0.97	0.54	0	0	0	0	0	0	0	0	0	0
M22	0.52	0.85	0	0	0	0	0	0.55	0	0.99	0	0	0	0	0	0
M23	0.93	0.94	0	0	0	0	0	0.8	0	0.68	0	0	0	0	0	0
M24	0	0	0	0	0	0	0.6	0	0.63	0.7	0.9	0.71	0	0.98	0	0.53

Data set - 20

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
M1	0.5	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0.8	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0
M2	0	0	0	0	0	0	0	0	0	0.8	0	0	0.9	0.9	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0	0	0	0	0	1	0.8	0	0	0	0
M3	0	0.6	0	0	0	0	0	0	0	0	0.9	0.5	0	0	0.5	0	0	0	0	0	0	0	1	0.8	0	0	0	0	0	0	0.7	0	0	0.6	0	0	0	0	0	0
M4	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0	0	0	0	0.5	0	0.7	0	0	0	0	0	0	0.6	0	0	0	0	0	0	0	0	0.6	0.5	0.7	0
M5	0	0	0	0	0	0	0	0	0	0.6	0	0	0.7	0.5	0	0	0	0	0	0	0	0.6	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0.7	0	0	0	0
M6	0	0	0	0.7	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0	0.8	1	0	0	0.9	0	0	0	0	0	0	0	0	0	0
M7	0	0	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0
M8	0	0	0	0.7	1	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0.9	0.7	0	0	0.9	0	0	0	0	0	0	0	0	0	0
M9	0	0	0	0	0	0.5	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0	0	0	0	0.8
M10	0	0	0	0	0	1	0.6	0	0	0	0	0	0	0	0	0	0	0	0	0.6	0	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0	0	0	0	0.7
M11	0	0	0	0	0	0	0	0	0	0.7	0	0	0.9	1	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0.7	0.6	0	0	0	0
M12	0	0	0	0.7	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0	0.6	0.8	0	0	0.6	0	0	0	0	0	0	0	0	0	0
M13	1	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0.5	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.6	0	0	0	0	0	0	0
M14	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0
M15	0	0	0	0.8	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0.6	0	0	0	0	0	0	0	0.6	0.7	0	0	0.9	0	0	0	0	0	0	0	0	0	0
M16	0	0	0	0	0	0	0	0.7	0	0	0	0	0	0	0	0	0	0	1	0	0.5	0	0	0	0	0	0	0.7	0	0	0	0	0	0	0	0	0.9	0.5	0.8	0
M17	0	0	0	0	0	0.9	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0	0	0	0.8
M18	0	0	0	0.5	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0.9	0.9	0	0	0.8	0	0	0	0	0	0	0	0	0	0
M19	0	0	0	0	0	0	0	0	0	0.7	0	0	0.5	0.7	0	0	0	0	0	0	0	0.6	0	0	0	0	0	0	0	0	0	0	0	0	0.5	0.7	0	0	0	0
M20	0	0.9	0	0	0	0	0	0	0	0	0.7	0.7	0	0	0.8	0	0	0	0	0	0	0	0.9	0.8	0	0	0	0	0	0	0.6	0	0	0.6	0	0	0	0	0	0
M21	1	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0.9	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0
M22	0.8	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0.5	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0
M23	0	0	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.6	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
M24	0	0	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0.7	0	0	0	0	0	0	0	0

Data set - 21

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
M1	0.5	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0.8	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0
M2	0	0	0	0	0	0	0	0	0	0.8	0	0	0.9	0.9	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
M3	0	0.8	0	0	0	0	0	0	0	0	0.6	0.9	0	0	0.5	0	0	0	0	0	0	0	0.5	1	0	0	0	0	0	0	0.8	0	0	0.7	0	0	0	0	0	0
M4	0	0	0	0	0	0	0	0.6	0	0	0	0	0	0	0	0	0	0	0.5	0	0.5	0	0	0	0	0	0	0.7	0	0	0	0	0	0	0	0	0.6	0.6	0
M5	0	0	0	0	0.5	0	0	0	0	0.7	0	0	0	0.6	0	0	0	0	0	0	0	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0.5	0.6	0	0	0
M6	0	0	0	0.9	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0.7	0	0	0	0	0	0	0	0.7	0.8	0	0	0.8	0	0	0	0	0	0	0	0	0	0
M7	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0	0
M8	0	0	0	0.8	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0.7	0	0	0	0	0	0	0	1	0.9	0	0	0.9	0	0	0	0	0	0	0	0	0	0
M9	0	0	0	0	0	0.7	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0	0	0	0	0.5
M10	0	0	0	0	0	0.5	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0.6
M11	0	0	0	0	0	0	0	0	0	0.6	0	0	0.5	0.7	0	0	0	0	0	0	0	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0.9	1	0	0	0	0
M12	0	0	0	0.9	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0.6	0	0.7	0	0	0	0.9	0	0	0.8	0	0	0.6	0	0	0	0	0	0	0	0	0	0
M13	0.8	0	0	0	0	0	0	0	0.6	0	0	0	0	0	0	1	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0
M14	0	0	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.6	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
M15	0	0	0	0.9	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0	0.7	0.6	0	0	0	0	0	0	0	0	0	0.6	0	0	0
M16	0	0	0	0	0	0	0	0.7	0	0	0	0	0	0	0	0	0	0	0.9	0	0.7	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0.5	0.7	0.9	0
M17	0	0	0	0	0	0.5	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0	0.8	0	0.5	0	0	0	0	0	0	0	0	0.9
M18	0	0	0	0.8	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0.7	0	0	0	0	0	0	0	0.9	0.9	0	0	0.9	0	0	0	0	0	0	0	0	0	0
M19	0	0	0	0	0	0	0	0	0	0.8	0	0	0.7	0.5	0	0	0	0	0	0	0	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0.6	0.5	0	0	0	0
M20	0	0.7	0	0	0	0	0	0	0	0	0.9	0.7	0	0	0.7	0	0	0	0	0	0	0	0.8	0.9	0	0	0	0	0	0	0.8	0	0	0.6	0	0	0	0	0	0
M21	0.6	0	0	0	0	0	0	0	1	0.5	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0
M22	0.8	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	1	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0
M23	0	0	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.7	0	0	0	0	0	0	0.6	0	0	0	0	0	0	0	0
M24	0	1	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0	0

Data set 22

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30	P31	P32	P33	P34	P35	P36	P37	P38	P39	P40	P41
M1	0	0	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0.8	0.9	0	0	0	0	0	0.8	0	0.8
M2	0	0	0	0	0	0	0	0	0	0	0.9	0.9	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0.8	0	0	0	0	0	0.6	0	0.9
M3	0	0	0	0	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0	1	0.8	0	0	0	0	0	0	0	0.7	0
M4	0	0	0	0	0	0	0.6	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0	0	0.5	0.7	0	0	0.6	0	0	0	0	0	0	0.6	0	0	0	0	0	0
M5	0	0	0	0	0.5	0	0	0	0	0	0	0	0	0	0	0	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.6	0	0.7	0	0	0	0	0
M6	0	0	0	0	0	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0	0	0	0.7	0	0	0	0	0	0	0.7	0	0.7	0	0	0	0	0
M8	0.8	0	0	0	0	0	0	0.8	1	0	0	0	0	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0	0	0	0	0	0
M9	0	0	0.8	0	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M10	0	1	0	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0.9	0	0	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0
M11	0	0	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0.5	0	0.8
M12	0	0	0	0	0	0	0	0	0	0	0.8	1	0	0	0	0	0	0.6	0.6	0	0	0	0.5	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.7	0.9	0
M13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0.7	0	0	0
M14	0	0	0	0.6	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0	0	0	0.6	0	0	0	0	0
M15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.6	0	0	0	0
M16	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0
M17	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0	0	0	0.6	0	0	0	0	0	0	0	0	1	0	0	0	0	0
M18	0	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0.9	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0.7	0	0	0	0	0	0	0
M19	0.6	0	0.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.7	0.9	0	0	0	0	0	0	0	0.7	0	0	0	0	0	0	0	0	0	0	0
M20	1	0	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.7	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M21	0	0	0	0	0	0	0	0	0	0.5	0	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0.8	0.5	0	0	0	0	0.9	0.8	0
M22	0	0	0	0	0	0	0	0	0	0.5	0	0.7	0	0	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0.9	0.9	0	0	0	0	0	0.8	0.7	0
M23	0	0.5	0	0	0	0	0	0	0	0	0	0.7	0	0	0	0	0	0	0	0	0	0	0.6	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0.7	0	0
M24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0	0
M25	0	0	0	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.7	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0.9	0	0.8	0	0	0	0	0
M27	0	0	0	0	0	0	0	0.6	0	0	0	0	0	0	0.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M28	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0.9	0	0	0	0	0	0	0
M29	0.9	0	0.8	0	0	0	0	0	0.8	0	0	0	1	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0	0	0	0	0
M30	0.9	0	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Data set 23

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30	P31	P32	P33	P34	P35	P36	P37	P38	P39	P40	P41	P42	P43	P44	P45	P46	P47	P48	P49	P50
M1	0.5	0	1	0.8	0	0.9	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M2	0	0.8	0	0.6	0	0.9	0	0	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M3	0.5	0	0	0	1	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M4	0.7	0	0	0	0.5	0	0	0.7	0.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M5	0	0	0.6	0	0	0	0.5	0	0	0	0.7	0.6	0.7	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M6	0	0.6	0	0	0	0	0.9	0	0	0.7	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M7	0	0	0.7	0	0.8	0	0	0	0	0.8	1	0.9	0.8	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M8	0	0.7	0	0.9	0.5	0	0.5	0.5	0.8	0	0	0	0	0	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M9	1	0.6	0	0	0.6	0	0	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M10	0	0	0	0	0.7	0.8	0	0	0	0	0	1	0.9	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M11	0.6	0	0.7	0.9	0	0	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0.5	0	0.8	0	0	0	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.7	0.6	0	0	1	0	0	0	0.9	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.7	0	0	0.7	0	0	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.5	1	0	0	1	0	0	0.5	0	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.7	0	0	0	0.9	0	0.9	0	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.6	0.6	0.6	0	0	0	0.8	0	0	0.8	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.7	0.8	0	1	0	0.6	0	0.6	0	0.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0.9	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.7	1	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	1	0.5	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.7	0.6	0.7	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.5	1	0.9	0.9	0.8	0	0	0	0	0	0	0	0	0	0	0
M26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0.7	0.9	0	0	0	0	0	0	0	0	0	0	0	0
M27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0.8	0.5	0.7	0.9	0.9	0.9	0	0	0	0
M28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0.9	0.8	0	0	0.8	0	0	0	0
M29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.7	0.7	0.77	0
M30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0.6	0.96	0.53

Data set 24

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30	P31	P32	P33	P34	P35	P36	P37	P38	P39	P40	P41	P42	P43	P44	P45	P46	P47	P48	P49	P50
M1	0	0	0	0	0	0	0	0.5	1	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0
M2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0.9	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	1	0	0	0	0
M3	0.8	0	0	0.6	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.5	0.5	0	0	0	0	1	0	0	0	0	0	0	0	0	0.83	0
M4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.7	0	0	0	0	0.6	0	0	0	0	0	0	0	0	0	0.5	0	0.5	0	0	0	0	0	0	0	0	0.7	0	0	0	0	0	0	0
M5	0	0.6	0.6	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M6	0	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.6	0	0	0	0	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0	0	0.6	0	0
M7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0	0.7	0.7	0.7	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0	0	0	0
M8	0.8	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0.8	0	0	0.8	0	0	0	0	0	0
M9	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0.7	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0.9	0	0	0	0	0	0	0	0.7	0
M10	0	0.9	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.5	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0.8	0	0	0	0	0	0	1	0.6	0	0	0
M11	0	0	0	0	0	0	0	0.6	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0.7	0	0.7	0	0	0	0	0	0	0	0	0	0.9	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
M12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0.7	0	0	0.6	0.7	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0
M13	0	0.6	0.8	0	0	0	0	0.6	1	0	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.5	0	0	0	0.9	0	0	0	0	0	0
M14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.6	1	0	0	0	0	0	0	0	0	0	0	0.9	0.9	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0.7	0	0
M15	0	0	0	0	0	0	0	0	0	0.6	0	0.6	0	0	0	0	0	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0.7	0	0	0	0.98
M16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0	0	0	0.7	0.9	0	0	0.5	0	0	0	0.8	0	0	0	0	0	0	0	0	0	0	0
M17	0	0	0	0	0.9	0	0.8	0	0	0	0	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0.8	0	0	0	0	0	0.5	0	0	0	0	0	0	0.7	0	0	0	0	0	0.92
M18	0	0	0	0	0.9	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M19	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.7	0	0	0	0	0.6	0	0.5	0	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0	0.7	0.7	0	0	0.77
M20	0	0	0	0	0.9	0.8	0.6	0	0	0	0	0.6	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0.5	0	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M21	0	0	0	0	0.9	0	0.8	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0.5	0	0	0	0	0	0.8	0	0	0	0	0
M22	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M23	0	0	0	0	0	0.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0.9	0	0	0.8	0	0	0	0	0.7	0	0	0	0	0	0	0.7	0	0	0	0
M24	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
M25	0.6	0	0	0.8	0	0	0	0	0	0	0	0	0	0	0	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0	0.9	0	0	0	0.5	0	0	0	0	0.6	0	0	0	0	0
M26	0	0	0	0	0	0	0	0	0	0.6	0	0	0.6	0	0	0	0.8	0	0	0.8	0	0	0	0	0	0	0	0	0	0	1	0	0	0.7	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M27	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.6	0	0	0	0	0	0	0	0	0.6	0	0	0	0	0	0	0
M28	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0.7	0	0	0	0.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0.8	0	0	0	0	0	0
M29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.7	0	0	0	0	0	0.9	0.6	0.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.5	0	0	0	0.8	0	0	0	0	0	0	0	0	0	0	0	0

Data set – 25

	P1	P2	P3	P4	P5	P6	P7	P8
M1	0.3	0	0	0.4	0	0	0.5	0
M2	0	0.3	0.1	0	0.5	0.2	0	0.6
M3	0	0.3	0.3	0	0.1	0.2	0	0.2
M4	0.6	0	0	0.7	0	0	0.8	0
M5	0.1	0	0	0.2	0	0	0.3	0

Data set – 26

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11
M1	0	0.3	0.25	0	0	0	0.3	0	0	0	0
M2	0.2	0	0	0	0.28	0	0	0	0	0	0.35
M3	0	0	0	0	0	0	0	0	0	0.14	0.7
M4	0.15	0	0.1	0	0	0.7	0	0	0	0	0
M5	0	0	0	0	0.68	0	0	0.2	0	0	0
M6	0.13	0	0	0.28	0	0	0	0.15	0.3	0.1	0
M7	0	0	0.3	0.12	0	0.1	0.4	0	0.08	0	0

Data set – 27

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20
M1	0.16	0	0.18	0.14	0	0	0	0	0.23	0.08	0	0	0	0.07	0.19	0.04	0	0.26	0.07	0
M2	0	0.3	0.09	0.1	0	0.05	0.02	0	0.12	0	0.07	0	0	0	0	0	0	0.16	0	0.03
M3	0	0	0	0	0.14	0.21	0.18	0.11	0	0	0.22	0.08	0.07	0	0	0.05	0.25	0	0.17	0.24
M4	0	0	0.03	0.05	0	0	0.06	0.01	0.09	0.12	0	0	0.03	0.16	0.24	0	0.09	0.28	0.06	0.04
M5	0.12	0.1	0.18	0	0	0.28	0	0	0	0.15	0	0.17	0.14	0	0.16	0.3	0.24	0	0.15	0.26
M6	0.11	0.07	0	0	0.32	0	0.26	0.22	0	0.1	0.21	0.19	0	0.08	0.14	0.17	0	0.13	0.13	0
M7	0	0	0	0	0.22	0.19	0.21	0.09	0	0	0.1	0.11	0.19	0	0	0.11	0.15	0	0.18	0.2
M8	0.06	0.1	0.11	0.21	0.02	0	0	0.04	0.1	0.05	0	0	0.07	0.29	0	0.03	0.08	0	0	0

Data set – 28

	P1	**P2**	**P3**	**P4**	**P5**	**P6**	**P7**	**P8**	**P9**	**P10**	**P11**	**P12**	**P13**	**P14**	**P15**	**P16**	**P17**	**P18**	**P19**	**P20**
M1	0	0.53	0.99	0	0	0	0	0.83	0.91	0	0.82	0	0.83	0.91	0	0.92	0.86	0	0.97	0
M2	0	0	0.79	0.56	0	0.88	0.53	0	0	0	0	0	0	0.51	0	0	0	0.98	0	0.83
M3	0	0.71	0	0	0	0	0	0.58	0.54	0	0.54	0	0.74	0.63	0	0.63	0.53	0	0.69	0
M4	0	0	0.63	0.68	0	0.51	0.61	0	0	0.94	0	0	0	0	0	0	0	0.68	0	0.67
M5	0.7	0	0	0	0.84	0.79	0	0	0	0.99	0	0.94	0	0	0.84	0	0.78	0	0	0
M6	0.93	0	0	0	0.73	0	0	0	0.98	0.92	0	0.92	0	0	0.7	0	0	0	0	0.89
M7	0	0	0.52	0.52	0	0.54	0.77	0	0	0	0.76	0.96	0	0	0	0	0	0.6	0	0.61
M8	0	0	0.54	0.67	0	0.7	0.85	0	0	0	0	0	0	0	0	0	0	0.99	0	0.87

Data set – 29

	P1	P2	P3	P4	P5	P6	P7	P8	P9
M1	0.3	0.1	0	0	0.3	0	0	0	0
M2	0.1	0.3	0	0	0	0.1	0	0	0.3
M3	0	0	0.3	0	0	0	0.3	0.3	0
M4	0	0.1	0.3	0.1	0	0	0	0.1	0
M5	0.3	0	0	0.1	0.3	0	0	0.1	0
M6	0	0.1	0	0	0	0.1	0	0	0.2
M7	0	0	0.3	0	0	0	0.3	0.1	0
M8	0	0	0.3	0.1	0.1	0	0.1	0.3	0
M9	0	0	0	0	0	0.2	0	0	0.3

Data set – 30

	P1	**P2**	**P3**	**P4**	**P5**	**P6**	**P7**	**P8**	**P9**	**P10**	**P11**	**P12**	**P13**	**P14**	**P15**
M1	0	0.53	0	0	0	0	0.99	0	0	0	0.83	0.91	0	0	0
M2	0	0	0	0	0.82	0	0	0	0.83	0.91	0	0	0.92	0	0.86
M3	0.97	0	0.79	0	0	0.56	0	0	0	0	0	0	0	0.88	0
M4	0.53	0	0.51	0.98	0	0.83	0	0	0	0	0	0	0	0.71	0
M5	0	0	0	0	0	0	0	0.58	0.54	0.54	0	0	0.74	0	0.63
M6	0	0.63	0	0	0	0.53	0.69	0	0	0	0.63	0.68	0	0	0
M7	0	0.51	0	0	0	0.61	0	0	0	0	0.94	0.68	0	0	0
M8	0.67	0	0.7	0.84	0	0.79	0	0	0	0	0.99	0	0	0.94	0
M9	0	0	0.84	0.78	0	0.93	0	0	0	0	0	0	0	0.73	0
M10	0	0	0	0	0	0	0	0.98	0.92	0.92	0	0	0.7	0	0.89

Data set – 31

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14
M1	0.7	0	0	0.8	0	0.7	0	0	0.5	0.4	0.5	0	0	0
M2	0	0	0	0	0.8	0	0	0	0	0	0	0.9	0.7	0.6
M3	0	0	0	0	0.6	0	0	0	0	0	0	0.7	0.8	0.7
M4	0	0.7	0.8	0	0	0	0.4	0.5	0	0	0	0	0	0
M5	0.6	0	0	0.7	0	0.4	0	0	0.7	0.8	0.9	0	0	0
M6	0	0	0	0	0.7	0	0	0	0	0	0	0.8	0.9	0.6
M7	0	0.8	0.5	0	0	0	0.8	0.9	0	0	0	0	0	0
M8	0	0.4	0.6	0	0	0	0.7	0.5	0	0	0	0	0	0

Data set – 32

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
M1	0	0.8	0	0	0	0.6	0	0	0.5	0
M2	0	0	0.4	0	0	0	0.5	0	0	0.6
M3	0	0.5	0	0	0	0.6	0	0	0.7	0
M4	0	0	0.9	0	0	0	0.6	0	0	0.4
M5	0.5	0	0	0.6	0.7	0	0	0.8	0	0
M6	0.9	0	0	0.7	0.6	0	0	0.8	0	0
M7	0	0	0.5	0	0	0	0.6	0	0	0.7
M8	0	0.7	0	0	0	0.9	0	0	0.7	0
M9	0.8	0	0	0.6	0.9	0	0	0.8	0	0

Data set 33

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30	P31
M1	0	0.53	0	0	0	0	0	0	0	0	0.99	0	0	0.83	0.91	0.82	0	0	0	0	0	0	0	0.83	0	0.91	0	0.92	0.86	0	0.97
M2	0	0	0.79	0.56	0	0	0	0.88	0	0	0	0.53	0	0.51	0	0	0.98	0.83	0.71	0	0	0	0	0.58	0	0	0.54	0.54	0.74	0.63	0
M3	0	0	0.63	0	0	0.53	0	0	0	0	0	0	0	0	0	0	0.69	0.63	0.68	0.51	0	0	0	0	0	0	0	0	0	0	0
M4	0.61	0.94	0.68	0	0	0	0	0	0	0	0	0	0	0	0	0	0.67	0	0	0	0	0	0	0.7	0	0	0	0.84	0.79	0.99	0
M5	0	0	0	0	0.94	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M6	0	0	0	0	0	0	0	0	0	0	0	0	0.84	0	0	0	0	0	0	0	0	0	0	0	0	0.78	0	0	0	0	0
M7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.93	0.73	0	0	0	0	0.98	0	0	0	0
M8	0	0	0	0	0	0	0.92	0.92	0.7	0.89	0.52	0.52	0	0	0	0	0	0	0	0	0	0	0.54	0.77	0	0	0.76	0.96	0.6	0.61	0.54
M9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.67	0	0	0	0	0	0
M10	0	0	0	0.7	0	0	0	0	0	0	0	0	0	0.85	0.99	0	0	0	0	0	0.87	0.67	0	0	0	0	0.63	0	0	0	0
M11	0.74	0	0	0	0	0	0.85	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.78	0	0	0	0.55	0.81	0	0
M12	0	0	0	0	0	0	0	0	0	0	0	0.63	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Data set - 34

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30
M1	0	0	0	0	0.53	0	0	0.99	0	0.83	0	0.91	0	0	0	0	0	0	0	0.82	0	0.83	0.91	0	0	0.92	0	0	0.86	0
M2	0	0.97	0	0	0.79	0	0	0	0.56	0	0	0	0	0	0	0	0.88	0	0	0.53	0.51	0.98	0	0	0.83	0.71	0.58	0	0	0
M3	0	0	0	0	0	0	0	0	0	0	0.54	0.54	0.74	0	0.63	0	0	0.63	0	0	0	0.53	0	0	0	0	0	0	0	0
M4	0	0	0.69	0	0	0	0.63	0	0	0	0	0	0.68	0.51	0	0	0	0	0	0	0	0.61	0.94	0	0	0	0	0	0	0.68
M5	0	0	0.67	0.7	0	0	0	0	0	0	0.84	0.79	0.99	0	0.94	0.84	0	0	0	0	0	0.78	0	0	0	0	0	0	0	0
M6	0	0	0	0	0	0	0	0	0	0.93	0	0	0	0	0	0	0	0	0	0	0.73	0	0.98	0	0	0.92	0.92	0.7	0.89	0.52
M7	0	0	0	0	0	0	0.52	0	0	0	0.54	0	0	0.77	0.76	0.96	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M8	0.6	0	0	0	0	0.61	0	0	0	0	0.54	0	0	0	0	0.67	0	0.7	0	0	0	0	0	0	0	0	0	0	0	0
M9	0	0.85	0	0	0	0	0	0	0.99	0.87	0	0.67	0	0	0	0	0	0	0	0	0	0.63	0	0.74	0.85	0	0	0.78	0	0.55
M10	0.81	0	0	0	0	0	0	0	0	0	0	0.63	0	0	0.97	0.54	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M11	0	0	0	0	0	0	0	0.52	0.85	0	0	0.55	0	0	0	0.99	0	0	0	0	0	0	0	0.93	0	0	0	0.94	0	0.8
M12	0.68	0	0.6	0.63	0	0	0.7	0	0	0	0.9	0	0.71	0	0.98	0.53	0	0.68	0	0	0.91	0	0	0	0	0	0	0	0	0
M13	0	0	0	0	0	0	0	0	0.53	0.76	0	0	0	0	0.88	0	0.79	0	0	0	0	0	0.52	0	0.94	0.78	0.52	0	0.72	0
M14	0	0	0	0	0	0	0	0.92	0	0	0	0	0	0	0	0	0	0	0	0.92	0	0	0.86	0	0.8	0	0	0	0	0
M15	0.67	0	0.53	0.69	0	0	0.59	0	0	0	0.54	0	0.66	0.87	0	0.74	0	0	0.7	0	0	0	0	0	0	0	0	0	0	0
M16	0	0	0	0	0	0	0	0.77	0	0	0	0	0	0	0	0	0.85	0	0	0.81	0	0	0.63	0.6	0.96	0.53	0.9	0.92	0.83	0.78

Data set - 35

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30
M1	0	0	0	0	0.53	0.99	0	0.83	0	0.91	0.82	0.83	0.91	0	0	0	0	0	0	0	0	0	0	0	0	0.92	0	0	0	0
M2	0	0	0	0.86	0.97	0	0	0	0.79	0.56	0.88	0.53	0.51	0	0.98	0.83	0	0.71	0	0	0	0	0	0	0	0.58	0	0	0	0
M3	0.54	0.54	0	0.74	0	0	0	0	0	0	0	0	0	0	0	0	0.63	0	0	0	0	0	0	0	0	0	0	0	0	0
M4	0	0	0.63	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.53	0	0	0.69	0	0.63	0.68	0.51	0	0.61	0.94	0	0
M5	0	0.68	0	0	0.67	0.7	0	0.84	0.79	0.99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.94	0	0.84	0	0
M6	0	0	0.78	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.93	0.73	0	0.98	0.92	0	0	0.92	0	0.7	0.89
M7	0	0	0	0	0	0	0.52	0	0	0	0	0	0	0	0	0	0	0	0	0.52	0	0	0.54	0.77	0.76	0	0.96	0.6	0.61	0
M8	0	0	0	0.54	0.67	0	0	0	0.7	0.85	0	0.99	0.87	0	0	0	0	0	0.67	0	0	0	0	0	0	0.63	0	0	0	0
M9	0	0	0	0	0	0	0	0	0	0	0.74	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M10	0.85	0.78	0	0	0	0	0	0	0	0	0	0	0	0.55	0	0	0.81	0	0	0	0	0	0	0	0	0	0	0	0	0
M11	0	0	0	0	0.63	0.97	0	0.54	0.52	0.85	0.55	0	0.99	0	0	0	0	0	0.93	0	0	0	0	0	0	0.94	0	0	0	0
M12	0	0	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.68	0	0.6	0.63	0	0.7	0	0	0	0	0.9	0.71	0.98
M13	0	0	0.53	0	0	0	0.68	0	0	0	0	0	0	0	0	0	0	0.91	0	0	0.53	0.76	0	0.88	0	0	0.79	0	0	0.52
M14	0	0	0	0	0.94	0	0.78	0	0	0.52	0.72	0.92	0	0	0	0.92	0	0	0.86	0	0	0	0	0	0	0	0	0	0	0
M15	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0.67	0.53	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M16	0.69	0.59	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.54	0	0	0	0	0	0	0	0	0	0	0	0	0

Data set – 36

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30
M1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.53	0	0	0	0	0	0	0	0	0.99	0	0.83	0	0
M2	0	0	0	0.91	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.82	0	0	0	0.83	0	0	0	0	0	0.91
M3	0.92	0	0	0	0	0.86	0	0	0	0	0	0	0.97	0	0.79	0	0.56	0	0.88	0	0.53	0.51	0	0	0	0	0	0	0	0
M4	0.98	0	0.83	0	0	0.71	0.58	0	0.54	0.54	0	0	0.74	0.63	0.63	0	0.53	0	0	0	0.69	0.63	0	0	0	0	0	0	0	0
M5	0	0.68	0	0	0.51	0	0	0	0.61	0	0	0	0	0	0	0.94	0	0.68	0	0	0	0	0	0.67	0	0	0.7	0	0	0
M6	0	0.84	0	0	0	0	0	0	0	0	0.79	0	0	0	0	0.99	0	0	0	0	0	0	0	0.94	0	0	0.84	0	0	0
M7	0.78	0	0.93	0	0	0	0.73	0	0.98	0.92	0	0	0.92	0	0.7	0.89	0.52	0	0	0	0.52	0.54	0	0	0	0	0.77	0	0	0
M8	0	0	0	0	0	0	0.76	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.96	0	0.6	0	0	0.61	0.54	0
M9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.67	0	0.7	0.85	0	0	0.99	0
M10	0	0	0	0	0	0	0	0.87	0	0	0	0.67	0	0	0	0	0	0	0	0.63	0	0	0	0	0	0	0	0	0	0.74
M11	0	0	0	0.85	0	0	0	0	0	0	0	0.78	0	0	0	0	0	0	0	0.55	0	0	0	0	0	0	0	0	0	0.81
M12	0	0	0	0.63	0	0	0	0.97	0	0	0	0	0	0	0	0	0	0	0	0.54	0	0	0	0	0	0	0	0	0	0
M13	0.52	0.85	0	0	0.55	0	0	0	0	0	0	0	0	0	0	0.99	0	0.93	0	0	0	0	0	0.94	0	0	0.8	0	0	0
M14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.68	0	0	0	0.6	0	0	0.63	0	0.7	0.9	0
M15	0	0	0	0	0	0	0	0	0	0	0.71	0	0	0	0	0.98	0	0.53	0	0	0	0	0	0.68	0	0	0.91	0	0	0
M16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.53	0	0	0.76	0	0.88	0.79	0

Data set - 37

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30
M1	0	0	0	0	0	0.53	0.99	0	0	0	0	0	0	0	0	0.83	0	0	0.91	0	0	0	0	0.82	0	0	0.83	0.91	0.92	0
M2	0	0.86	0	0	0	0	0	0	0.97	0	0	0	0.79	0.56	0	0	0	0.88	0	0	0.53	0.51	0	0	0	0.98	0.83	0	0	0
M3	0.71	0	0	0.58	0	0.54	0	0	0	0	0.54	0	0	0	0	0.74	0.63	0	0	0	0	0	0	0	0.63	0	0	0.53	0.69	0
M4	0	0.63	0	0	0	0	0	0	0.68	0	0	0	0	0	0	0	0	0.51	0	0	0	0	0	0	0	0	0	0	0	0
M5	0.61	0	0	0.94	0	0.68	0	0	0	0	0.67	0	0	0	0	0.7	0	0	0.84	0	0	0	0.79	0.99	0	0	0	0	0.94	0
M6	0.84	0	0	0.78	0	0	0	0	0	0.93	0.73	0.98	0	0	0	0	0	0	0.92	0	0	0	0.92	0	0	0	0	0.7	0.89	0
M7	0	0.52	0.52	0	0	0.54	0	0	0	0.77	0	0	0	0.76	0	0.96	0.6	0	0	0	0	0	0	0	0	0	0.61	0	0.54	0.67
M8	0	0	0	0	0	0	0	0	0	0	0	0	0.7	0	0	0	0	0.85	0	0	0	0	0	0	0	0	0	0	0	0
M9	0.99	0	0	0	0	0.87	0	0	0	0.67	0	0	0	0	0	0	0	0	0	0	0	0	0.63	0	0	0	0	0.74	0	0.85
M10	0	0	0	0	0	0	0.78	0	0.55	0	0	0	0	0	0	0	0	0	0	0	0.81	0.63	0	0	0	0.97	0	0	0.54	0
M11	0.52	0.85	0	0	0.55	0	0	0.99	0	0	0	0	0.93	0	0	0	0	0.94	0.8	0	0	0	0	0	0.68	0	0	0	0	0
M12	0	0	0.6	0.63	0	0	0	0	0	0	0	0.7	0	0	0	0.9	0.71	0	0	0	0	0	0.98	0.53	0	0	0	0.68	0	0.91
M13	0	0	0	0	0	0	0	0.53	0	0	0	0	0	0.76	0	0	0	0	0	0	0	0	0	0	0	0.88	0	0	0	0
M14	0	0	0	0	0	0	0	0	0	0	0	0.79	0	0	0.52	0	0	0.94	0	0	0.78	0	0	0	0.52	0	0	0	0	0.72
M15	0	0	0	0	0.92	0	0	0.92	0	0	0	0.86	0	0	0	0.8	0	0.67	0	0.53	0	0.69	0	0	0.59	0.54	0	0	0	0
M16	0	0	0	0	0.66	0.87	0	0	0	0	0	0	0	0.74	0	0	0	0.7	0	0	0	0	0	0	0	0.77	0	0	0	0

Data set - 38

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30
M1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.53	0.99	0	0	0	0	0	0.83	0.91	0	0	0.82	0	0
M2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.83	0	0.91	0	0	0	0	0	0	0.92	0.86	0	0.97	0	0
M3	0	0	0.79	0	0.56	0	0.88	0	0.53	0	0.51	0	0	0.98	0	0	0	0	0.83	0	0	0	0	0	0	0	0	0	0	0
M4	0	0	0	0	0.71	0	0.58	0	0	0	0.54	0	0	0.54	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M5	0.74	0	0	0.63	0	0	0	0.63	0	0.53	0	0	0	0	0.69	0	0.63	0	0	0	0	0	0	0	0.68	0	0	0	0	0.51
M6	0	0.61	0	0	0	0	0	0	0	0.94	0	0.68	0.67	0	0	0	0	0.7	0.84	0	0.79	0.99	0.94	0	0	0	0	0	0	0
M7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.84	0.78	0	0	0	0	0	0.93	0.73	0.98	0	0.92	0	0
M8	0	0	0.92	0	0.7	0	0	0	0	0	0.89	0	0	0.52	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.52	0
M9	0	0	0.54	0	0.77	0	0.76	0	0.96	0	0.6	0	0	0.61	0	0	0	0	0	0	0	0	0	0	0.54	0	0	0	0	0
M10	0.67	0	0	0.7	0	0	0	0.85	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.99
M11	0	0	0	0.87	0	0	0	0.67	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.63
M12	0.74	0	0.85	0	0.78	0	0.55	0	0.81	0	0.63	0	0	0.97	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M13	0	0.54	0	0	0	0.52	0	0	0	0.85	0	0.55	0.99	0	0	0	0	0	0.93	0.94	0.8	0	0.68	0	0	0	0.6	0	0	0.63
M14	0	0.7	0	0	0	0	0.9	0	0	0.71	0	0.98	0	0	0.53	0	0	0	0.68	0.91	0.53	0.76	0	0	0	0	0	0	0.88	0
M15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.79	0.52	0	0	0	0	0	0	0.94	0.78	0.52	0	0.72	0	0
M16	0	0	0	0	0	0.92	0	0	0	0.92	0	0.86	0.8	0	0	0	0	0	0	0.67	0.53	0.69	0.59	0	0	0	0.54	0	0.66	0

Data set - 39

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30
M1	0	0	0	0.53	0.99	0	0	0	0	0	0.83	0	0.91	0	0	0	0.82	0.83	0	0	0	0.91	0	0	0	0	0	0	0	0.92
M2	0	0	0	0	0.86	0.97	0	0.79	0.56	0	0	0	0	0	0.88	0	0	0	0	0.53	0.51	0	0	0	0	0	0	0	0	0
M3	0	0	0	0	0	0	0	0	0	0.98	0	0	0	0	0	0	0	0	0	0	0	0	0.83	0	0	0	0.71	0.58	0	0
M4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.54	0.54	0	0.74	0.63	0	0	0
M5	0.63	0	0.53	0	0.69	0.63	0	0.68	0.51	0.61	0	0	0	0	0.94	0	0	0	0.68	0	0.67	0	0	0	0	0	0	0	0	0
M6	0	0.7	0	0	0	0	0	0	0	0	0	0	0	0.84	0	0.79	0	0	0	0	0	0	0	0	0.99	0	0	0.94	0.84	0
M7	0	0	0	0	0.78	0.93	0	0.73	0.98	0	0	0	0	0	0.92	0	0	0	0.92	0	0.7	0	0	0	0	0	0	0	0	0
M8	0.89	0	0.52	0	0.52	0.54	0	0.77	0.76	0	0	0	0	0	0	0	0	0	0.96	0.6	0.61	0	0	0	0	0	0	0	0	0
M9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.54	0.67	0	0.7	0.85	0	0	0
M10	0.99	0	0.87	0	0.67	0	0	0.63	0	0	0	0	0	0	0.74	0	0	0	0	0	0.85	0	0	0	0	0	0	0	0	0
M11	0	0	0	0.78	0	0	0.55	0	0	0	0.81	0.63	0.97	0	0	0	0.54	0.52	0	0	0	0.85	0	0	0	0	0	0	0	0.55
M12	0	0	0	0	0	0	0.99	0	0	0	0	0	0	0.93	0	0.94	0	0	0	0	0	0	0	0	0.8	0	0	0.68	0.6	0
M13	0	0.63	0	0	0	0	0	0	0	0	0.7	0	0	0.9	0	0.71	0	0	0	0	0	0	0	0	0.98	0	0	0.53	0.68	0
M14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.91	0.53	0	0	0	0	0	0
M15	0	0.76	0	0	0.88	0	0	0	0	0	0	0	0	0	0	0.79	0	0	0	0	0	0	0	0	0.52	0	0	0.94	0.78	0
M16	0	0.52	0	0	0	0	0	0	0	0	0	0	0	0.72	0	0.92	0	0	0	0	0	0	0	0	0.92	0	0	0.86	0.8	0

Data set - 40

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30
M1	0.53	0	0	0	0	0.99	0.83	0.91	0	0	0	0	0.82	0.83	0.91	0.92	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M2	0	0	0.86	0	0.97	0	0	0	0	0.79	0	0	0	0	0	0.56	0	0	0	0	0	0	0.88	0.53	0	0	0.51	0	0.98	0.83
M3	0.71	0	0	0	0	0	0	0.58	0.54	0	0	0	0.54	0	0.74	0	0.63	0	0	0	0	0	0	0	0	0	0	0	0	0
M4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.63	0.53	0	0	0	0	0.69	0	0.63	0	0
M5	0.68	0.51	0	0	0	0.61	0.94	0.68	0.67	0	0	0	0	0.7	0	0.84	0.79	0	0	0	0	0	0	0	0	0	0	0	0	0
M6	0.99	0.94	0	0	0	0	0.84	0	0.78	0	0	0	0.93	0.73	0	0.98	0.92	0.92	0	0.7	0	0	0	0	0	0	0	0	0	0
M7	0	0	0	0.89	0	0	0	0	0	0	0	0.52	0	0	0	0	0	0.52	0.54	0	0	0.77	0	0	0	0	0	0	0	0
M8	0.76	0.96	0	0	0	0.6	0.61	0.54	0.67	0	0	0	0	0.7	0.85	0.99	0.87	0	0	0	0	0	0	0	0	0	0	0	0	0
M9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.67	0.63	0	0	0	0.74	0	0	0.85	0	0
M10	0.78	0	0	0	0	0.55	0.81	0	0.63	0	0	0	0.97	0.54	0.52	0.85	0.55	0	0	0	0	0	0	0	0	0	0	0	0	0
M11	0	0	0	0.99	0.93	0	0	0	0	0	0	0.94	0	0	0	0	0	0.8	0.68	0	0	0.6	0	0	0	0	0	0	0	0
M12	0	0	0	0	0	0	0	0	0	0	0	0.63	0	0	0	0	0	0.7	0.9	0	0	0	0	0	0	0	0	0	0	0
M13	0.71	0	0	0	0	0.98	0.53	0.68	0.91	0	0	0	0.53	0.76	0	0.88	0.79	0	0	0	0	0	0	0	0	0	0	0	0	0
M14	0	0	0.52	0	0	0	0	0	0	0	0	0	0	0.94	0	0	0	0	0	0	0	0	0.78	0.52	0.72	0	0.92	0	0.92	0.86
M15	0	0	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.67	0.53	0	0	0.69	0.59	0	0	0	0
M16	0	0	0	0	0.54	0	0	0	0	0.66	0	0	0	0	0	0	0	0	0	0	0	0	0.87	0.74	0	0.7	0	0	0.77	0.85

Data set - 41

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30
M1	0	0.53	0	0	0	0	0	0.99	0	0	0	0	0.83	0.91	0.82	0.83	0.91	0	0.92	0.86	0	0	0	0	0	0	0	0.97	0.79	0.56
M2	0	0	0	0	0	0.88	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.53	0	0	0.51	0.98	0.83	0.71	0	0	0
M3	0	0.58	0	0	0	0	0	0.54	0	0	0	0	0	0	0.54	0	0.74	0.63	0.63	0	0	0	0	0	0	0	0	0.53	0.69	0.63
M4	0.68	0	0	0	0	0	0	0	0.51	0	0.61	0	0.94	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M5	0	0	0	0	0	0.68	0	0	0	0	0	0.67	0	0	0	0	0	0	0	0	0.7	0.84	0.79	0	0.99	0	0.94	0	0.84	0
M6	0.78	0	0	0	0.93	0	0.73	0.98	0	0	0.92	0	0.92	0	0	0	0	0	0.7	0	0	0	0	0	0.89	0	0	0	0	0
M7	0	0	0	0	0	0	0	0	0	0.52	0	0.52	0	0	0	0	0	0	0	0	0.54	0.77	0	0.76	0	0.96	0.6	0	0	0
M8	0.61	0	0	0.54	0	0	0.67	0	0	0	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M9	0.85	0	0	0	0	0	0	0.99	0	0	0.87	0	0	0.67	0	0	0	0	0	0.63	0	0	0.74	0	0	0	0	0	0.85	0.78
M10	0	0	0	0.55	0.81	0	0	0	0.63	0	0.97	0	0.54	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M11	0	0	0	0	0	0	0	0	0	0	0	0	0	0.52	0.85	0.55	0	0.99	0.93	0.94	0	0	0	0	0	0	0	0.8	0	0
M12	0	0	0.68	0	0	0.6	0	0	0	0.63	0	0.7	0	0	0	0	0.9	0	0	0	0	0.71	0	0.98	0.53	0.68	0	0	0	0
M13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.91	0	0.53	0.76	0	0	0	0	0	0	0	0	0	0.88	0
M14	0	0	0.79	0	0	0.52	0	0	0	0.94	0	0.78	0	0	0	0	0	0	0	0	0	0.52	0.72	0	0.92	0	0.92	0	0	0
M15	0	0	0	0	0	0	0	0	0	0.86	0	0	0	0.8	0	0	0.67	0	0.53	0	0	0	0	0	0	0	0	0	0.69	0
M16	0	0	0	0	0.59	0	0	0.54	0	0	0	0	0	0.66	0.87	0.74	0.7	0.77	0	0.85	0	0	0	0	0	0	0	0.81	0	0.63

Data set - 42

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30
M1	0	0	0	0	0	0	0	0	0.53	0	0	0	0	0	0	0	0	0	0.99	0	0	0	0	0	0	0.83	0	0.91	0.82	0
M2	0	0.83	0.91	0.92	0.86	0	0.97	0.79	0	0	0	0.56	0.88	0	0.53	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M3	0	0	0.51	0.98	0	0	0	0	0	0.83	0.71	0	0	0	0.58	0.54	0.54	0	0	0	0	0	0	0.74	0	0	0	0	0	0.63
M4	0	0	0	0	0.63	0.53	0.69	0.63	0	0	0	0.68	0.51	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M5	0	0	0	0	0	0	0	0	0	0	0	0	0	0.61	0	0	0	0.94	0	0	0.68	0.67	0.7	0	0.84	0	0	0	0	0
M6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.79	0	0	0.99	0.94	0	0	0.84	0	0	0	0	0
M7	0.78	0	0.93	0.73	0	0	0	0	0.98	0.92	0.92	0	0	0	0.7	0.89	0.52	0	0	0	0	0	0	0.52	0	0	0	0	0	0.54
M8	0	0	0	0.77	0	0	0	0	0	0	0	0	0	0.76	0	0	0	0.96	0	0	0.6	0	0	0	0.61	0	0	0	0.54	0
M9	0	0	0	0	0	0	0	0	0.67	0	0	0	0	0	0	0	0	0	0	0.7	0	0	0	0	0	0.85	0.99	0.87	0.67	0
M10	0.63	0	0	0	0	0	0	0	0	0.74	0.85	0	0	0.78	0	0	0	0.55	0	0	0.81	0.63	0.97	0	0.54	0	0	0	0	0
M11	0.52	0	0.85	0.55	0	0	0	0	0	0.99	0.93	0	0	0	0	0.94	0.8	0	0	0	0	0	0	0.68	0	0	0	0	0.6	0.63
M12	0	0	0.7	0.9	0	0	0	0	0	0.71	0	0	0	0	0	0.98	0.53	0	0	0	0	0	0	0.68	0	0	0	0	0.91	0
M13	0	0	0	0	0	0	0	0	0	0	0	0	0	0.53	0	0	0	0.76	0	0	0.88	0	0.79	0	0.52	0	0	0	0	0
M14	0.94	0	0.78	0	0	0	0	0	0	0	0.52	0	0	0	0.72	0.92	0.92	0	0.86	0	0	0	0	0.8	0	0	0	0	0	0.67
M15	0.53	0	0.69	0.59	0	0	0	0	0	0.54	0	0	0	0	0.66	0.87	0.74	0	0	0	0	0	0	0.7	0	0	0	0	0	0.77
M16	0	0	0	0	0	0	0	0	0.85	0	0	0	0	0	0	0	0	0	0.81	0.63	0	0	0	0	0	0.6	0.96	0.53	0.9	0

Data set - 43

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30
M1	0.53	0	0.99	0.83	0	0	0.91	0	0.82	0	0.83	0	0	0	0.91	0.92	0	0	0	0	0	0	0	0	0	0	0.86	0	0	0
M2	0	0	0	0	0	0	0	0	0	0.97	0	0	0	0.79	0	0	0	0	0	0	0	0	0.56	0.88	0	0	0	0.53	0.51	0
M3	0	0	0.98	0	0	0	0	0	0	0	0	0	0	0.83	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.71
M4	0	0	0	0	0	0	0	0	0	0	0.58	0	0.54	0	0	0	0	0.54	0.74	0	0	0.63	0	0	0.63	0	0.53	0	0	0
M5	0	0	0	0	0	0	0	0.69	0	0.63	0	0	0	0	0	0	0	0	0	0	0	0	0.68	0.51	0	0	0	0	0.61	0.94
M6	0	0	0	0.68	0.67	0.7	0	0	0.84	0	0.79	0	0	0	0.99	0.94	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M7	0	0	0	0	0	0.84	0	0.78	0	0.93	0	0	0	0.73	0	0	0	0	0	0	0	0	0	0	0	0	0	0.98	0	0.92
M8	0.92	0.7	0.89	0.52	0.52	0.54	0.77	0	0.76	0	0.96	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.6	0	0	0	0
M9	0.61	0.54	0.67	0.7	0	0.85	0.99	0	0	0	0.87	0	0	0	0.67	0.63	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M10	0.74	0.85	0.78	0	0.55	0.81	0	0	0.63	0	0.97	0	0	0	0.54	0.52	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M11	0	0	0	0	0	0	0	0.85	0	0.55	0	0	0	0	0	0	0	0	0	0	0	0	0.99	0	0	0	0	0.93	0.94	0.8
M12	0.68	0	0	0	0	0	0.6	0	0	0	0	0	0	0	0.63	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M13	0	0	0.9	0	0	0	0	0	0	0.71	0	0	0	0	0	0	0	0	0.98	0	0	0	0.53	0.68	0	0	0	0	0.91	0.53
M14	0	0	0	0	0	0	0	0	0	0	0	0.76	0.88	0	0	0	0.79	0.52	0.94	0.78	0	0.52	0	0	0.72	0	0.92	0	0	0
M15	0	0	0	0	0	0	0	0.92	0.86	0	0	0	0	0.8	0	0	0	0	0	0	0	0	0.67	0	0	0	0	0.53	0	0
M16	0	0	0	0	0	0	0	0	0	0	0	0.69	0	0	0	0	0	0	0	0.59	0	0	0	0	0	0.54	0	0	0	0

Data set 44

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30	P31	P32	P33	P34	P35	P36	P37	P38	P39	P40	P41	P42	P43
M1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.53	0	0	0	0	0.99	0
M2	0	0.83	0	0	0	0	0	0	0	0.91	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.82	0	0	0	0.83	0	0	0	0	0.91	0.92	0	0.86	0	0.97	0
M3	0	0	0	0	0	0	0.79	0	0	0	0	0	0	0	0	0	0.56	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.88	0.53	0.51	0	0	0	0	0	0	0
M4	0	0	0	0	0.98	0	0	0	0.83	0	0	0	0	0.71	0	0	0	0	0.58	0	0.54	0	0.54	0	0	0	0	0	0.74	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M5	0	0	0	0	0.63	0	0	0.63	0.53	0	0	0	0	0.69	0.63	0.68	0	0	0.51	0	0.61	0	0.94	0	0	0	0	0	0.68	0	0	0	0.67	0	0	0	0	0	0	0	0.7	0	0.84
M6	0.79	0.99	0	0	0	0.94	0.84	0.78	0	0	0	0.93	0.73	0.98	0	0	0.92	0	0.92	0	0	0	0.7	0	0	0	0	0	0	0	0	0.89	0.52	0.52	0	0	0.54	0	0.77	0.76	0	0.96	0.6
M7	0.61	0	0	0	0	0	0	0	0	0	0	0	0.54	0	0	0	0	0	0	0	0	0	0	0	0.67	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M8	0.7	0.85	0.99	0	0	0	0	0.87	0.67	0	0.63	0.74	0	0	0.85	0	0	0	0.78	0.55	0.81	0	0.63	0.97	0	0	0.54	0.52	0	0	0.85	0	0	0	0	0	0.55	0.99	0	0	0.93	0	0.94
M9	0	0.8	0	0.68	0	0	0	0	0	0.6	0	0	0	0	0	0	0	0.63	0	0	0	0	0	0	0	0	0	0.7	0	0	0	0.9	0	0	0	0	0.71	0.98	0	0.53	0	0.68	0
M10	0.91	0	0	0	0	0	0	0	0	0	0	0.53	0.76	0	0	0	0	0	0	0	0	0	0	0	0.88	0.79	0	0	0	0	0.52	0	0	0	0	0	0	0	0.94	0	0	0	0
M11	0	0	0.78	0	0	0	0	0	0.52	0	0	0	0	0	0	0	0	0	0	0.72	0	0	0	0.92	0	0	0.92	0	0	0.86	0	0	0	0	0	0	0	0	0	0	0	0	0
M12	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0	0	0	0	0.67	0	0.53	0	0	0.69	0	0	0.59	0	0	0	0	0	0	0	0	0	0	0	0	0
M13	0	0	0.54	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.66	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M14	0	0.87	0	0	0	0.74	0	0	0	0	0	0	0	0	0	0	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.77	0	0	0	0	0	0	0	0
M15	0	0	0	0	0.85	0	0	0	0	0	0	0	0	0.81	0	0	0	0	0.63	0	0.6	0	0	0	0	0	0	0	0	0	0	0	0.96	0	0	0	0	0	0	0	0.53	0	0.9
M16	0	0.92	0	0	0	0	0.83	0	0	0.78	0	0	0	0	0	0	0	0.99	0	0	0	0	0	0	0	0	0	0	0	0	0	0.51	0	0	0	0	0.82	0.89	0	0	0	0.65	0

Data set – 45

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20
M1	0.4	0	0	0.5	0	0	0.6	0	0	0	0	0	0	0	0	0	0	0	0	0
M2	0	0.4	0.2	0	0.6	0	0	0.7	0	0.3	0	0	0	0	0	0	0	0	0	0
M3	0	0.4	0.4	0	0.2	0	0	0.3	0	0.3	0	0	0	0	0	0	0	0	0	0
M4	0.7	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0
M5	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0.9	0.5	0	0.6	0.7	0	0.8
M6	0.2	0	0	0.3	0	0	0.4	0	0	0	0	0	0	0	0	0	0	0	0	0
M7	0	0	0	0	0	0.3	0	0	0.4	0	0.5	0.6	0	0	0	0.7	0	0	0.8	0
M8	0	0	0	0	0	0.3	0	0	0.4	0	0.2	0.3	0	0	0	0.4	0	0	0.3	0
M9	0	0	0	0	0	0	0.9	0	0	0	0	0	0.4	0.5	0.6	0	0.7	0.8	0	0.2
M10	0	0	0	0		0.4	0	0	0.7	0	0.8	0.2	0	0	0	0.4	0	0	0.3	0

Data set – 46

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16
M1	0.7	0	0	0.8	0.4	0	0	0.2	0.1	0.2	0	0	0	0	0	0
M2	0	0	0.4	0	0	0	0	0	0	0	0	0.2	0	0	0.1	0
M3	0	0	0	0	0	0	0	0	0	0	0	0	0.1	0	0	0.2
M4	0.3	0	0	0.4	0.1	0	0	0.4	0.5	0.6	0	0	0	0	0	0
M5	0	0	0.1	0	0	0	0	0	0	0	0	0.2	0	0	0.3	0
M6	0	0	0	0	0	0.5	0	0	0	0	0	0	0.6	0	0	0.7
M7	0	0.1	0	0	0	0	0.2	0	0	0	0.3	0	0	0.4	0	0
M8	0	0.5	0	0	0	0	0.6	0	0	0	0.7	0	0	0.8	0	0
M9	0	0	0	0	0	0.1	0	0	0	0	0	0	0.2	0	0	0.3
M10	0	0	0.4	0	0	0	0	0	0	0	0	0.5	0	0	0.6	0
M11	0	0.4	0	0	0	0	0.2	0	0	0	0.5	0	0	0.7	0	0

Data set 47

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30	P31	P32	P33	P34	P35	P36	P37	P38	P39	P40	P41	P42	P43
M1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.53	0	0	0	0	0.99	0
M2	0	0.83	0	0	0	0	0	0	0	0.91	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.82	0	0	0	0.83	0	0	0	0	0.91	0.92	0	0.86	0	0.97	0
M3	0	0	0	0	0	0	0.79	0	0	0	0	0	0	0	0	0	0.56	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.88	0.53	0.51	0	0	0	0	0	0	0
M4	0	0	0	0	0.98	0	0	0	0.83	0	0	0	0	0.71	0	0	0	0	0.58	0	0.54	0	0.54	0	0	0	0	0	0	0.74	0	0	0	0	0	0	0	0	0	0	0	0
M5	0	0	0	0	0.63	0	0	0.63	0.53	0	0	0	0	0.69	0.63	0.68	0	0	0.51	0	0.61	0	0.94	0	0	0	0	0	0.68	0	0	0	0.67	0	0	0	0	0	0	0	0.7	0	0.84
M6	0.79	0.99	0	0	0	0.94	0.84	0.78	0	0	0	0.93	0.73	0.98	0	0	0.92	0	0.92	0	0	0	0.7	0	0	0	0	0	0	0	0	0.89	0.52	0.52	0	0	0.54	0	0.77	0.76	0	0.96	0.6
M7	0.61	0	0	0	0	0	0	0	0	0	0	0	0.54	0	0	0	0	0	0	0	0	0	0	0	0.67	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M8	0.7	0.85	0.99	0	0	0	0	0.87	0.67	0	0.63	0.74	0	0	0.85	0	0	0	0.78	0.55	0.81	0	0.63	0.97	0	0	0.54	0.52	0	0	0.85	0	0	0	0	0	0.55	0.99	0	0	0.93	0	0.94
M9	0	0	0	0.8	0	0	0	0	0	0.68	0	0	0	0	0	0	0	0.6	0	0	0	0	0	0	0	0	0	0.63	0	0	0	0.7	0	0	0	0	0.9	0.71	0	0.98	0	0.53	0
M10	0.68	0	0	0	0	0	0	0	0	0	0	0.91	0.53	0	0	0	0	0	0	0	0	0	0	0	0.76	0.88	0	0	0	0	0.79	0	0	0	0	0	0	0	0.52	0	0	0	0
M11	0	0	0.94	0	0	0	0	0	0.78	0	0	0	0	0	0	0	0	0	0	0.52	0	0	0	0.72	0	0	0.92	0	0	0.92	0	0	0	0	0	0	0	0	0	0	0	0	0
M12	0	0	0	0	0	0	0	0	0	0	0.86	0	0	0	0	0	0	0	0	0	0	0.8	0	0.67	0	0	0.53	0	0	0.69	0	0	0	0	0	0	0	0	0	0	0	0	0
M13	0	0	0.59	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.54	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M14	0	0.66	0	0	0	0.87	0	0	0	0	0	0	0	0	0	0	0.74	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.7	0	0	0	0	0	0	0	0
M15	0	0	0	0	0	0.77	0	0	0	0	0	0	0	0.85	0	0	0	0	0.81	0	0.63	0	0	0	0	0	0	0	0	0	0	0	0.6	0	0	0	0	0	0	0	0.96	0	0.53
M16	0	0.9	0	0	0	0	0.92	0	0	0.83	0	0	0	0	0	0	0	0.78	0	0	0	0	0	0	0	0	0	0	0	0	0	0.99	0	0	0	0	0.51	0.82	0	0	0	0.89	0

Data set 48

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30	P31	P32	P33	P34	P35	P36	P37	P38	P39	P40
M1	0	0	0	0	0	0	0	0	0.53	0	0	0	0	0	0	0.99	0.83	0	0.91	0	0	0	0	0	0	0	0	0	0	0	0	0.82	0	0	0	0	0	0	0
M2	0	0	0	0	0	0	0	0	0	0.83	0	0	0.91	0.92	0	0	0	0	0	0	0	0.86	0	0	0	0	0	0	0	0	0	0	0.97	0	0	0.79	0	0	0	0
M3	0	0.56	0	0	0	0	0	0	0	0	0.88	0.53	0	0	0.51	0	0	0	0	0	0	0	0.98	0.83	0	0	0	0	0	0	0	0	0	0.71	0	0	0	0	0	0
M4	0	0	0	0	0	0	0	0.58	0	0	0	0	0	0	0	0	0	0	0.54	0	0.54	0	0	0	0	0	0	0.74	0	0	0	0	0	0	0	0	0	0.63	0.63	0
M5	0	0	0	0	0.53	0	0	0	0	0.69	0	0	0	0.63	0	0	0	0	0	0	0	0.68	0	0	0	0	0	0	0	0	0	0	0	0	0.51	0.61	0	0	0	0
M6	0	0	0	0	0.94	0	0	0	0	0	0	0	0	0	0	0	0	0.68	0	0	0	0	0	0	0	0.67	0.7	0	0	0.84	0	0	0	0	0	0	0	0.79	0	0
M7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.99	0	0	0	0	0	0	0.94	0	0	0	0	0	0	0	0.84
M8	0	0	0	0.78	0.93	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.73	0.98	0	0	0.92	0	0	0	0	0	0	0	0	0	0
M9	0	0	0	0	0	0.92	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.89	0	0	0	0.52	0	0	0	0	0	0	0.52
M10	0	0	0	0	0	0.54	0.77	0	0	0	0	0	0	0	0	0	0	0	0	0.76	0	0	0	0	0	0.96	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M11	0	0	0	0	0	0	0	0	0	0.6	0	0	0.61	0	0	0	0	0	0	0	0	0.54	0	0	0	0	0	0	0	0	0	0	0	0	0.67	0.7	0	0	0	0
M12	0	0	0	0.85	0.99	0	0	0	0	0	0	0	0	0	0	0	0	0.87	0	0.67	0	0	0	0.63	0	0	0.74	0	0	0.85	0	0	0	0	0	0	0	0	0	0
M13	0.78	0	0	0	0	0	0	0	0.55	0	0	0	0	0	0	0.81	0.63	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.97	0	0	0	0	0	0	0
M14	0	0	0.54	0	0	0	0	0	0	0	0	0	0	0.52	0	0.85	0	0	0	0	0	0	0	0	0.55	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M15	0.99	0	0	0.93	0.94	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0	0.68	0.6	0	0	0.63	0	0	0	0	0	0	0.7	0	0	0
M16	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0	0	0	0.71	0	0.98	0	0	0	0	0	0	0	0	0	0	0.53	0	0	0	0	0.68	0.91	0.53	0
M17	0	0	0	0	0	0	0.76	0	0	0	0	0	0	0	0	0	0	0	0	0.88	0	0	0	0	0	0	0	0	0.79	0	0.52	0	0	0	0	0	0	0	0	0.94
M18	0	0	0	0.78	0.52	0	0	0	0	0	0	0	0	0	0	0	0	0.72	0	0	0	0	0	0	0	0.92	0.92	0	0	0.86	0	0	0	0	0	0	0	0	0	0
M19	0	0	0	0	0	0	0	0	0	0.8	0	0	0.67	0.53	0	0	0	0	0	0	0	0.69	0	0	0	0	0	0	0	0	0	0	0	0	0.59	0.54	0	0	0	0.66
M20	0	0.87	0	0	0	0	0	0	0	0	0.74	0.7	0	0	0.77	0	0	0	0.85	0	0	0	0.81	0.63	0	0	0	0	0	0	0.6	0	0	0.96	0	0	0	0	0	0
M21	0.53	0	0	0	0	0	0	0	0.9	0.92	0	0	0	0	0	0.83	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.78	0	0	0	0	0	0	0
M22	0.99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.51	0.82	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.89	0	0	0	0	0	0	0
M23	0	0	0.65	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.57	0	0	0	0	0	0	0.97	0	0	0	0	0	0	0	0
M24	0	0.88	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.72	0	0	0	0	0	0	0	0

Data set 49

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30	P31	P32	P33	P34	P35	P36	P37	P38	P39	P40
M1	0	0	0	0	0	0	0	0	0.53	0	0	0	0	0	0	0	0.99	0	0.83	0	0	0	0	0	0	0	0	0	0	0	0.91	0	0.82	0	0	0	0	0	0	0
M2	0	0	0	0	0	0	0	0	0	0	0	0	0.83	0.91	0	0	0	0	0	0	0	0.92	0	0	0	0	0	0	0	0	0	0	0.86	0	0	0	0	0	0	0.97
M3	0	0	0	0	0	0	0	0	0	0	0	0.79	0	0	0.56	0	0	0	0	0	0	0	0.88	0.53	0	0	0	0	0	0	0	0	0	0.51	0	0	0	0	0	0
M4	0	0	0	0	0	0	0	0.98	0	0	0.83	0	0	0	0	0	0	0	0	0	0.71	0	0	0	0	0	0	0.58	0	0	0	0	0	0.54	0	0	0	0	0	0
M5	0	0	0	0	0.54	0.74	0	0	0	0.63	0	0.63	0	0	0	0	0	0	0	0	0	0.53	0	0	0	0	0	0	0	0	0	0	0	0	0.69	0	0	0	0	0
M6	0	0	0	0	0	0	0	0	0	0.63	0	0	0	0	0	0	0	0	0.68	0	0	0	0	0	0	0.51	0	0	0	0.61	0	0	0	0	0	0	0	0.94	0	0
M7	0	0.68	0	0	0	0	0	0	0	0	0	0	0.67	0	0	0	0	0	0	0	0	0	0	0	0.7	0	0	0.84	0	0	0	0.79	0	0	0	0	0	0	0	0
M8	0	0	0	0.99	0.94	0	0	0	0	0	0	0	0.84	0	0	0	0	0	0	0	0	0	0	0	0.78	0	0	0.93	0	0	0	0.73	0	0	0	0	0	0	0	0
M9	0	0	0	0	0	0.98	0	0	0	0	0	0.92	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.92	0	0	0	0	0	0.7	0.89
M10	0	0	0	0	0	0.52	0	0	0.52	0	0	0	0	0	0	0	0	0	0.54	0	0	0	0	0.77	0	0.76	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M11	0	0	0.96	0	0	0	0	0	0	0.6	0	0	0	0	0.61	0	0	0	0	0	0	0.54	0.67	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M12	0	0	0	0.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0.85	0	0.99	0	0.87	0	0.67	0	0	0.63	0	0	0	0	0	0	0	0	0	0	0	0	0
M13	0	0	0.74	0	0	0	0	0	0	0	0	0.85	0	0	0	0.78	0.55	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.81	0	0	0	0	0
M14	0	0	0	0	0	0	0	0.63	0	0	0.97	0	0	0.54	0	0.52	0	0	0	0	0	0	0	0	0.85	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M15	0.55	0	0	0	0.99	0	0	0	0	0	0	0	0	0.93	0	0	0	0	0	0	0	0	0	0	0	0.94	0.8	0	0	0	0	0	0	0	0.68	0	0	0	0	0
M16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.6	0	0	0	0	0	0	0.63	0	0	0	0	0	0.7	0	0	0	0.9	0.71	0	0.98	0
M17	0	0	0	0	0	0	0.53	0	0	0	0	0	0	0	0	0	0.68	0	0	0.91	0	0	0	0	0	0	0	0	0.53	0	0.76	0	0	0	0	0	0	0	0	0
M18	0	0	0	0.88	0.79	0	0	0	0	0	0.52	0	0	0	0	0	0	0.94	0	0	0	0	0	0	0	0	0.78	0	0	0.52	0	0	0	0	0	0	0	0	0	0
M19	0	0	0	0	0	0	0	0	0.72	0	0	0	0	0.92	0.92	0	0	0	0	0	0	0	0	0	0	0	0.86	0	0	0	0	0	0	0	0.8	0.67	0	0	0	0.53
M20	0	0.69	0	0	0	0	0	0	0	0	0.59	0.54	0	0	0.66	0	0	0	0.87	0	0	0	0	0	0.74	0	0	0	0	0	0.7	0	0	0	0	0	0	0	0	0
M21	0.77	0	0.85	0	0	0	0	0	0.81	0.63	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.6	0	0	0	0	0	0	0
M22	0.96	0	0	0	0	0	0.53	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0.92	0	0	0	0	0	0	0	0	0	0
M23	0	0	0.83	0	0	0	0	0	0	0	0	0.78	0.99	0	0	0	0	0	0	0	0.51	0	0	0	0	0	0	0	0	0	0	0.82	0	0	0	0.89	0	0	0	0
M24	0	0.65	0	0	0	0	0	0.57	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.97	0	0.88	0	0	0	0	0	0

Data set - 50

	M1	M2	M3	M4	M5	M6	M7
P1	1	3	2	0	0	0	0
P2	1	2	0	0	0	0	0
P3	0	0	1	2	0	0	0
P4	0	0	1	2	0	0	0
P5	0	0	0	0	1	2	3
P6	0	0	2	0	4	3	1
P7	0	0	0	3	1	2	4

Data set - 51

	M1	M2	M3	M4	M5	M6	M7	M8
P1	0	0	0	0	2	1	0	0
P2	1	0	2	0	0	0	0	0
P3	2	1	0	5	0	0	3	4
P4	0	1	0	2	0	0	3	4
P5	0	0	0	0	2	1	0	0
P6	0	1	0	2	5	0	3	4
P7	0	4	0	2	0	0	3	1
P8	1	0	2	0	0	0	0	0
P9	1	0	3	0	0	2	0	0
P10	0	0	0	2	3	1	0	0
P11	3	0	2	0	0	0	1	0
P12	0	0	0	0	1	3	2	0
P13	1	0	2	0	0	0	0	0
P14	1	2	3	0	0	0	0	0
P15	0	0	0	1	2	0	0	0
P16	1	0	2	0	0	0	0	0
P17	3	0	1	0	2	0	0	0
P18	0	2	0	1	0	0	4	3
P19	1	0	2	0	0	0	0	0
P20	0	2	0	1	0	3	4	5

Data set - 52

	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12	M13	M14	M15	M16	M17	M18	M19	M20
P1	2	0	0	0	0	0	0	0	3	0	0	1	0	0	0	0	0	4	0	5
P2	0	2	3	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
P3	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	3	2
P4	0	3	1	0	0	0	0	0	0	4	2	0	0	0	0	0	0	0	0	0
P5	0	0	0	1	0	3	4	0	0	0	0	0	0	0	2	0	0	0	0	0
P6	0	0	0	0	5	0	0	0	0	0	1	0	0	2	0	3	4	0	0	0
P7	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	2	3	0	0	0
P8	0	0	0	5	0	0	3	0	4	0	0	0	2	0	1	0	0	0	0	0
P9	4	0	0	0	0	0	0	0	2	0	3	5	0	0	0	0	0	1	0	0
P10	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	1	2
P11	0	0	3	0	0	0	0	1	0	0	0	0	0	2	0	0	0	0	0	0
P12	5	0	0	0	3	0	0	0	1	0	0	4	0	0	0	0	0	2	0	0
P13	0	0	0	0	0	1	2	0	0	0	0	0	0	0	3	0	4	0	0	0
P14	3	4	0	0	0	0	0	1	0	2	0	0	0	0	0	0	0	0	0	0
P15	0	0	0	0	0	0	0	0	0	0	0	0	1	2	0	3	4	0	0	0
P16	0	0	0	0	0	3	2	0	0	0	0	0	0	0	1	0	0	0	4	0
P17	2	0	0	0	0	0	0	0	1	0	0	3	0	0	0	0	0	0	0	0
P18	0	0	0	0	0	0	0	1	0	4	0	0	0	0	0	0	0	0	2	3
P19	0	2	1	0	4	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0
P20	3	0	0	0	0	0	0	0	0	2	0	4	0	0	0	0	0	1	0	0

Data set 53

	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12	M13	M14	M15	M16	M17	M18	M19	M20	M21	M22	M23	M24	M25
P1				5			3			1						4		2				6			
P2	2	3															4								1
P3			2	3							3									1					
P4												1											2		
P5				3								2						1							
P6												3				2							1		
P7							2			5						4		1							
P8				1	1											3			2						
P9			3								4									1					2
P10								2	1																3
P11								2					3								1				
P12	1		4														3							2	5
P13			3	2							2									1					
P14			4		1						2									3					
P15		4		4	3					5						1			2						
P16							3									2		4							
P17							1			3								2							
P18														3	2							1			
P19								1	3	2															
P20																							1		
P21								1	3	2															
P22			3					4	2								1								
P23					2											3			1						
P24					1											2									
P25						1									3						2				
P26												3			4								1		
P27												1									3	2			
P28								2	1	3															
P29					3	2															1				
P30							2									3		1							
P31					2												1		3						
P32													2	1	3							4			
P33											1									3					2
P34												2											1	3	
P35						2									4						1	3			
P36	2	3									4						1								
P37							3					2											1		
P38								2	3													1			
P39												1													
P40	2									3						1									

Data set – 54

	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10
P1	0	0	1	2	0	3	0	0	0	0
P2	0	1	0	0	3	0	0	4	0	2
P3	0	2	0	0	4	0	0	1	0	3
P4	0	0	0	1	0	0	3	0	2	0
P5	3	0	5	4	0	1	0	2	0	0
P6	0	0	0	1	0	0	3	0	2	0
P7	0	1	0	0	3	0	0	0	0	2
P8	0	1	0	0	0	0	0	0	0	2
P9	4	0	1	0	0	2	0	3	0	0
P10	3	0	1	0	0	2	0	0	0	0
P11	0	0	0	0	0	0	1	0	0	0
P12	0	3	0	0	2	0	0	1	0	0

Data set - 55

	M1	M2	M3	M4
P1	0	1	0	0
P2	1	0	2	0
P3	0	1	0	2
P4	1	0	2	0
P5	1	0	0	0

Data set - 56

	M1	M2	M3	M4	M5
P1	0	1	0	2	0
P2	1	0	2	0	0
P3	1	0	2	0	3
P4	0	2	0	1	0
P5	2	1	0	0	3

Data set - 57

	M1	M2	M3	M4	M5
P1	1	2	0	3	0
P2	0	1	2	0	3
P3	2	0	0	1	3
P4	0	1	2	0	3
P5	1	2	0	3	0
P6	3	0	1	0	2
P7	0	3	0	2	1

Data set - 58

	M1	M2	M3	M4	M5	M6
P1	0	1	2	0	3	4
P2	0	0	1	0	3	2
P3	0	0	1	0	2	3
P4	1	0	0	2	0	0
P5	0	0	1	0	0	2
P6	1	2	3	0	0	0
P7	0	1	0	0	0	2
P8	0	2	0	1	0	0

Data set - 59

	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12
P1	1	0	0	2	0	0	0	3	4	0	0	0
P2	1	2	0	4	0	3	6	5	0	0	0	0
P3	1	2	0	3	0	0	4	5	6	0	0	0
P4	1	0	0	2	0	0	3	0	4	0	0	0
P5	1	0	0	0	0	2	4	0	5	3	0	0
P6	0	0	0	0	0	1	3	4	5	2	0	0
P7	0	0	0	2	0	1	0	3	4	0	0	0
P8	0	3	1	5	2	4	0	6	7	0	0	0
P9	0	0	1	4	2	3	0	5	6	0	0	0
P10	0	0	1	3	0	2	0	4	0	0	0	0
P11	0	0	0	0	0	1	0	0	0	0	0	2
P12	0	0	0	0	0	1	2	0	0	3	0	0
P13	0	0	0	0	0	0	2	0	0	1	0	0
P14	0	0	0	0	0	0	2	0	0	0	1	3
P15	0	0	0	0	0	0	3	0	0	2	1	4
P16	0	0	0	0	0	0	2	0	0	3	1	0
P17	0	0	0	0	0	0	0	0	0	2	1	0
P18	0	0	0	0	0	0	0	0	0	0	1	2
P19	0	0	0	0	0	0	2	0	0	0	1	3

Data set – 60

	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12
P1	0	0	3	0	2	1	0	5	0	0	6	4
P2	0	3	0	0	5	1	0	0	2	4	0	0
P3	4	2	0	3	5	0	0	0	0	1	6	0
P4	2	0	4	1	0	0	0	0	0	3	0	0
P5	0	0	2	0	0	0	3	0	1	0	0	0
P6	0	3	0	0	2	4	0	1	0	0	0	0
P7	0	0	0	0	0	0	0	2	3	0	0	1
P8	0	2	0	3	0	0	0	0	1	0	0	0
P9	0	1	3	0	0	0	2	0	0	0	4	5
P10	1	4	0	3	0	0	2	0	5	0	0	0
P11	0	3	2	0	6	0	0	5	0	0	4	1
P12	0	0	3	0	0	0	4	0	2	1	0	0
P13	0	0	0	0	4	0	2	0	0	1	3	0
P14	0	0	1	2	0	0	4	0	0	3	0	0
P15	0	2	0	3	1	0	0	0	0	0	0	0
P16	0	5	4	0	0	1	2	0	0	0	3	0
P17	0	0	0	0	1	0	0	2	3	4	0	0
P18	0	0	0	1	3	0	0	2	0	0	0	0
P19	0	2	1	0	0	0	0	0	4	3	0	5
P20	0	3	0	0	0	1	2	0	0	0	0	0

Data set – 61

	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12	M13	M14	M15	M16	M17	M18	M19	M20
P1	0	0	0	0	0	0	3	0	2	0	0	1	0	0	0	0	0	0	0	0
P2	3	0	0	0	0	0	1	0	0	0	0	2	0	0	0	0	0	0	0	0
P3	1	0	0	0	0	0	4	0	2	0	0	3	0	0	0	0	0	0	0	0
P4	1	0	0	0	0	0	3	0	0	0	0	2	0	0	0	0	0	0	0	0
P5	2	0	0	0	0	0	0	0	1	0	0	3	0	0	0	0	0	5	6	4
P6	0	3	0	0	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P7	0	0	0	0	1	4	0	0	0	0	0	0	0	0	0	2	0	0	3	0
P8	0	3	0	0	0	2	0	0	0	0	0	0	0	0	0	1	0	0	0	0
P9	0	2	0	0	0	4	0	0	0	0	0	0	0	0	0	1	0	0	3	0
P10	0	1	0	0	3	4	0	0	0	0	0	0	0	0	0	2	0	0	0	0
P11	0	0	2	0	0	0	0	1	0	0	3	0	0	0	0	0	0	4	0	0
P12	0	0	1	0	0	0	0	2	0	0	0	0	0	0	0	0	0	3	0	0
P13	0	0	3	0	0	0	0	2	0	0	1	0	0	0	0	0	0	4	0	0
P14	0	0	0	0	0	0	0	0	0	1	0	0	0	3	0	0	4	0	0	2
P15	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	2	0	0	1
P16	0	0	0	0	0	0	0	0	0	2	0	0	0	3	0	0	0	0	0	1
P17	0	0	0	0	0	0	0	0	0	3	0	0	0	2	0	0	1	0	0	0
P18	0	0	0	2	0	0	0	0	0	0	0	0	1	0	3	0	0	0	0	0
P19	0	0	0	1	0	0	0	0	0	0	0	0	2	0	3	0	0	0	0	0
P20	0	0	0	2	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

Data set - 62

	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12	M13	M14	M15
P1	0	0	0	3	0	0	1	0	0	2	0	0	0	0	4
P2	0	0	0	4	0	0	0	2	0	3	1	0	0	0	0
P3	0	0	1	0	0	0	4	0	0	2	0	3	0	0	0
P4	0	0	0	0	2	0	1	0	0	0	0	3	0	4	0
P5	3	0	4	0	0	0	0	0	0	0	2	0	0	0	1
P6	0	0	0	2	3	0	0	0	0	1	0	0	0	0	4
P7	0	0	0	0	0	0	2	0	4	0	0	1	0	3	0
P8	0	0	0	0	3	0	0	0	0	2	0	4	0	1	0
P9	0	3	0	0	0	0	1	0	0	0	0	0	0	4	2
P10	0	0	2	4	1	0	0	0	0	0	0	0	0	3	0
P11	1	0	2	0	0	0	3	0	0	4	0	0	0	0	0
P12	0	0	0	3	2	0	1	0	0	0	0	4	0	0	0
P13	0	0	0	0	0	3	0	4	2	0	0	0	1	0	0
P14	0	0	0	1	0	2	0	0	0	3	4	0	0	0	0
P15	0	0	0	0	0	0	1	3	0	0	4	0	2	0	0
P16	0	0	0	0	0	0	3	4	0	0	2	1	0	0	0
P17	0	0	0	0	3	0	0	0	0	0	0	4	0	1	2
P18	0	0	0	0	1	0	0	0	4	0	0	3	0	2	0
P19	0	3	0	0	0	2	0	0	0	4	0	0	0	0	1
P20	0	2	0	0	0	0	0	0	0	0	1	3	4	0	0
P21	4	0	0	0	0	1	0	0	3	0	0	0	2	0	0
P22	0	0	2	3	0	0	0	0	0	4	0	0	1	0	0
P23	0	1	0	0	0	0	0	0	0	0	0	4	3	2	0
P24	2	0	0	3	0	0	1	0	0	0	4	0	0	0	0
P25	0	0	0	0	0	0	2	3	0	0	0	1	0	4	0
P26	0	0	0	0	0	2	3	1	0	4	0	0	0	0	0
P27	4	0	2	0	0	0	0	0	3	1	0	0	0	0	0
P28	0	0	0	0	0	4	0	2	0	0	1	3	0	0	0
P29	3	0	0	0	0	1	0	0	0	0	0	4	0	2	0
P30	4	0	0	0	2	0	0	0	3	0	0	1	0	0	0

Data set – 63

	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12	M13	M14	M15	M16	M17	M18	M19	M20
P1	0	0	2	3	0	0	0	0	0	1	0	4	0	0	0	0	0	0	0	0
P2	0	0	0	0	0	0	0	4	0	0	0	0	0	1	0	0	2	3	0	0
P3	0	0	4	3	0	0	0	0	0	0	0	0	0	0	0	0	0	2	1	0
P4	0	0	1	0	0	0	0	0	0	0	4	0	0	2	3	0	0	0	0	0
P5	0	0	0	0	3	0	0	0	0	0	0	0	0	1	0	0	0	4	2	0
P6	0	0	0	0	0	0	0	0	3	0	1	4	0	0	0	0	0	2	0	0
P7	0	0	2	1	0	0	0	0	0	0	0	0	0	0	0	4	0	3	0	0
P8	0	0	0	0	0	0	0	0	0	0	1	0	0	4	2	0	0	3	0	0
P9	0	0	0	2	3	0	0	0	0	0	0	0	0	0	0	0	0	0	4	1
P10	2	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	4	0	0	3
P11	0	0	0	0	2	0	0	0	3	0	0	1	0	0	0	0	0	0	4	0
P12	0	0	0	0	0	0	3	0	1	0	0	0	0	4	2	0	0	0	0	0
P13	0	0	0	2	0	0	0	0	0	4	0	0	0	0	3	1	0	0	0	0
P14	0	0	0	0	0	0	1	0	0	0	3	0	0	0	0	0	0	4	0	2
P15	0	0	0	2	0	0	4	0	0	0	0	0	0	0	0	0	0	3	0	1
P16	0	0	0	0	0	3	0	0	0	0	0	1	0	0	2	0	0	0	4	0
P17	0	0	0	0	2	0	4	0	0	0	0	3	0	0	0	0	0	0	0	1
P18	0	2	1	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	3
P19	0	0	2	0	0	0	0	0	0	1	0	0	0	0	0	0	0	3	4	0
P20	0	2	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	4	0	1
P21	0	0	3	4	0	0	0	0	0	0	0	2	1	0	0	0	0	0	0	0
P22	0	0	0	0	0	0	1	0	0	2	0	0	0	0	0	3	0	0	4	0
P23	0	0	0	0	1	3	0	0	0	0	0	0	0	0	2	0	0	4	0	0
P24	0	0	1	0	4	0	0	0	0	3	0	0	0	0	2	0	0	0	0	0
P25	0	2	1	0	0	4	0	0	0	3	0	0	0	0	0	0	0	0	0	0
P26	0	0	2	0	4	0	3	0	0	0	0	0	0	1	0	0	0	0	0	0
P27	2	0	0	0	0	0	0	0	4	1	0	0	0	0	3	0	0	0	0	0
P28	0	0	0	4	0	2	0	0	0	0	3	0	0	0	1	0	0	0	0	0
P29	4	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	2	3	0	0
P30	0	1	0	0	0	0	0	0	2	0	3	0	0	0	0	0	4	0	0	0
P31	0	0	0	0	0	0	0	0	0	4	2	0	0	0	0	0	0	1	3	0
P32	0	0	0	3	0	0	0	2	0	0	0	0	0	4	0	0	0	0	0	1
P33	0	0	0	0	0	0	0	0	0	4	1	0	0	0	0	2	0	0	0	3
P34	0	0	0	0	0	0	0	0	0	0	0	0	0	3	4	2	0	0	1	0
P35	0	0	0	1	0	0	4	0	0	0	0	2	0	0	0	0	3	0	0	0
P36	4	0	0	0	3	1	0	0	0	0	0	0	0	0	0	0	0	2	0	0
P37	0	0	0	0	0	0	0	0	0	0	0	0	0	2	4	0	1	3	0	0

Data set – 64

	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12	M13	M14	M15	M16	M17	M18	M19	M20	M21	M22	M23	M24	M25
P1	1	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	4
P2	0	0	3	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	2	0	4	0	0	0	0
P3	2	0	0	0	0	0	0	0	0	0	0	0	0	3	1	0	0	0	4	0	0	0	0	0	0
P4	0	0	4	1	3	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P5	0	1	0	0	3	0	0	0	2	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0
P6	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	1	0	4	0	0	0	0	0	2
P7	0	0	0	0	1	0	0	0	0	4	0	3	0	0	0	0	0	0	0	0	0	0	0	0	2
P8	0	1	0	0	0	0	0	4	0	3	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0
P9	0	0	3	0	0	0	1	0	4	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0
P10	3	0	0	1	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0
P11	0	0	3	0	0	0	0	4	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	1
P12	0	0	1	0	0	4	3	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P13	0	0	0	0	0	0	0	2	3	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
P14	4	0	0	0	0	0	2	0	0	1	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0
P15	0	0	0	0	0	1	0	4	0	0	0	0	0	0	2	0	0	0	3	0	0	0	0	0	0
P16	0	0	0	2	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	4	0	1	0	0	0
P17	0	0	0	0	0	0	1	0	3	0	0	0	0	0	0	0	0	0	4	0	0	2	0	0	0
P18	0	0	0	0	0	1	3	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	4
P19	0	1	0	3	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	2	0	0
P20	0	0	2	0	0	0	3	0	0	0	4	0	0	0	0	0	0	0	0	0	1	0	0	0	0
P21	4	0	2	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0
P22	0	0	0	0	4	0	0	0	2	0	3	0	0	0	1	0	0	0	0	0	0	0	0	0	0
P23	0	0	2	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	4	0	0	0	0
P24	0	0	0	0	0	0	3	0	0	0	0	0	0	0	2	0	1	0	0	0	0	0	0	0	4
P25	4	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	2	0	0	1
P26	0	0	0	0	0	0	2	3	0	0	1	0	0	0	0	0	0	4	0	0	0	0	0	0	0
P27	0	0	0	0	0	0	0	0	0	0	0	0	4	1	0	0	0	0	0	0	0	0	0	3	2
P28	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	1	0	0	4	3	0	0	0
P29	0	0	4	0	0	0	0	0	0	0	3	0	0	2	0	0	0	0	0	0	0	0	0	1	0
P30	0	0	0	0	0	0	0	0	0	4	2	1	0	0	0	0	0	0	0	0	0	0	0	3	0
P31	0	2	0	0	0	0	3	0	0	1	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0
P32	0	0	0	0	2	0	3	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	4	0
P33	4	0	0	0	0	0	0	0	0	0	0	0	0	3	2	0	0	0	0	0	0	0	0	1	0
P34	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	1	0	0	0	3	0	0	0	4	0
P35	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	2	1	0	0	4	0	0	0	0	0
P36	0	0	0	0	0	0	0	1	0	2	4	0	3	0	0	0	0	0	0	0	0	0	0	0	0
P37	0	0	0	0	0	0	1	0	0	0	0	2	4	0	3	0	0	0	0	0	0	0	0	0	0
P38	0	0	0	0	0	0	0	4	3	0	0	0	0	0	0	0	0	0	2	0	0	0	1	0	0
P39	2	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	4	0	0	0	3	0	0
P40	0	0	0	0	0	0	0	0	0	0	0	3	0	1	0	0	4	0	0	2	0	0	0	0	0
P41	0	2	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	3	0	0	0	0	1	0	0
P42	0	0	1	3	0	0	0	0	4	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0
P43	0	2	0	0	1	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	3
P44	0	0	0	0	0	0	0	0	0	0	0	3	0	1	0	0	0	0	2	0	4	0	0	0	0
P45	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	2	0	3
P46	0	0	0	0	0	0	0	0	4	0	0	1	0	2	3	0	0	0	0	0	0	0	0	0	0
P47	3	4	0	0	0	0	2	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
P48	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	3	0	0	1	4	0
P49	0	0	0	0	0	0	0	0	0	1	0	3	0	0	0	0	0	0	0	0	2	4	0	0	0
P50	0	0	0	0	0	0	4	0	0	0	1	0	0	0	0	0	0	0	0	0	0	2	3	0	0

Data set - 65

	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12	M13	M14	M15	M16	M17	M18	M19	M20
P1	1	0	0	0	0	0	0	2	0	7	5	0	0	3	6	0	0	0	0	4
P2	0	0	1	0	2	0	0	0	0	3	7	6	0	0	4	5	0	0	0	0
P3	0	0	0	0	4	2	0	0	0	0	1	0	6	0	5	0	0	7	3	0
P4	0	0	7	0	0	0	0	0	0	0	2	3	6	0	0	5	0	4	0	1
P5	0	0	2	1	5	0	0	0	0	7	0	3	0	4	0	0	0	0	6	0
P6	0	0	0	3	0	5	0	0	6	7	0	0	0	0	0	0	2	4	1	0
P7	0	0	0	0	0	0	6	2	0	0	3	0	7	0	1	0	4	0	5	0
P8	0	0	0	0	7	1	0	0	3	5	0	0	2	0	4	0	0	0	6	0
P9	2	0	0	0	0	0	0	5	0	0	0	6	0	0	7	3	4	0	0	1
P10	6	0	0	0	0	5	0	4	2	0	0	1	0	0	0	0	0	0	7	3
P11	6	0	7	0	0	5	4	3	0	1	0	0	0	0	0	2	0	0	0	0
P12	6	0	0	5	0	3	7	0	2	0	0	0	0	0	0	0	1	0	4	0
P13	5	0	0	0	0	7	0	1	3	2	0	0	0	0	0	0	0	6	4	0
P14	0	2	6	0	5	0	0	0	0	0	0	3	7	0	1	0	0	4	0	0
P15	0	0	2	6	0	0	0	3	0	1	4	0	7	5	0	0	0	0	0	0
P16	0	4	2	3	0	0	7	0	0	0	0	0	6	0	0	0	1	0	5	0
P17	7	0	1	0	0	0	0	0	0	0	5	0	4	0	2	0	0	6	3	0
P18	0	3	0	1	0	5	0	0	4	0	2	0	0	0	6	0	0	7	0	0
P19	6	0	0	0	0	0	4	0	5	0	1	7	0	0	2	0	0	0	0	3
P20	0	4	7	5	0	0	6	0	0	0	0	0	3	0	0	2	0	0	1	0
P21	7	0	2	0	1	0	0	0	0	5	0	0	6	0	0	0	4	0	3	0
P22	0	0	4	1	0	0	0	2	0	0	5	6	0	0	7	0	0	0	3	0
P23	1	7	0	0	0	0	0	0	0	6	5	0	4	0	3	0	0	2	0	0
P24	4	0	0	5	7	2	0	0	0	0	0	0	0	0	1	6	0	0	0	3
P25	7	3	0	0	0	0	6	2	0	0	0	0	0	0	0	0	5	1	0	4
P26	7	0	0	5	1	0	0	0	0	3	0	0	6	0	0	0	2	0	4	0
P27	0	1	2	0	0	6	0	7	5	3	0	4	0	0	0	0	0	0	0	0
P28	7	6	0	0	2	0	0	0	0	0	0	1	0	0	0	0	5	4	3	0
P29	0	3	0	0	6	0	1	0	0	4	5	0	2	0	0	0	7	0	0	0
P30	0	5	0	0	0	0	0	7	3	0	0	0	1	4	6	0	0	0	2	0
P31	0	3	4	0	0	0	0	1	0	0	7	5	0	0	0	2	0	0	6	0
P32	0	0	0	2	4	0	7	0	0	0	1	0	3	0	6	0	5	0	0	0
P33	0	0	0	5	0	1	0	0	0	0	7	0	0	3	4	0	6	0	2	0
P34	0	0	2	0	0	4	6	7	0	1	0	0	0	0	0	0	5	3	0	0
P35	0	6	0	0	7	0	0	0	2	1	4	5	0	0	0	0	0	0	0	3
P36	0	0	0	0	6	0	0	0	1	0	5	0	0	0	0	0	7	2	4	3
P37	2	0	0	3	4	0	1	0	0	0	0	0	6	0	0	0	7	0	5	0
P38	0	0	1	3	0	6	0	0	0	0	0	0	0	4	0	5	0	0	7	2
P39	0	0	2	0	0	6	0	0	0	4	7	5	0	0	3	0	1	0	0	0
P40	0	6	0	0	0	5	2	1	3	0	0	0	4	0	0	0	0	0	7	0
P41	0	0	0	0	0	0	2	0	4	7	0	0	3	0	6	5	0	0	1	0
P42	0	0	3	0	0	0	0	0	0	4	0	0	7	6	1	2	0	0	5	0
P43	0	0	0	1	0	0	0	0	7	0	6	0	4	0	0	3	5	2	0	0
P44	0	0	0	7	5	3	0	1	0	0	4	0	0	0	0	0	2	6	0	0
P45	0	0	0	0	3	0	0	5	2	0	6	4	0	0	0	0	1	0	0	7
P46	0	0	0	7	0	2	0	6	4	0	1	0	0	0	3	0	0	5	0	0
P47	0	0	2	0	0	6	0	0	0	0	7	0	0	1	3	5	0	4	0	0
P48	0	2	0	0	7	0	5	0	4	0	6	0	3	0	1	0	0	0	0	0
P49	0	2	0	0	7	1	3	0	0	0	6	0	0	0	0	4	5	0	0	0
P50	3	5	0	0	6	0	7	1	0	0	0	0	2	0	0	0	0	4	0	0
P51	0	0	0	2	0	0	0	0	6	0	5	0	0	4	0	0	0	1	7	3
P52	0	0	0	4	2	0	0	6	0	1	0	0	0	0	7	3	0	5	0	0
P53	0	0	0	1	2	0	4	5	0	0	0	3	0	0	0	0	6	0	0	7
P54	0	4	0	3	0	0	2	0	1	0	0	0	7	0	0	5	6	0	0	0
P55	0	0	4	0	0	0	3	0	0	2	0	1	0	7	0	0	0	0	5	6

Data set - 66

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28
P1	0	0	0	0	0	0	0	0	2	0	0	0	0	1	0	0	0	4	0	0	0	0	0	0	0	0	3	0
P2	2	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	3	0	0	0	0
P3	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	3	0	0	0	0	1	0	2	0	0	0	0
P4	0	0	0	0	2	0	0	0	0	0	0	0	0	3	0	0	4	0	0	0	0	0	0	0	0	0	0	1
P5	0	0	0	0	0	0	3	0	0	0	2	4	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
P6	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	1	0	0	3	0	0	0
P7	0	0	0	0	0	0	1	0	0	2	4	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0
P8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	4	0	0	0	2	0	0	0	1	0	0
P9	0	0	0	0	0	3	0	0	0	0	0	2	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	1
P10	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	3	2
P11	0	0	0	0	0	4	0	0	0	2	0	0	0	0	0	0	0	0	0	3	1	0	0	0	0	0	0	0
P12	0	0	0	0	2	0	1	0	0	0	0	0	4	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0
P13	0	2	0	0	0	0	3	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	4	0	0	0	0	0
P14	0	0	0	0	0	0	0	0	3	4	1	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P15	0	0	0	0	0	0	0	0	0	0	2	0	0	4	0	0	0	0	0	0	0	1	0	0	0	0	3	0
P16	0	0	0	0	0	0	0	3	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	2	0	0	1	0
P17	0	0	0	0	0	2	0	0	0	0	0	4	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	3
P18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	4	2	3	0	0
P19	0	2	0	4	0	0	0	0	3	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P20	0	0	3	0	0	0	2	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0
P21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	4	3	2
P22	0	3	0	0	0	0	0	1	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	2	0	0	0	0
P23	0	0	2	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	4
P24	0	0	0	0	0	0	3	0	2	0	0	0	0	0	4	0	0	0	0	0	1	0	0	0	0	0	0	0
P25	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	2	3	0	0	0	0	0	0	0
P26	0	0	0	0	0	3	0	0	0	0	0	0	0	4	0	0	2	0	1	0	0	0	0	0	0	0	0	0
P27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	2	0	4	3	0
P28	0	0	0	0	0	0	0	0	0	1	0	0	0	3	0	0	0	0	0	0	4	0	0	0	0	2	0	0
P29	0	0	0	0	0	0	2	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	1	0	0
P30	0	3	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	2	0	1	0	0	0	0	0	0
P31	0	0	1	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	4	2	0	0	0	0	0	0	0	0
P32	1	0	0	0	0	0	2	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	3
P33	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	1	4	2	0	0	0	0	0	0	0	0	0	0
P34	0	0	0	0	4	0	0	1	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	3	0	0
P35	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	3	0	0	0	0	0	1	0	2
P36	0	0	0	1	0	4	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	2	0	0	0	0	0	0
P37	0	0	3	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	1	4	0	0	0
P38	0	0	0	2	0	0	0	0	0	0	4	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
P39	0	0	0	0	0	0	0	0	0	0	0	4	0	2	0	0	0	0	0	0	1	0	0	0	3	0	0	0
P40	0	0	3	0	0	4	0	0	2	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
P41	0	2	0	0	0	0	0	4	0	0	0	0	1	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0
P42	0	0	0	0	0	0	0	1	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	3	0	0	2	0
P43	0	0	0	0	4	1	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0
P44	0	0	0	0	0	0	3	1	0	0	0	0	0	0	0	0	0	2	0	0	0	4	0	0	0	0	0	0
P45	0	0	2	3	0	1	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P46	0	0	0	2	0	0	0	0	0	0	0	0	0	3	0	0	0	4	0	0	0	0	1	0	0	0	0	0
P47	0	0	0	2	0	0	0	0	0	1	0	0	0	0	0	3	0	0	0	0	0	4	0	0	0	0	0	0
P48	0	0	0	4	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	3	0	0	0	2	0	0
P49	4	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	2	0
P50	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	3	2	0	0	0
P51	0	0	0	0	0	0	0	0	0	0	1	0	4	0	0	0	0	0	0	0	0	2	0	0	0	3	0	0
P52	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	3	0	0	0	0	0	2	0	0
P53	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	3	0	0	0	2
P54	1	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	2
P55	0	0	0	0	0	0	2	1	0	0	0	3	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0
P56	0	0	0	0	0	0	2	0	1	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	4	0	0	0
P57	0	2	0	0	0	3	0	0	0	0	0	0	0	1	0	0	0	0	0	0	4	0	0	0	0	0	0	0
P58	0	0	0	0	2	4	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
P59	0	0	0	1	0	0	0	0	0	4	0	0	0	0	2	0	0	0	0	0	0	0	0	3	0	0	0	0
P60	0	0	0	0	0	0	2	0	0	0	0	4	0	0	0	0	0	0	0	0	0	3	0	1	0	0	0	0

Data set 67

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33
M1		3								1																			5				
M2	3										3					2															2		
M3	4		5	3		5	4			2																						5	
M4					4	4													2	5						1	5				3		
M5					5						1												5	2									
M6		4											3		2	1						4											
M7			1	5						5	4	3	1	2		5									3		2						
M8		2			1	2	1		5			5		1						4								1					2
M9							3		2				5				5					2		3				2	3				
M10	5						2				2														5	5							1
M11									3			4			1												4						
M12			2	4		3														1						2				2		2	
M13																	1			3			1										3
M14	2											1			5		4					3		4						1			
M15								3			5							1					3					4					
M16										3				3																4			
M17								2					2												4								
M18				2			5							5				5	4			5							1			3	
M19	1								1						4		2						4		2				4				
M20																3					2												
M21																		3	1			1				3	3						5
M22								1	4	4											4									5			
M23																4		4						5									
M24			3										4					2					2	1	1							1	
M25			4									2		4					5		1							5					4
M26																					5					4					4		
M27		5			2															2											5		
M28						1		4							3		3				3						1						
M29		1			3			5											3									3		3	1	4	
M30				1																									2				

	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65
M1			5							4											3									1		
M2															4											1						
M3														4									5	5				3				
M4	3				1																							1				
M5										3			3	3					2		2				1		3					
M6			4																3	1										2		
M7		1										4							5						2	5					4	
M8			3			4																		1								
M9							5	5									1									3				3	3	
M10					4					1		3				1		3		2			2									
M11	1	4			2	3		4							1	2	5			3		2				2		5		4	1	
M12		3	2					1	1						5									3					4			
M13	5			2			2										4				1		4						5			
M14							4							2				5														
M15									5		3														5		2			5		5
M16														5																		1
M17						1	1						5	1						5							4					
M18																		2			5							4				3
M19											4	1				5	2	4				1		2	4			2				
M20			1							2																						2
M21	4																								3		1					
M22	2			3						5		2	1		2			1				4	3				5				2	
M23									4											4				4					3			
M24				1				3	3		1																		1			
M25				5									4		3		3															
M26				4		2					5					3					4										5	4
M27		2			5								2																			
M28								2	2		2	5							4													
M29		5				5	3									4						5										
M30					3														1			3	1			4			2			

Data set 68

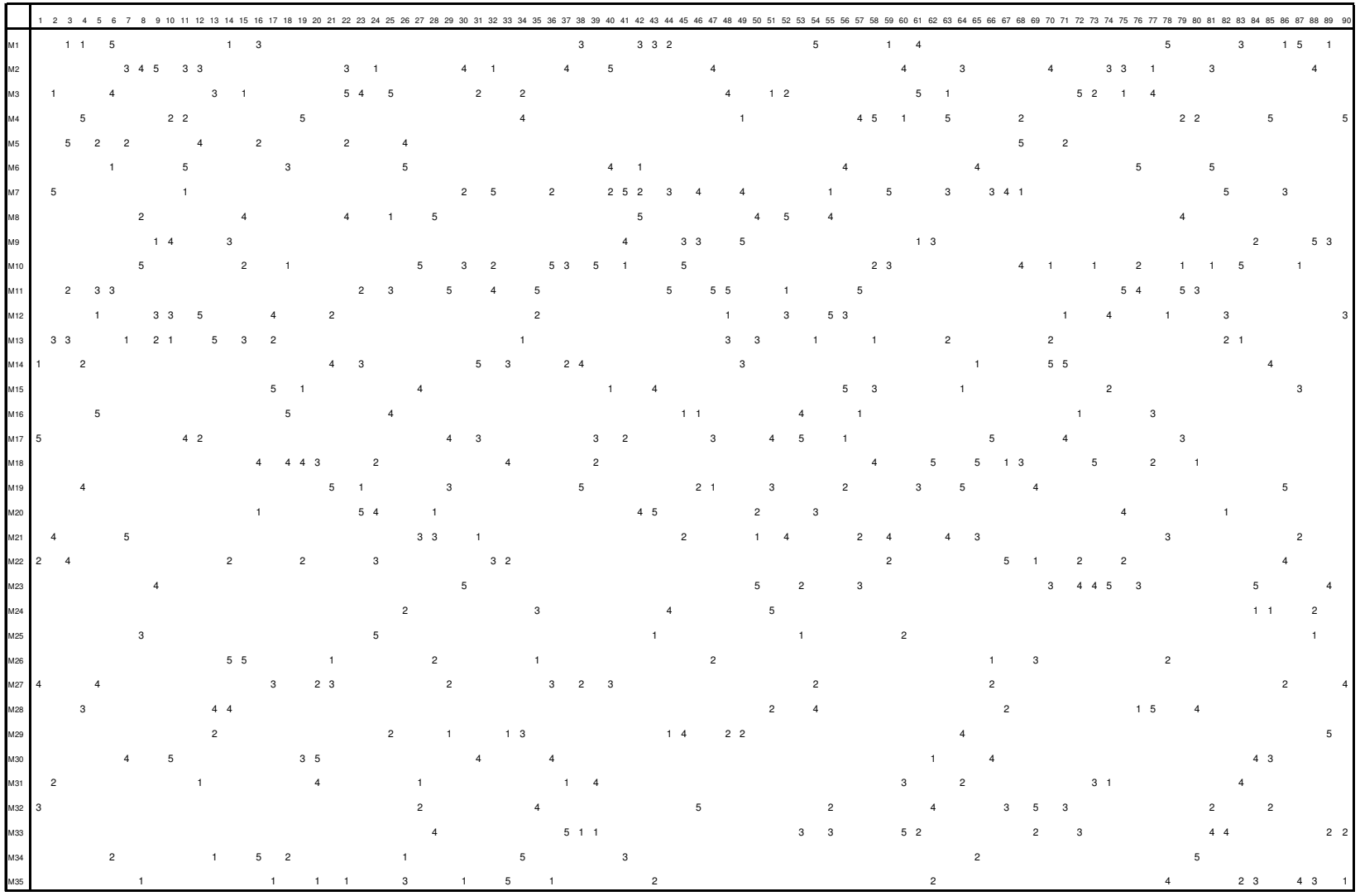
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
M1			1	3		2				1		3								
M2																				
M3		3							2											
M4	2	4														1			4	
M5																				
M6		2									4									1
M7				1	3	3				3										2
M8			2		2		1													
M9					1															
M10											2									
M11														2						4
M12						4	3								3			1		
M13								1												
M14					4					4				4						
M15									4			2					1			
M16									1	2										
M17				2								4		1	2		2			
M18		1				1														
M19																	3		1	
M20	3												4							
M21								2									4			
M22	4										3	1	1			3				
M23	1															2				
M24																				
M25																				
M26														3				2		
M27			3															4		
M28			4				4						3							
M29									3							4		3	2	
M30											1								3	3
M31				4				4												
M32							2	3					2		4					

	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
M1	1						3													
M2			2								3		4							
M3		1															1			
M4		4																3		
M5					3	1			4										3	
M6		2			2			1	1					3						
M7										3			2							
M8																				
M9				1											4					3
M10								3	3											
M11				3														2		
M12	3																			
M13						3							3							
M14					1															
M15	2				4		2					2	1					4		
M16						4		4				4		2			2			
M17										4									4	
M18			4									1						1		
M19																	4			
M20															1					
M21														1		3				
M22			3	4		2	1				1								1	1
M23																	3			
M24							4				2				2	4			2	
M25								2												
M26																				
M27				2								3								
M28		3								1					3	2				
M29											4			4		1				2
M30									2	2										
M31	4																			4
M32			1																	

	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
M1											3						4		4	
M2			2								1									
M3												2			1	2				
M4			1		4								1							
M5						1														
M6																	3			
M7											2									
M8					1				4					1						
M9	3																			2
M10	4			3						2										
M11							1						2							
M12			3					3				1	4	2						
M13		4		2					3											
M14							3	4	1						2					
M15													3				2			
M16	2			4		2		2								1			2	
M17		3							2			3		4						
M18				1														4		
M19										3										
M20								1								4	1	3		4
M21						4								3						
M22		2			3															
M23	1					3														
M24							4								3			2	3	
M25							2			1										
M26												4								
M27		1																		3
M28										4									1	
M29											4									
M30			4															1		
M31					2										4	3				
M32																				1

	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
M1								1							1	4		2		
M2			4						1				2	4						
M3																				
M4					4				4	1			3							
M5		3				3												1		
M6											2								2	
M7					1															
M8		4										3								
M9				3								1	1							2
M10								2		2						1			4	
M11																				
M12		2			3									2						
M13			3							3										4
M14									2											
M15				1		2	3									2				
M16			1				2													
M17							1													
M18	3					1					3								3	
M19										4							1			
M20		1										4								
M21											1						4			
M22																3		4		
M23					2															
M24	1													3						
M25				4					3								2			
M26	4							3												
M27											4				4					3
M28								4												1
M29				2		4														
M30							4					2					3	3		
M31	2												4	1	3					
M32			2												2				1	

Data set 69



PUBLICATION FROM THE THESIS WORK

1. **SudhakaraPandian, R.** and Mahapatra, S.S. 2008. Cell Formation with Ordinal Level Data Using ART1 Based Neural Networks. *International Journal of Services and Operations Management,* 4 (5) 618-630.

2. Mahapatra, S.S. and **SudhakaraPandian, R.** 2007. Machine Cell Formation in CMS using Ratio level Data. *International Journal of* Advanced *Manufacturing Technology*, DOI-10.1007/s00170-007-1029-5

3. **SudhakaraPandian, R.** and Mahapatra, S.S. 2007. Cell formation with workload data in cellular manufacturing system using genetic algorithm. Proceedings of "*The International Conference on Industrial Engineering and Engineering Management-IEEM07"*, December, 2-5 Singapore.

4. Mahapatra, S.S. and **SudhakaraPandian, R.** 2006. Genetic cell formation using ratio level data in CMS, *AIMTDR 2006*, December 21-23, IIT Roorkee.

5. Mahapatra, S.S. and **SudhakaraPandian, R.** 2006.Cell formation using MART for ratio level data. *Sixth International conference-COSMAR 2006*, September, 21-23, IISc, Bangalore.

6. Mahapatra, S.S. and **SudhakaraPandian, R.** 2006. A metaheuristic approach for cell formation using operational time in cellular manufacturing systems. *ICOMAST 2006*, August, 28-30, MMU, Melaka, Malaysia. (Sponsored by CSIR)

7. Mahapatra, S.S. and **SudhakaraPandian, R.** 2006. Cell formation using workload data in cellular manufacturing systems a metaheuristic approach. *47th Annual Technical Session*, January, 22-24, Institution of Engineers (India). Orissa State Centre. Bhubaneswar. (Won the Best paper award in Industrial Engineering)

8. Mahapatra, S.S. and **SudhakaraPandian, R.** 2005. Generalized cell formation in CMS using metaheuristic algorithm. ICORAID-2005-ORSI, December, 27-29, IISc, Bangalore.

9. Mahapatra, S.S. and **SudhakaraPandian, R.** 2005. Generalized machine cell formation using modified ART1. *ICPQR-2005*. December, 12-15, IIT Delhi.

Communicated

1. **SudhakaraPandian, R.** and Mahapatra, S.S. Manufacturing cell formation with production data using neural networks. Computers and Industrial Engineering. (Revised version submitted)