A Comprehensive Review of Production-Oriented Manufacturing Cell Formation Techniques

Jeffrey A. Joines, Graduate Research Assistant
Russell E. King, Associate Professor of Industrial Engineering
C. Thomas Culbreth, Associate Professor of Industrial Engineering

Keywords: cellular manufacturing, group technology, cluster analysis

Department of Industrial Engineering
Furniture Manufacturing and Management Center
North Carolina State University
Box 7906
Raleigh, NC 27695-7906

August 1, 1996

This research was funded in part by the
NCSU Furniture Manufacturing and Management Center, and the
National Science Foundation (under grant number DDm-92-15432)

Abstract

This paper offers a comprehensive review and classification of techniques to manipulate part routing sequences for manufacturing cell formation. Individual techniques are aggregated into methodological groups including array-based clustering, hierarchical clustering, non-h ierarchical clustering, graph theoretic approaches, artificial intelligence, math programming, and other heuristic approaches. Discussion of each model includes assumptions, analytic approach, performance criteria, and limitations. Where possible, empirical results and comparisons of methods are provided. Evaluation measures are discussed in terms of their practical consequences on the cell design process. Recommendations are made for future research in the domain of stochastic search techniques.

1 Introduction

International competitiveness and market demand for rapid response have led many firms to consider non-traditional approaches to the design and control of manufacturing systems. Flexibility and efficiency in producing a large number of products in small-to-medium lot sizes are necessary to be competitive. One approach is the application of Group Technology (GT) described as "recognizing and exploiting similarities in three distinct ways: (1) by performing like activities together, (2) by standardizing similar tasks, and (3) by efficiently storing and retrieving information about recurring problems."[155] In essence, GT attempts to decompose the manufacturing system into several manageable subsystems or groups.[194, 248, 283]

One important facet of GT is the development of a cellular manufacturing (CM) system in which similar parts are aggregated into part families and dissimilar machines are grouped into cells. The ideal cell (1) is independent, i.e., part family(s) are completely produced within the cell; (2) has balanced setups; and (3) requires minimal backtracking. The result is simplified scheduling, control, and implementation of automation. CM provides the benefits of a mass production system for a discrete part, batch production system[59, 82, 79] including reduced setup times, work in progress (WIP), throughput time, and material handling as well as encouraging improved product quality.[35, 155, 272, 372] Employee/worker benefits include worker flexibility, importance of social group, reduced frustration, and improved status and job security.[107]

At the highest level, methods for part family/machine cell formation can be classified as design-oriented or production- oriented. Design-oriented approaches group parts into families based on similar design features while production-oriented techniques aggregate parts requiring similar processing. Classification and coding schemes are design-oriented tools that can be used to implement GT applications.[202, 155, 269, 268, 372] An overview of classification and coding is presented by Askin and Vakharia[17] while a survey of the various techniques is provided by Ham et al.[135] Analysis of codes facilitates rapid prototyping, the development of new parts, and can be used for machine cell formation. Since part codes are assigned based upon physical geometry, parts having similar design features have similar codes providing a weak connection between part features and machine groupings.[83, 179, 371] Ham and Han[136] as well as Jung[165] have developed a multi- objective cluster analysis tool using design features to form machine cells while Offodile [266, 264] used a hierarchical clustering technique. Lee et al.,[210] Li and Ding,[213] Xu and Wang,[378] and Ben-Arieh and Triantaphyllou[24] used fuzzy logic and design features to form part families while Dutta et al. applied design-based grouping to a flexible manufacturing system.[102] Several researchers have used the artificial neural network back-propagation algorithm to form part families from design features [86, 171, 166, 243] while Awwal[19] used Hopfield neural networks. El Maraghy and Gu[104] used expert systems, pattern recognition and formal languages to design cells based on design features. Classification and coding involves substantial implementation effort and cost. Much prerequisite part data must be developed in order to apply the design-oriented techniques.[349] To aid in this developement, Billo et al. [34, 33, 30] and Huq et al. [153] have applied engineering database modeling and object-oriented modeling principles to the classification and coding problem.

This paper focuses on production-oriented cell formation techniques that manipulate part routing sequences. Early work in this area was performed by Mitrofanov [239, 240] and Burbidge.[38] Production flow analysis (*PFA*) by Burbidge [38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 53, 53, 99, 373] is one of the first and most comprehensively recognized methodologies associated with GT.[113] Component flow analysis by El-Essawy and Torrance[103]is considered by many to be equivalent to *PFA*. The goal of most of these techniques is to obtain independent machine cells by minimizing intercell movement.

Within the same context, previous reviews of clustering methods for the cell formation problem include. [70, 21, 79, 82, 150, 146, 182, 193, 200, 252, 248, 265, 309, 325, 371] Comprehensive comparisons of several cell formation techniques have been developed.[55, 342, 131, 129, 37, 60, 85, 83, 145, 168, 197, 193, 203, 236, 250, 299, 304, 314, 321, 351, 352] In addition, several researchers have utilized simulation as a means to compare different cell formation solutions, design cells, and to justify manufacturing cell formation/group technology versus functional and/or job shop layouts.[341, 340, 8, 27, 54, 88, 93, 101, 106, 109, 110, 115, 128, 228, 246, 289, 294, 315, 322, 336, 343, 313, 320, 247, 116, 319, 261, 270, 271]

Variations in breadth of coverage, timeliness, and level of detail concerning model description exist in all previous reviews. The objective of this paper is to provide an exhaustive survey of research to date including emerging work in the area of stochastic search techniques such as genetic algorithms and neural network approaches. Models are discussed in terms of assumptions, analytic methods, performance criteria, and limitations. Emphasis is given to reporting empirical results and comparative evaluations of techniques. A methodological classification of techniques (shown in Figure 1) is adopted to improve readability and to facilitate an understanding of the basic advantages/limits of generic approaches. The impact of various cell formation evaluation measures on the cell design process is discussed from a practical standpoint. Finally, recommendations on the most promising techniques for continuing research are offered.

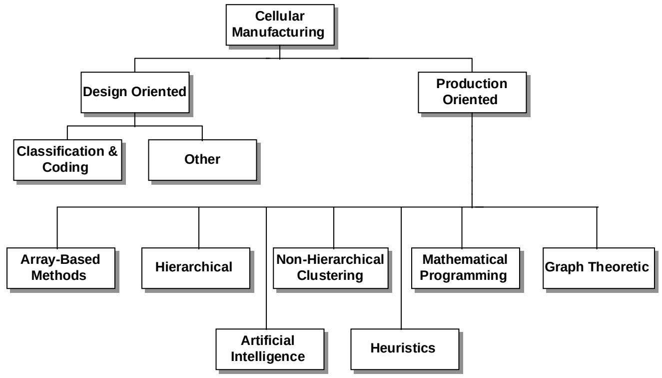

Figure 1: Categories of Grouping Approaches

2 Array-Based Clustering

Array-based clustering is one of the simplest classes of production-oriented cell formation methods. It operates on a 0-1 part/machine incidence array performing a series of column and row manipulations trying to produce small clustered blocks along the diagonal (see Figure 2). The part/machine incidence matrix, A, consists of elements $a_{ij} = 1$ if part j requires processing on machine i, otherwise $a_{ij} = 0$. Any tightly clustered blocks represent the candidate part families and machine cells, which are formed simultaneously. Chandrasekharan and Rajagopalan [64] and Venugopal and Narendran[360] have done analysis of the 0-1 machine/part incidence matrix in order to exploit properties of this matrix for developing cell formation algorithms. Arvindh and Irani [12] used principal component analysis to determine the block diagonal format (BDF) capability of the matrix before clustering. Table 1 contains the major contributions for this method.

	1	2	3	4	5	6	7
1		1			1		
2	1						1
3			1	1		1	
4			1	1		1	
5	1						1
6			1	1		1	
7		1			1		

(a) Part/Machine Matrix

	1	7	3	4	6	2	5
2	1	1					
5	1	1					
3			1	1	1		
4			1	1	1		
6			1	1	1		
1						1	1
7						1	1

(b) Optimal Clustering

Figure 2: Part/Machine Matrix and Optimal Clustering

Table 1: Contributions in Array-Based Clustering

Method	Approach	References
BEA	General	[4, 9, 83, 121, 120, 122, 233]
	Worst Case Bound	[257]
	TSP	[212, 257]
DCA		[59, 83, 370]
ROC	General	[83, 87, 180, 181, 235]
	ROC2	[182]
	MODROC	[62]
OVM		[158, 179]
Others	CIA	[199, 191, 193]
	ECA	[26]
	Order-Based GA	[160]
	SSP	[15, 327]

2.1 Bond Energy Algorithm (*BEA*)

BEA[233] is a general purpose clustering algorithm that can be applied to any non-negative array of numbers. It exploits the interrelationships (or bonds) between an element in the array and its four neighboring elements. These bonds create an energy which is defined as the sum of products of adjoining elements. For a particular row permutation(π) and column permutation(ρ), the total bond energy (TBE) is given by the following.

$$TBE(\pi, \rho) = \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n} a_{ij} * [a_{i,j-1} + a_{i,j+1} + a_{i-1,j} + a_{i+1,j}]$$

where

$$\begin{aligned} a_{0,j} &= a_{m+1,j} = a_{i,0} = a_{i,n+1} = 0 \\ m &= \text{the number of machines} \\ n &= \text{the number of parts} \end{aligned}$$

BEA seeks to maximize the *TBE* over all $n!m!$ permutations. McCormick et al. [233] noticed that since the vertical bonds are unaffected by rearrangement of columns and likewise for the horizontal bonds by rows, the problem is decomposed into two separate optimization problems. Arabie [9] termed a variation of *BEA* the *b*est insertion algorithm since the next column or row to be placed can be located at the beginning, end, or anywhere in between the previously positioned columns or rows. Another version, termed the *best* *neighbor algorithm*,[9, 83] efficiently locates the next column or row immediately following the previously positioned one. King [181] applied this algorithm to cell formation. Lenstra [212] showed that this problem is a traveling salesman problem (*TSP*), and Ng [257] used a graph theoretic approach (Christofides' algorithm) [77] to solve the *TSP* in order to establish a worst-case bound.Even though *BEA* produces tight clusters, they sometimes resemble a checker board rather than a diagonal. In order to ensure that *BEA* produces what Arabie termed a matrix in Robinson form, an alternative objective with a pairwise interchange of columns/rows must be employed.[9]

2.2 Rank Order Clustering (*ROC*)

ROC, as proposed by King,[180, 181] rearranges columns and rows using a binary representation to reduce the computational effort of *BEA*. With the iterative procedures of King,[182] Chan and Milner,[59] and Chu and Tsai,[83] exceptional elements (out-of-cell operations) are first identified and temporarily removed. Then the algorithm is applied to the reduced matrix in an attempt to produce the proper diagonal form. Bottleneck machines are duplicated for each part and the algorithm is reapplied to this new matrix. If any of the duplicated machines fall within the same cell, they are recomposed into a single machine.

Several limitations have been identified and explained by other researchers as well as by King himself. The quality of the result is dependent upon the initial machine/part incidence matrix.[59, 83, 371] Therefore, identification of exceptional elements and bottleneck machines is somewhat arbitrary.[62] Also, a binary representation restricts the size of the matrix to the internal word size of the computer .[83, 371] If exceptional elements exist, the influence is much greater in the higher order bits, which can lead to a non-block form. Even if natural diagonal clusters exist, *ROC* (or *ROC2* discussed below) may not be able to find them,[62] contrary to statements by King and Nakornchai.[182] A revised version, *ROC2*, was developed which overcomes the size limitation and increases the computational efficiency.[182] However, other drawbacks were not addressed.

2.2.1 Modified Rank Order Clustering (*MODROC*)

MODROC [62] helps to overcome some of the limitations of *ROC*. *ROC* tends to produce one cluster in the northwest corner, leaving the rest of the matrix relatively disorganized. *MODROC* identifies this block, removes the columns associated with the block, and then reapplies *ROC* to the sub-matrix. This technique generally produces a large number of clusters with a small number of parts/machines associated with them. In a subsequent stage, some of the clusters are combined using a hierarchical clustering method (described below) to form larger clusters.

2.3 Direct Clustering Algorithm (*DCA*)

DCA [59] was proposed to form tight groups along the diagonal of the machine/part matrix. It rearranges the matrix by moving the rows with the left-most positive cells to the top and the columns with the top-most positive cells to the left where a positive cell has $a_{ij=1}$. Identical outcomes result from any initial starting matrix, unlike *ROC*. *DCA* does not have any size limitation due to computer word length and converges in a relatively few iterations. Exceptional elements and bottleneck machines are removed from consideration by marking them and reapplying the algorithm.

There are inherent flaws with the statement of the algorithm and the implementation used in the paper. According to Wemmerlov,[370, 371] the proposed algorithm may not produce viable or acceptable solutions because it redirects the diagonal with each iteration. A modified version, $\overline{DCA}$ by Wemmerlov,[370] removes this flaw and can reproduce the examples in the original paper by Chan and Milner.[59] If one removes the initialization stage, *DCA* and $\overline{DCA}$ look very much like *ROC2*.[370, 371] Chu and Tsai[83] showed that even the modified version of $\overline{DCA}$ has difficulty producing natural diagonal blocks even when they exist in the input matrix. $\overline{DCA}$ has been shown to perform poorly when applied to large, real-world data sets as it tends to form one small group in the northwest corner and then a very large, sparse group containing the rest of the machines and parts.[370, 371]

2.4 Comparison of Array-Based Methods

Comparing each of the three array-based clustering techniques, *BEA*, *ROC* and *DCA*, Chu concluded that *BEA* significantly outperformed the other two in problems with and without exceptional elements and bottleneck machines.[83] The array-based clustering techniques used in the design of manufacturing cells are both efficient and simple to apply to the part/machine matrix. However, these algorithms generally do not take into account other types of manufacturing data such as cost of machines and maximum cell size, and they usually require visual inspection of the output to determine the composition of the manufacturing cells.

3 Hierarchical Clustering

Hierarchical clustering techniques operate on an input data set described in terms of a similarity or distance function and produce a hierarchy of clusters or partitions. At each similarity level in the hierarchy, there can be a different number of clusters with different numbers of members. Unlike the array-based techniques, hierarchical clustering methods do not form machine cells and part families simultaneously. These methods can be described as either divisive and agglomerative. Divisive algorithms start with all data (machines or parts) in a single group and create a series of partitions until each machine (part) is in a singleton cluster. Stanfel [332] is the only researcher to apply a divisive method to cellular manufacturing; therefore, attention is focused on agglomerative clustering algorithms that start with singleton clusters and proceed to merge them into larger partitions until a partition containing the whole set is obtained.

Hierarchical clustering methods involve a two-stage process that first calculates similarity coefficients between each pair of individuals (machines or parts). This can be represented as a lower triangular matrix since the similarity coefficient between individuals is commutative. The second stage of the process determines how the pairs with roughly equivalent similarity levels should be merged. The specific logic for each individual method is described below.

3.1 Choice of Similarity Measure

Because similarity coefficients can incorporate manufacturing data other than just the binary part/machine incidence matrix,[132, 305, 297] a variety of similarity measures have been defined. McAuley[232] used the generic Jaccard coefficient to form machine cells. Carrie,[57] who applied McAuley's work to several real problems, defined a similarity coefficient between pairs of parts to form part families first. There does not appear to be any inherent advantage to forming the part families or machine cells first.[371]

Gupta and Seifoddini's [132] similarity coefficient incorporates production requirements, the machine/part incidence matrix, the actual sequence of operations, the average production volume for each part, and the unit processing time for each of the part's operations. Seifoddini and Djassemi [303] modified the Jaccard similarity to take into account production volume. When compared with the Jaccard similarity, the production volume based similarity reduces the sum of intercellular and intracelluar movements as well as improves the scheduling process. Mosier [252, 251] proposed the additive similarity coefficient (*ASC*), a weighted adaptation of the Jaccard coefficient that incorporates the relative importance of each part, and the multiplicative similarity coefficient (*MSC*), which is approximately a correlation coefficient. De Witte [98] and Taboun et al. [343] compared a variety of different similarity measures and coefficients.

3.2 Hierarchical Clustering Algorithms

The first step in hierarchical clustering is to group the two individuals, i and j, with the highest level of similarity into one cluster, ij. The combined cluster behaves as if it is a single individual. The similarity between this cluster and individual k, as defined by the *SLC* [232] algorithm, is the maximum of the similarities between k and the component members of the cluster ij. Iterations continue to merge the groups with the largest similarity coefficient until a single group exists.

The most common way to display the hierarchy of clusters generated by the algorithm is in the form of a dendrogram. The cell designer must choose a similarity level or threshold in order to define the number of clusters. As the threshold increases, the number of cells increases while the size of each cell decreases. Seifoddini and Wolfe [306] selected a threshold that produces the minimum total material handling cost (intercellular plus intracellular). Hierarchical clustering algorithms do not cluster machines and parts simultaneously, so initially only cells or families are formed. The final step is to reapply hierarchical clustering or a secondary procedure, such as *ROC*, to allocate parts (machines) to the families (cells).

SLC has a severe chaining problem, which means that two clusters can be grouped based merely upon a single bond between one machine in each cluster.[132, 232, 249, 305, 295, 297]

The chaining problem can lead to improper machine assignment in the groups.[297, 317, 318] To help reduce the chaining problem, Seifoddini and Wolfe [295] applied the average linkage clustering (*ALC*) algorithm. The similarity between two clusters is defined as the average of the similarity coefficients for all of the members of the two clusters. A weighted average can also be employed.[132] Complete linkage clustering (*CLC*) further reduces the chaining problem by selecting the minimum similarity coefficient as the in-between cluster relationship instead of the maximum.[132, 249]

SLC, *ALC*, and *CLC* algorithms can deal with both similarity coefficients as well as Euclidean distances. The Centroid and the Ward method [348] deal with Euclidean distance measures only. Clusters are merged by selecting the minimum distance between clusters instead of the maximum similarity. Miyamoto [241] presented more efficient ways of updating the similarity or distance measures for *SLC*, *ALC*, *CLC*, Centroid, and Ward algorithms.

3.3 Comparison of Hierarchical Methods

Hierarchical clustering methods can be implemented easily and have an advantage relative to array-based clustering, i.e., they have the flexibility to incorporate manufacturing data other than the binary machine/part incidence matrix.[132, 305, 297] One disadvantage is that the designer must decide on an appropriate similarity level to select the groups. In small applications, this is not a problem since the designer can visually evaluate the dendrogram. However, as applications become too large for output in the form of a dendrogram, other means of storing the hierarchy must be employed, such as minimum spanning trees.[232] The duplication of bottleneck machines is not handled by most algorithms, although Seifoddini and Wolf [305] employed a strategy for this problem.

Mosier,[249] Shafer,[317, 318] and Vakharia and Wemmerlov[352] conducted an in-depth comparison of many different hierarchical clustering algorithms with different similarity and distance coefficients. Gupta compared four hierarchical clustering algorithms, *SLC*, *ALC*, *CLC*, and weighted average linkage clustering (*WALC*) and evaluated their performance with respect to their chaining effect. He concluded that the chaining problem is increasingly severe in order of *CLC*, *WALC* , *ALC*, and *SLC*.[132, 129] Seifoddini and Hsu [304] show that the weighted similarity coefficient produces better solutions based on the number of exceptional elements than the Jacard similarity and the commonality score. They show that grouping efficiency, grouping efficacy, and the grouping capability indices were not consistent performance evaluation measures (see Section 10 on performance measures). Table 2 lists the contributions in the area of hierarchical clustering methods.

4 Non-hierarchical Clustering

Non-hierarchical clustering techniques operate on an input data set by prespecifying the number of clusters to be formed using a similarity or distance function. These techniques produce a single partition of the data. The advantage of non- hierarchical clustering over

Table 2: Agglomerative Hierarchical Clustering Methods

Method	Approach	References
SLC	General	[32, 298, 304, 235, 376, 29, 264, 318, 323, 350]
	Jacard/Modified Jacard	[57, 232, 302, 304]
	ASC/WSC Measure	[252, 251, 304]
	Different Measures	[249, 317, 348, 350, 343]
	Different Process Plans	[323]
ALC	General	[303, 132, 131, 129, 297, 305, 295, 298, 299, 318, 350]
	Weighted ALC	[132, 132, 129]
CLC		[132, 131, 129, 130, 249, 318, 348, 350]
Median/Lance and Williams		[348]
Centroid		[249, 348, 350]
Wards		[249, 318, 348, 350]
Set Merging		[74, 76, 354, 350]
Selection of a Threshold		[306]

hierarchical clustering is that a similarity or distance matrix does not need to be computed or stored.[7] More natural clusters tend to be formed because data members are not permanently bound to a group in the early stages of clustering.[61] The obvious disadvantage is that the number of clusters must be specified *a* priori, potentially forcing some natural clusters to be merged or partitioned. However, the number of clusters can be changed and the data reprocessed in order to evaluate the sensitivity of the results. Table 3 summarizes the work in this area.

4.1 Ideal Seed Non-hierarchical Clustering (*ISNC*)

Chandrasekharan and Rajagopalan [61] applied a non-hierarchical technique (*ISNC*) using an evaluation criterion called "grouping efficiency," η , which measures intercell movement and within-cell machine utilization. To overcome the limitation of specifying the number of clusters, k, *a priori*, the problem is first formed as a bipartite graph. Then, a theoretical upper limit on the maximum number of independent part families or machine cells is developed. A modified MacQueen's k-means method was adopted.[229] The original MacQueen algorithm selects the first k data units or vectors as the initial seed points. The remaining data units are assigned to the cluster with the nearest centroid. After each assignment, the centroid is updated to include the current data unit. After all the data units have been assigned to a cluster, the existing cluster centroids are taken as fixed seeds and the algorithm

Table 3: Non-Hierarchical Clustering Methods

Method	Approach	References
Classical Approaches	General	[211]
	K-means/Revised K-means	[229, 63, 61, 124, 123]
	GRAPHICS	[331]
Non-Classical Approaches	Fuzzy C-Means	[80]
	Unsupervised Neural Networks	[173, 174, 172, 216, 215, 285, 90, 284, 287, 359, 55, 342]

reassigns all data units to the nearest seed points without any updating.[7] The number of natural clusters is more likely to be smaller than the initial upper limit. However, the original algorithm forces every cluster to have at least one member (the initial seed selection), which is not appropriate for this problem. The modified algorithm selects the last k data units as the initial seed points. By the time these units are assigned to clusters, the cluster centroids will have shifted considerably from the original values.[61]

4.2 Zero-One Data: *ISNC* (*ZODIAC*)

ZODIAC, developed by Chandrasekharan and Ragagopalan,[63] is a much improved and expanded version of *ISNC*.[61] The initial seed selections in the first stage can be arbitrary, artificial, representative, or natural and are no longer limited to an arbitrary choice like that of classical cluster analysis.[7] The evaluation criterion, η , was expanded by the introduction of a "limiting efficiency," η_O, or upper bound. Seed selection uses the statistical distribution of inter-point distances to ensure that all the seeds belong to different clusters. The authors suggest that the selection process could be based upon similarities rather than distances. Natural seeding tends to produce better groupings over the artificial-representative method.[63]

4.3 Grouping Using Assignment Method for Initial Seed Selection (*GRAFICS*)

Srinivasan and Narendran [331] showed that the initial seed selection of *ZODIAC* can still lead to a collapse of some beneficial clusters or numerous groups with singleton members. Even natural seeds can produce erroneous results by generating fewer seeds than is needed, thus reducing the machine utilization.[331] Also, the minimum rectilinear distance used as the basis for clustering does not truly represent the machine processing that is required by an individual part.

GRAFICS overcomes these limitations by generating initial seeds from an assignment problem, which maximizes the similarity between machines. Each of the sub-tours is identified and used as initial seeds in a non-hierarchical clustering algorithm using a maximum density rule as the clustering criterion.

5 Comparison of Non-hierarchical Methods

An extensive comparison using 38 data sets was made between *GRAFICS* [331] and *ZODIAC*.[63] *GRAFICS* outperformed *ZODIAC* in the areas of grouping efficiency, grouping efficacy [183], and computational requirements. *GRAFICS* performed better for matrices containing exceptional elements.[331] Miltenburg and Zhang,[236] in a comprehensive comparison of nine well-known algorithms including array-based and hierarchical clustering techniques, found that the non-hierarchical clustering method, *ISNC*, outperformed the other eight. This conclusion was based upon an evaluation using a primary measure, η_g (grouping measure) and two secondary measures, η_c (clustering measure), and *TBE* (bond energy). The measures η_c andη_g are discussed in Section 10.

6 Graph Theoretic Approaches

Graph theoretic approaches, listed in Table 4, structure the cell formation problem in the form of networks, bipartite graphs, etc. Rajagopalan and Batra were among the first to apply a purely graph theoretic approach to the cell formation problem in which the nodes represent the machines and the arcs indicate the similarity among the machines.[279] They employed a graph partitioning approach to form the machine cells by assembling cliques determined from the graph and point out that the minimum amount of intercell movement does not always reflect the true cost. For example, if the intercell movement occurs in the middle of the operational sequence, two (not one) intercell movements will be required. After the allocation has taken place, the intercell movement, along with the machine loads, can be used to assign duplicate machines to individual cells. De Witte [98] used this approach with different similarity coefficients to expand on earlier work by de Beer et al. [97] to design primary, secondary, and tertiary cells. Other approaches include the minimum spanning tree (MST) by Ng [260, 258] and a heuristic graph partitioning approach by Askin and Chiu.[13]

6.1 The Network Flow Approach

Vohra et al. [364] applied a network approach using a modified Gomory-Hu algorithm [119] to find the minimum intercellular interaction. Lee and Garcia-Diaz [206] represented the clustering problem as a capacitated circulation network that measures the functional similarity between machines. They employed the primal-dual algorithm developed by Bertsekas and Tseng [25] to determine a complete loop and several sub-loops representing the machine cells. Once the machine cells have been determined, other algorithms are needed to assign parts or part families to the various machine cells.

Table 4: Graph Theoretic Methods

Method	Approach	References
Graph Partitioning Algorithms		[13, 98, 279, 278]
Network Flow	Relaxation Method	[25, 206]
	Gomory-Hu	[364, 375]
Bipartite Graphs		[63, 61, 148, 184, 186, 357]
Minimum Spanning Tree		[3, 60, 67, 258, 260, 329]
SSP		[15, 327]
Other		[6, 10, 105, 156, 275]

The special properties of a network flow problem can be exploited to outperform mathematical programming approaches.[206] There are several specific advantages of using the network approach versus the p-median model described in Section 8. Natural clusters can be formed since there is no *a* priori specification of the required number of clusters, and the approach is more computationally and memory efficient.

6.2 Bipartite Graph

King and Nakornchai [182] suggested that cell formation could be represented as a bipartite graph by letting the parts and machines represent the two sets. An edge between the sets represents the processing of a part on a machine. Chandrasekharan and Rajagopalan demonstrated that the existence of independent machine cells and part families could be represented by the disjoint components of the bipartite graph. The authors then determined the maximum number of disjoint components (clusters) that can exist for any given bipartite graph.[61, 63]

7 Methods Based on Artificial Intelligence

Researchers have increasingly applied artificial intelligence (AI) techniques to the cellular manufacturing problem as shown in Table 5. Many of these methods use solution methodologies patterned after non-hierarchical clustering methods, array-based clustering methods, etc. However, their AI implementation offers advantages over traditional cell formation methods.

Table 5: Artificial Intelligence Methods

Method	Approach	Subcategory	References
Artificial Neural Networks	Supervised Learning	- Back Propagation	[170, 171, 245, 340] [243, 242, 244]
		- Hebbian Learning	[231]
	Unsupervised Learning	- ART	[69, 201, 91, 230, 173, 174, 172, 216, 215, 285, 90, 89, 284, 287, 359]
		- Fuzzy ART	[55, 342]
		- Other	[207, 286]
		- Competitive - Learning	[84, 243, 359]
		- Kohenen	[359]
Expert Systems/ Knowledge Base			[75, 138, 147, 189, 204, 190, 284]
Formal Logic/ Language Theory			[346, 374]
Fuzzy Logic			[379, 278, 80, 78, 55, 342]
Simulated Annealing			[209, 5, 36, 217, 321, 352, 361, 140]
Tabu Search			[149, 223, 224, 321, 352]
Genetic Algorithms	Order-Based		[177, 29, 28, 96, 160]
	Integer-Based		[339, 133, 162, 160, 161, 163, 362]

7.1 Artificial Neural Networks

Artificial neural networks have been applied successfully to many manufacturing areas.[380] Several researchers have applied a supervised learning approach to the classification and coding problem based on the back-propagation learning algorithm.[86, 171, 166, 242] This method can be also applied to a production- oriented method to determine the machine cells and part families. Unsupervised learning techniques are better suited for the general clustering problem. It is not necessary to specify *a* priori the number of clusters nor the representative members of these clusters. Once the part families and machine cells are determined, a supervised model can be trained to assign new parts to the existing cells.

Malave and Ramchandran [231] applied a modified version of the Hebbian learning rule to the cell formation problem, while others have applied other unsupervised neural learning algorithms such as competitive learning [84, 243, 359] and Kohonen nets.[359] Several researchers used the neural network classifier based on an unsupervised learning model by Carpenter-Grossberg [56] called adaptive resonance theory (*ART1*) and its variants.[69, 201, 91, 230, 173, 174, 172, 216, 215, 285, 90, 284, 287, 359] Unsupervised learning techniques such as *ART1* cluster the input vectors into separate groups based upon similarities.[174, 216, 285] Kaparthi and Suresh applied this technique to three data sets in the literature as well as several larger data sets. The artificial neural network technique executed quickly and obtained good clusters.[216] The real advantage is its ability to solve large data sets (10,000 parts and 100 machine types). *ART* and its variants can be classified as non-hierarchical clustering methods.

Another variant of the *ART* models, *Fuzzy-ART*, handles both analogue and binary-valued inputs while utilizing a new learning law.[55, 342] Burke and Kamal [55] compared *Fuzzy-ART* with *ART1*, *DCA*,[59] Hebbian Learning,[231] and a procedure by Ballakur and Steudel [22] and concluded that *Fuzzy-ART* was a viable algorithm that outperformed all the other algorithms. However, this comparison was based on very small data sets and did not test the robustness of each algorithm. Suresh and Kaparthi [341] tested *Fuzzy-ART* against *ART1*, *ART1/KS*,[174] *ROC2*,[182] and *DCA* [59] on large, imperfect data sets in a replicated set of experiments. *Fuzzy-ART* produced superior solutions in terms of the bond energy recovery ratio (BERR), where BERR is the ratio of the final bond energy to the initial bond energy. However, *ART1* and *ART1/KS* had faster execution times than *Fuzzy-ART*, which was faster than *ROC2* and *DCA*.

7.2 Fuzzy Logic

Most clustering methods assume "that part families are mutually exclusive and collectively exhaustive."[379, 80] While some parts definitely belong to certain part families, it is not always clear which family is appropriate.[80, 213, 378] Li and Ding [213] and Xu and Wang [378] applied fuzzy mathematics to this problem. Chu and Hayya [80] applied a fuzzy c-means clustering algorithm to production data. The fuzzy c-means clustering can be classified as a non-hierarchical method and suffers from the same problems associated with those methods. The number of part families, c, must be specified *a* priori. The authors stated that if c is underestimated, the result is far from optimal. Also, a poor stopping criterion leads to inferior clusters. However, the technique is unaffected by exceptional elements. The workload among machine cells can be balanced better by using a reallocation scheme that utilizes the degree of membership a part has in a particular family. Chu and Hayya compared the fuzzy approach to the optimal 0-1 integer programming model and an heuristic approach.[81] The fuzzy approach was clearly better than the integer programming (IP) approach in both execution time and the quality of the solution. It was not as efficient as the heuristic but provided more information than is available from a "crisp" definition of families and cells.

7.3 Syntactic Pattern Recognition

Wu et al. [374] applied a syntactic pattern recognition approach to forming the cellular manufacturing system. Utilizing analytic methods from formal language theory, complex patterns (routing sequences) are represented as strings of primitive characters (machine identifiers). The grammar of the language provides a set of rules for constructing complex sub-patterns and patterns out of the primitives (simple primitives or prime sub- patterns) and for identifying relations between these patterns. Given a set of complex patterns and rules, the recognition process, i.e., the assignment of new parts to cells, involves parsing the primitives. The authors note that "the similarity between manufacturing cells and grammars is immediately noticed by recognizing that each cell can speak a language (the family of components it can produce)." Advantages of syntactic pattern recognition include cell formation taking into account material flow patterns, operation precedence relations, and non-uniform importance of machines.[346]

7.4 Genetic Algorithms and Simulated Annealing

Genetic algorithms and simulated annealing are very efficient stochastic search algorithms that try to emulate natural phenoneoma. These algorithms have been used successfully to solve a wide range of optimization problems, especially combinatorial problems. Because of the NP-completeness of the grouping problem and existence of local minima, these stochastic search algorithms [184, 212, 361] offer promising solution techniques for large scale problems. Simulated annealing mimics the process of cooling a physical system slowly in order to reach a state of globally minimum potential energy.[209, 5, 36, 217, 321, 352, 361, 140] The stochastic nature of the algorithm allows it to escape local minimum, explore the state space, and find optimal or near- optimal solutions. Boctor [36] and Venugopal et. al. [361] used simulated annealing to solve integer programming formulations of the cell formation problem.

Genetic algorithms (*GA*s) mimic the evolutionary process by implementing a "survival of the fittest" strategy. *GA*s solve linear and nonlinear problems by exploring all regions of the state space and exponentially exploiting promising areas through mutation, crossover, and selection operations.[234] They have proven to be an effective and flexible optimization tool that can produce optimal or near-optimal solutions. Joines et al. [160, 161, 164] developed a genetic algorithm approach to solve integer programming formulations of the cell design problem, allowing multi-criteria objective functions and constraints on the number of permissible cells. The algorithm was tested on 17 data sets from the literature and was able to find as good solutions as, if not better than, those in the literature. Venugopal et al. [362] also used *GA*s to solve a multi-objective integer programming formulation of the cell formation problem.

These stochastic search techniques offer capabilities (missing in many of the more traditional methods) that can provide the basis for more practically useful cell formation algorithms. *GA*s do not make strong assumptions about the form of the objective function as do many other optimization techniques.[234] Also, the objective function is independent of the algorithm, i.e., the stochastic decision rules. The only objective function requirement is that it maps the solutions into a partially ordered set. This offers the flexibility to interchange various objective functions and to utilize multi-criteria objective functions. Convenient substitution of various evaluation functions allows the system designer to generate and review alternative cell designs quickly. Single-criteria objective functions limit a method's usefulness to that of assisting the cell designer rather than autonomously forming the system. To move toward a satisfactory algorithmic result, multiple criteria objective functions that include such things as setup time requirements, tooling and crewing requirements, alternative routings, cost of machines, intercell transfers, and reduced machine utilization are needed.

*GA*s also offer the ability to constrain the number of permissible cells or part families selectively. Most clustering algorithms cannot identify all naturally occurring clusters and find solutions with a constrained number of clusters. The cell designer, at least initially, might specify an unconstrained problem to identify the naturally occurring groups of parts and/or machines. Afterwards, practical limits on the number of cells arising from availability of floor space, maximum work team sizes, or excessive machine redundancy requirements can be imposed.

The ability to analyze the ordering of operations within routing sequences is important not just for material flow considerations, but also because cell throughputs are dependent upon setup times, which are usually sequence dependent. Joines [160] and Daskin [96] developed non- classical, array-based clustering techniques using order-based genetic algorithms. Order-based *GA*s have the potential for analyzing operation precedence relationships to further refine the cell design process.

Industrial data sets are often too large for visual methods to associate machine cells and part families effectively. *GA*s can form machine cells and part families simultaneously and avoid visual inspection of the data. Further exploitation of genetic algorithm capabilities makes practical solutions to industrial scale problems more realistic.

8 Mathematical Programming

Purcheck [274, 276] was among the first to apply linear programming techniques to the GT problem. As an optimization technique, the objective in cluster analysis is to maximize the total sum of similarities between each pair of individuals (machines or parts) or to minimize the distances between each pair. As stated by Kusiak,[200] the distance between any pair can be any symmetric function such that $d_i i = 0$, $d_{ij} = d_{ji}$, and $d_{iq} = d_{ip} + d_{pq}$. The Minkowksi, the weighted Minkowksi, and the Hamming distance measures are the most often used in connection with cell formation.[188, 200] Models developed with distance-based objective functions can easily be extended to similarities.

Mathematical programming approaches for the clustering problem are nonlinear or linear integer programming problems.[36, 188, 200] These approaches offer the distinct advantage of being able to incorporate ordered sequences of operations, alternative process plans, non-consecutive part operations on the same machine, setup and processing times, the use of multiple identical as well as outsourcing of parts. These formulations also suffer from three critical limitations. First, because of the resulting nonlinear form of the objective function, most approaches do not concurrently group machines into cells and parts into families.[36] Second, the number of machine cells must be specified *a* priori, affecting the grouping process and potentially obscuring natural cell formations in the data. Third, since the variables are constrained to integer values, most of these models are computationally intractable for realistically sized problems.[36, 206] Large scale problems typically require heuristic and approximate methods with Lagrangean relaxation and subgradient optimization having been proposed,[125, 126] as well as a variety of simulated annealing and genetic algorithm approaches. Table 6 summarizes the efforts of mathematical programming applied to the cell formation problem.

Table 6: Analytic Methods[307]

Method	Approach	References
Linear Programming	General	[78, 137, 256, 267, 274, 276, 275, 277]
	Dealing with exceptional elements	[312, 220]
Integer Programming	P-Median/Generalized P-Median	[114, 23, 106, 200, 188, 108]
	Alternative Process Plans	[18, 162, 188, 215, 280, 326]
	Budget and capacity of plans	[280, 72]
	Column Generation (Process plans)	[282]
	Comparison	[85]
	Cells and families	[363, 36, 127, 221, 222, 224, 280]
	Knap Snack Problem	[108]
	General	[87, 92, 95, 94, 125, 126, 127, 159, 175, 202, 214, 291, 338, 353, 352]
	NonLinear	[127, 222, 334, 292]
	Lagrangian Relaxations	[187, 205, 245, 255, 328, 358]
	Lot Splitting /Non-consective Operations	[224]
	Mixed	[16, 1, 134, 281, 344, 167]
	Goal programming	[121, 120, 122, 237, 290, 316, 318]
Assignment Problem		[363, 221, 233, 263, 324, 328, 330, 331]
Dynamic Programming		[186, 335, 357]
Eigenvectors		[356]

8.1 The p-median Model

A classical clustering model, the p-median model, is used to cluster n parts (machines) into p part families (machine cells). Constraints specify that each part can belong to only one part family and the required number of part families is p. A part can be only assigned to a part family that has been formed. Solutions obtained are optimal for a specified p, requiring that all values of p be evaluated to find the minimum objective function value.[330] The p-median model assumes that each part, i, has only one set of machining operations, i.e., one process plan. Kusiak [188, 200] relaxed this assumption in the generalized p-median model.

8.2 Assignment Problem

McCormick et al. formulated the clustering problem as a quadratic assignment problem. They chose to apply *BEA* to determine a good solution instead of optimally solving the assignment problem because of the computational inefficiencies.[233] Srinivasan and Narendran [331] solved a simple assignment problem as part of the non-hierarchical clustering technique, *GRAFICS*.

Shtub [324] proved that the general formulation of the GT problem (the p-median model) and the generalized GT problem (the generalized p-median model) are equivalent to the generalized assignment problem (*GAP*). He used a branch and bound algorithm by Ross and Soland [288] to solve the *GAP* where tasks are considered part processes and agents are the process families.

Srinivasan et al. [330] showed that the assignment model can overcome some of the limitations of the p- median model, i.e., the number of part families, p, is not specified *a* priori. A similarity measure between machines is maximized to determine closed loops that represent the groups. Their approach determines the machine cells by an assignment model. If the parts can be assigned to these machine cells such that they are disjoint, the algorithm stops. If this condition does not hold, another assignment problem is solved to determine the part families, which then are assigned to the machine cells.

8.3 Dynamic Programming

Steudel and Ballakur [335] developed a two-stage heuristic to solve the machine cell formation problem. The first stage uses a dynamic programming approach to determine a sequence or chain of machines, which maximizes a machine similarity. The second stage partitions the maximum machine chain into individual cells. A new similarity measure, Cell Bond Strength (*CBS*), was developed to overcome the underestimation of similarity inherent in the Jaccard coefficient. *CBS* incorporates the processing times of the parts.

8.4 Other Analytic Approaches

The optimization techniques discussed to this point are limited to the formation of part families or machine cells and require some other means to perform the assignment of machines to the cells. Boctor developed an analytic model that simultaneously clusters or assigns machines and parts to cells [36] while minimizing the number of exceptional elements. For problems of meaningful scale, this technique is inefficient due, in part, to the number of integrality constraints. However, the constraints for the parts can be relaxed to improve efficiency and not affect the solution adversely. Since the model has the unimodularity property and all right-hand side values are integers, all the basic solutions are integer. The number of parts is usually larger than the number of machines, which greatly increases the computational efficiency. Optimal solutions for large-scale problems were computationally intractable, leading Boctor to recommend a simulated annealing approach.

Gunasingh and Lashkari performed extensive research in the area of mathematical optimization applied to cell formation.[125, 126, 127] They developed several models that eliminate the assumption that each part operation is restricted to one machine, allowing more flexibility in forming cells and families. They also developed two 0-1 nonlinear integer models that simultaneously cluster parts and machines into families and cells, respectively. The first model groups the parts and machines by maximizing the compatibility between the parts and machines. The second model clusters the parts and machines into cells by minimizing the cost of duplicating the machines and the cost of intercell movements. The method of Glover and Woosley [118] was used to linearize the objective function of both formulations.

Kusiak's generalized p-median model [188, 200] considered alternative process plans to improve the quality of the cells but did not include the cost and capacity of machines. Choobineh's integer programming model [72] takes these into account but does not include different process plans explicitly. Rajamani et al. [280] developed three integer programming models that incorporate both budget and machine capacity, as well as alternative process plans. Logendran [220] developed a two-phase methodology to model the process of duplication of bottleneck machines that takes into account the sequence of operations and budgetary limitations. Then the model is solved using 0-1 integer programming.

8.5 Techniques Applied to Flexible Manufacturing Systems (FMS)

Ventura et al. [358] formulated the grouping of parts and tools in an FMS as a 0-1 integer programming model equivalent to a model by Kusiak et al..[188] An upper bound on this model was determined using a Lagrangian dual approach. Jain et al. [159] developed a 0-1 integer programming model with resource constraints, i.e., number of machines and the number of copies of tools, to minimize overall system cost. Mulvey and Crowder [255] and Kusiak [187] also employed a Lagrangian relaxation but only to the part family formation problem in FMS. Stecke [334] and Sankaran and Kasilingam [292] formulated the machine grouping problem as a nonlinear mixed integer problem, while Hwang [154] developed a mathematical model that does not require that a similarity criterion be maximized. Kumar et al. [184] developed a 0-1 quadratic program using a modified eigenvector approach to solve the problem.

9 Effective Heuristic Approaches

Other than the mathematical programming techniques, most cell formation methods are heuristics. However, those discussed so far have been placed in aggregate categories, e.g., array- based clustering, artificial intelligence techniques, etc., based on their general solution approach. This section explains an additional diverse set of heuristics as listed in Table 7.

9.1 Branch and Bound Based Algorithms

Branch and bound methods have been used to solve the integer programming models described earlier. The Cluster Identification Algorithm, *CIA*, is an efficient cell formation heuristic that works only for perfect data sets, i.e., data sets with no exceptional elements or bottleneck parts/machines. To overcome this limitation, Kusiak [192] developed three branch and bound schemes to be used in conjunction with *CIA*.[199] Kusiak [193] compared his procedure with nine other cell formation algorithms and concluded that the branch and bound approach produced better quality solutions.

Al-Qattan [4] formed machine cells and part families using a branch and bound method that uses network analysis by branching from a seed machine (the starting node) and bounding on a completed part, i.e., a part not requiring any more operations. It has been shown to outperform *ROC* [180, 181] and the hierarchical methods by Seifoddini and Wolfe.[305]

9.2 Multi-Objective Procedures

Practical cell formation objectives, such as minimizing machine duplication and crew and tooling requirements, must be balanced against conflicting objectives such as minimizing intercell transfers and maximizing machine utilization. Toward this goal, Wei and Gaither [366] extended Kumar and Vannelli's [186] single objective heuristic into a multi-objective heuristic. The heuristic averaged 96% of the optimal solution in minimizing bottleneck costs and intra/intercell load imbalances while maximizing the average cell utilization. Frazier et al. [112] developed an interactive,multi-objective cell formation heuristic. As an illustration, the authors chose to minimize the total cost of exceptional elements, the number of exceptional elements, and the utilization imbalance among the cells while maximizing the overall machine utilization across all cells.

9.3 Other Heuristics

*MA*chine-component *CE*ll formation (*MACE*) of Waghodekar and Sahu [365] groups machines into cells based upon a product similarity measure. It minimizes the number of exceptional elements that occur and outperforms the *ROC* algorithm.[180, 181] Co and Arrar [87] used a 0-1 integer programming model to maximize machine utilization and then apply a modified King's algorithm [180, 181] to cluster the machines. A direct-search algorithm is used to determine the number of cells and the composition of each cell. Chakravarty and Shtub [58] combined layout decisions with production scheduling decisions in the design process to minimize the setup and inventory carrying costs.

A simple two-part heuristic algorithm, which minimizes intercell movement for realistically dimensioned problems, was developed by Harhalakis et al.[139] This procedure takes into account the sequence of operations and number of non-consecutive operations on the same machine when minimizing intercell movement. Seifoddini [300] developed a probabilistic model to overcome assumptions of deterministic demand for parts. A variety of product mixes with different probabilities of occurrence was used to yield several different part/machine incidence matrices, which were then used as input to an existing grouping algorithm. An expected intercellular material handling cost was determined.

Kusiak and Chow [199] developed algorithms to solve standard and augmented formulations. The standard formulation uses the machine incidence matrix, while the augmented formulation associates each part with a cost and constrains the size of the cell. The approaches used, cluster identification algorithm ($O(2mn)$) and cost analysis algorithm ($O(2mn + nlog(n))$), are more efficient than *BEA*,[233] *ROC*,[180, 181] or p-median model.[200]

Wei and Kern [368, 369] used a similarity between machines score adapted from Kusiak[188] to aid in grouping. This algorithm can be tied loosely to the hierarchical methods.[73] Sundaram and Shong[338] used an integer programming model based on the hospitality and flexibility relationships advocated by Purcheck.[273] Other researchers have introduced labor resource allocation into the cell formation problem.[141, 151, 152, 237, 289]

Most cellular manufacturing techniques try to minimize intercell movement as represented by the number of exceptional elements. However, the number of exceptional elements may not reflect accurately the level of intercell movement required. To eliminate exceptional elements, the machine causing intercell movement can be duplicated or the parts routing sequence can be changed by subcontracting the part, redesigning the part, or using an alternative routing. Exceptional elements are interdependent because actions used to eliminate one element may affect other elements in the incidence matrix. Kern and Wei [178] presented a systematic approach for the elimination of exceptional elements.

Vannelli and Kumar [357] developed a methodology to minimize the number of bottleneck cells (cells containing a bottleneck part or machine) based on finding the minimal cut-nodes in either partition of a bipartite part-machine graph. These cells can be eliminated through duplication of the machines or subcontracting the parts. Two efficient algorithms by Kumar and Vannelli [186] were used to find the minimal number or minimal total cost of subcontracting parts that will produce disaggregated cells. Several researchers [200, 184] have used a linear transportation model to approximate this problem. Kumar and Vannelli [186] expanded earlier work to find the minimal number of bottleneck cells or bottleneck machines.

Okogbaa et al. [117] developed a versatile intercell flow reduction heuristic, which produces different alternatives for different designer input, e.g., number of cells, cell size restrictions, etc. It outperformed another reduction heuristic, *ICRMA*[345] and in terms of total intercell flow, outperformed *ROC* [180] and *WCU*.[22] Selvam and Balasubramanian [311] developed a similarity coefficient-based heuristic for cell formation which minimizes material handling cost plus machine idle time cost.

Logendran [218] developed an efficient heuristic that minimizes a weighted sum of both intercell and intracell movement. It also considers workload balance and determines machine utilization in the cell formation process. Stanfel [332] also included intracell movement, but it was not a true representation of the total intracell movement nor did it consider the workload on the machines. Askin and Subramanin [14] developed a cost-based heuristic approach to the cell formation problem. This procedure can be classified as a similarity coefficient-based method that takes into account the fixed machine costs, material handling costs, WIP inventory costs, production cycle inventory costs, variable production costs. and setup costs.

Minis et al. [238] developed a technique that groups production machines into cells and parts into families by minimizing the intercell traffic subject to capacity constraints. This method has the capability of including unique, as well as multiple-function, identical machines in the grouping procedure. Part setup and run times are used to evaluate capacity constraints using pallet traffic rather than part traffic in the minimization stage.

Vakharia and Wemmerlov [355] developed a similarity-based heuristic considering within-cell material flow. Boe and Cheng [37] used a closest-neighbor algorithm (*CNA*) in an extensive comparison against 10 other grouping algorithms on 11 different data sets from the literature. *CNA* was able to find the best results for all 11 data sets and, on average, it obtained the highest grouping efficiency while being the most efficient among the reliable methods (*ALC*, *BEA*, and *SSP*).

10 Cluster Evaluation

The designer of cellular manufacturing systems is faced with several decisions concerning a methodology for cell formation. These include: the algorithm(s) to employ; the criterion to use as the basis for clustering; and the policies used to handle exceptional elements and bottleneck machines. It is possible to utilize several techniques, compare solutions, and determine which one is most appropriate. However for problems of even moderate size, determination of algorithm performance becomes very difficult. A variety of performance measures have been proposed.[61, 63, 64, 183, 233, 258, 260, 308, 310] stated by Chu,[82, 79] the performance of cell formation algorithms can be based on their computational efficiency or their grouping effectiveness. According to Chu,[82, 79] Kusiak,[199] and Wei [368] computational efficiency of a method can be measured by the computational complexity, execution time, or memory storage requirements.

Computational issues aside, the determination of the grouping effectiveness measure is in itself a challenging task. Some measurement criterion is necessary to compare the clustering solution to the original data, a standard result, or solutions from other algorithms.[82, 79] This criterion can be an independent measure or an aggregate measure. Two of the most commonly used independent measures are the number of exceptional elements produced and the total bond energy (McCormick et al. [233]). Since many heuristics use an objective function based on costs, a natural aggregate measure can be based on the minimum cost.

Chandrasekharan and Rajagopalan [61, 63] developed grouping efficiency to measure the effectiveness of forming disjointed block diagonal submatrices. Grouping efficiency, η, is a weighted sum of intercell movements and within-group utilization (see below). A perfectly diagonal block solution with no voids in the blocks and no exceptional elements has an efficiency of 100%.

$$
\begin{aligned}
\eta &= q\eta_1 + (1-q)\eta_2 \qquad 0 \leq q \leq 1 \\
\eta_1 &= \frac{e_d}{D} \\
\eta_2 &= 1 - \frac{e_o}{mn - D} \\
e_d &= \text{the \# of elements along the diagonal blocks} \\
e_o &= \text{the \# of exceptional elements} \\
D &= \text{sum of the area covered by the diagonal blocks}
\end{aligned}
$$

Grouping efficiency has been used widely to determine cluster performance.[37, 61, 63, 82, 79, 83, 139, 260, 259, 258, 331] Chandrasekharan and Rajagopalan [63] showed that certain data sets could impose restrictions such that 100% efficiency is not possible. Therefore, the concept of a "limiting efficiency," a maximum attainable efficiency, and a "relative efficiency" (the ratio of grouping efficiency and limiting efficiency) was introduced. The maximum attainable grouping efficiency is a reflection of the mean and standard deviation of the Jaccard similarity coefficient.[64] If the standard deviation falls between 0.2 and 0.35, the problem is well structured for attaining block diagonalization; outside the range, the data is ill structured.

Chandrasekharan and Rajagopalan [61, 63] suggested giving equal weighting to intercell movement and machine utilization. However, Kumar and Chandrasekharan [183] observed in cases with more than two cells and large and/or sparse solution matrices, the machine utilization factor overshadows the intercell movement factor, making it virtually absent in the computation of the criterion.

There is evidence that large data matrices produce efficiency values close to one. Kumar and Chandrasekharan demonstrated this using the *ZODIAC* algorithm by solving 100 data sets that produced block diagonal solutions. Even in the worst cases where there was a large percentage of exceptional elements, the grouping efficiency never fell below 75%. "Grouping efficacy," Γ, which is unaffected by the size of the data set, was developed to overcome this limitation.[183]

$$
\Gamma = \frac{(1-\psi)}{(1+\phi)} = \frac{1 - \frac{e_o}{e}}{1 + \frac{e_v}{e}} = \frac{e - e_o}{e + e_v}
$$

where:

$$
\begin{aligned}
e &= \text{the \# of operations in the data matrix} \\
e_v &= \text{the \# of voids in the diagonal blocks} \\
e_d &= \text{the \# of elements along the diagonal blocks} \\
e_o &= \text{the \# of exceptional elements} \\
D &= \text{the area covered by the diagonal blocks}
\end{aligned}
$$

As was the case with grouping efficiency, two additional efficacy measures, "limiting efficacy" and "relative efficacy" were also developed.

Ng [259] experimentally showed that grouping efficiency is not entirely dependent on the mean and standard deviation but also on the size of the matrix and demonstrated that grouping efficiency and efficacy are not appropriate for the cell formation problem.[257, 259, 258, 260] Consider the following two partial derivatives of grouping efficiency, η:$\frac{\partial \eta}{\partial e_d} = \frac{q}{D}$ reflects the rate of change of grouping efficiency with respect to the nonzero elements in the diagonal blocks (e_d), while $\frac{\partial \eta}{\partial e_o} = \frac{q-1}{mn-D}$ reflects the rate of change of grouping efficiency with respect to the exceptional elements (e_o). In many situations, the designer would want the effect of exceptional elements on η to be much larger than the effect of nonzero entries in the diagonal blocks. In this case, the cost of an intercell movement would be greater than the cost of a slight reduction in machine utilization within the cell. Therefore, Ng [259, 258, 260] developed a weighted grouping efficacy, γ, measure that correctly addresses this issue.

$$
\gamma = \frac{r(1-\psi)}{\psi + r(1+\phi-\psi)} = \frac{q(e-e_o)}{q(e+e_v-e_o)+(1-q)e_o} = 1 - \frac{qe_v+(1-q)e_o}{qD+(1-q)e_o}
$$

where:

$$
\begin{aligned}
r &= \frac{q}{1-q}, \qquad 0 \leq q \leq 1 \\
q &= \text{the weight associated with the voids in the diagonal blocks} \\
(1-q) &= \text{the weight associated with the exceptional elements}
\end{aligned}
$$

In comparing 11 data sets from the literature and 10 randomly generated problems, solutions obtained using the weighted grouping efficacy had a smaller percentage of exceptional elements than grouping efficacy and grouping efficiency. While the differences between the two efficacies are minor for well-structured matrices, the differences are significant when they are ill-structured.[260]

In Chu and Tsai's [83] comparison of three array- based clustering methods, they chose four measures of performance: total bond energy, percentage of exceptional elements, machine utilization, and grouping efficiency. Kusiak,[193] in a comparison of ten different algorithms, found that *BEA*, *SSP*, *ROC*, and *DCA* did not always produce block diagonal structures. *CIA* also could not solve all the problems correctly since it was designed to solve problems without bottleneck machines and exceptional elements. Of the techniques that solved all the problems, the branch and bound scheme by Kusiak gave, on average, the best quality solutions, outperforming the *WCU* algorithm of Ballakur and Steudal.[22] However, the *SLC*, *ALC* and *ZODIAC* algorithms produced the best results for at least one of the problems.

Miltenburg and Zhang [236] stated that the objectives of any cell formation algorithm should be to maximize the machine utilization while minimizing the number of exceptional elements. They chose to use one primary measure and two secondary measures to evaluate the effectiveness of nine different algorithms. The primary measure was

$$\eta_g \;=\; \eta_u - \eta_m, \qquad -1 \leq \eta_g \leq 1$$

where:

$$\begin{aligned}
\eta_u &= \frac{e_d}{D} \text{ measure of machine utilization} \\
\eta_m &= \frac{e_o}{e} \text{ measure of part movement between groups}
\end{aligned}$$

Notice that eta_u is the first term, η_1, from grouping efficiency[61, 63] while eta_m is a term from grouping efficacy. Miltenburg and Zhang stated that they chose to use eta_m over the η_2 term from grouping efficiency for three reasons: (1) η_2 does not satisfy $0 \leq \eta_2 \leq 1$; (2) $\eta_2 \neq 0$ when there are no exceptional elements; and (3) η_2 is more complex than eta_m. However, their comparison of eta_m with η_2 was invalid. The term, eta_m, is a reflection of the number of exceptional elements while η_2 is a reflection of the number of void elements outside the cluster groups. As the number of exceptional elements decreases, eta_m decreases while η_2 increases. Therefore, the more appropriate comparison would have been $1 - \eta_2$ with eta_m. They also used two secondary measures to aid in the comparison of the algorithms: the ability to produce tight clusters around the diagonal and the total bond energy.

In Boe and Cheng's[37] comparison of 11 different algorithms, they chose to use grouping efficiency and the minimum number of exceptional elements for comparison. The algorithms were ranked on their ability to give the best result for 11 data sets. The computational efficiency of each algorithm was also compared in terms of the average execution time. The interested reader is referred to Chu,[79, 82] Selim et al.,[309] and Offodille[265] for comprehensive review of various evaluation measures.

11 Conclusions and Recommendations for Further Study

A comprehensive overview of the production-oriented cell formation literature has been presented. Although much overlap naturally occurs, major methodological categories have been identified, partitioning the paper into discussions of array- based methods, hierarchical clustering techniques, non- hierarchical clustering, graph-theoretic approaches, methods based on artificial intelligence, mathematical programming models, and various heuristics. Within each category, breadth of coverage has been the target, reserving detailed discussion for the most significant approaches. The results of several comparative studies of cell formation techniques are provided to reinforce the relative strengths/weaknesses of various methods. Finally, a number of useful evaluation measures employed in cell formation problems are discussed in terms of their practical implications on cell configuration.

Clearly, the literature is rich with a large and diverse set of clustering methodologies for cell design and part family identification. It is also clear that no methodology addresses all of the issues needed to solve large-scale industrial applications. Burbidge's recent caution that many of the papers "seem to have lost touch with the basic need to design methods that can be used in industry" [52] is noteworthy. Further research in the area should focus on avoiding incremental improvements in favor of developing innovative approaches that meet the test of industrial application.

To stimulate this activity, attention has been given to some of the newer AI techniques such as artificial neural networks, simulated annealing and genetic algorithms. Inherent advantages of these stochastic search techniques include the ability to: (1) employ multi-criteria objective functions; (2) conveniently, and interchangeably, utilize several non-linear evaluation measures; (3) selectively include or exclude constraints on the number of part families/machine cells, and (4) simultaneously form part families and machine cells without visual inspection of the output.

Table 7: Other Heuristics

Method	Approach	References
Branch and Bound	General	[176, 328]
	Alternative Process Plans	[198, 262]
	CIA	[71, 196, 192, 195]
	A*	[195, 68]
	Network Analysis	[4]
Multi-objective		[209, 28, 286, 230, 92, 112, 111, 121, 120, 122, 136, 149, 237, 290, 326, 366]
Cost based heuristics		[2, 14, 58, 311] [72]
Other	General	[367, 66, 87, 81, 100, 134, 142, 139, 225, 228, 235, 254, 301, 333, 347, 228]
	Mulitple, Functionally Identical Machines	[363, 224, 286, 238, 377]
	Covering Problem	[20, 218]
	Intracell and intercell moves along with workload balancing	[89, 218, 224, 219, 226]
	Virtual Cell layout/Intercell and Intracell Layout Included	[16, 293, 58, 143, 5, 11, 156, 157, 65, 215, 320]
	Alternative Routings	[18, 31, 198, 262, 326, 323, 223, 162, 188, 215, 280, 326, 130, 123, 169, 256, 337, 323]
	Capacity Constraints	[286, 377, 142, 366, 134, 256]
	Cell Similarity Coefficient Alg.	[227]
	Closet Neighbor Algorithm	[37]
	Moments-based	[253]
	Hospital. and flexibil. relation.	[273]
	Inter-cell flow reduction	[117, 345]
	Linear Cell Clustering Alg.	[73, 368, 369, 304]
	MACE	[365]
	Machine and Human cells	[319, 141, 151, 152, 237, 289]
	Machine and Robotic cells	[200, 264]
	Material Flow with Constraints	[355]
	Probabilistic demands	[300, 137]
	Eliminate Exceptional Elements	[178, 312]
	Duplication of bottleneck mach.	[356, 312, 148, 220, 293, 296, 360]
	Subcontracting parts	[357, 356, 312, 148, 185, 186, 360]
	Principal Component Analysis	[12]
	Design Constraints	[286, 142, 144, 145, 320]
	AGV considerations	[106, 208]
	Within Cell Utilization	[22]

References

[1] G.K. Adil, D. Ragamani, and D. Strong. A mathematical model for cell formation considering investment and operational costs. *EJOR*, 69:330–341, 1993.

[2] M.N. Ahmed, N. Ahmed, and U. Nandkeolyar. A volume and material handling cost based heuristic for designing group technology cells. *Journal of Operations Management*, 10:488–511, 1991.

[3] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network Flows: Theory, Algorithms and Applications*. Prentice-Hall, New York, 1992.

[4] I. Al-Qattan. Designing flexible manufacturing cells using branch and bound method. *IJPR*, 28(2):325–336, 1990.

[5] A. Alfa, M. Chen, and S. Heragu. Integrating the grouping and layout problems in cellular manufacturing systems. *Computers Industrial Engineering*, 23(1-4):55–58, 1992.

[6] F. Amirahmadi and F. Choobineh. A heuristic grouping procedure for component family formation in a cellular manufacturing manufacturing environment. *Computers Industrial Engineering*, 23(1-4):69–72, 1992.

[7] M. R. Anderberg. *Cluster Analysis for Applications*. Academic Press, New York, 1973.

[8] C.L. Ang and P. Willey. A comparative study of the performance of pure and hybrid group manufacturing systems using computer simulation techniques. *IJPR*, 22(2):192–233, 1984.

[9] P. Arabie and L.J. Hubert. The bond energy algorithm revisted. *IEEE Transactions on Systems, Man, and Cybernetics*, 20(1):268–274, 1990.

[10] B. Arvindh, C. Y. Chen, and S. A. Irani. Travel chart clustering techniques for cell formation. Technical Report Working Paper, 1994.

[11] B. Arvindh and S. A. Irani. Cell formation: The need for integrated solution of the subproblems. *IJPR*, 32(5):1197–1218, 1994.

[12] B. Arvindh and S. A. Irani. Principal component analysis for evaluating the feasibility of cellular manufacturing without initial machine-part matrix clustering. *IJPR*, 32(8):1909–1938, 1994.

[13] R. Askin and K. Chiu. A graph partitioning procedure for machine assignment and cell formation. *IJPR*, 28(8):1555–1572, 1990.

[14] R. Askin and S. Subramanian. A cost-based heuristic for group technology configuration. *IJPR*, 25(1):101–113, 1987.

[15] R.G. Askin, S.H. Cresswell, J.B. Goldberg, and A.J. Vakharia. A hamiltonian path approach to reordering the part-machine matrix for cellular manufacturing. *IJPR*, 29(6):1081–1100, 1991.

[16] R.G. Askin and M.G. Mitwasi. Integrating facility layout with process selection and capacity planning. *EJOR*, 57:162–173, 1992.

[17] R.G. Askin and A.J. Vakharia. *Group Technology - Cell Formation and Operation*, pages 317–366. 1990.

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

[19] A.A.S. Awwal and M.A. Karim. Machine parts rechognition using a trinary associative memory. *Optical Engineering*, 28(5):537–543, 1989.

[20] K.N. Balasubramanian and R. Panneerselvam. Covering technique - based algorithm for machine grouping to form manufacturing cells. *IJPR*, 31(6):1479–1504, 1993.

[21] A. Ballakur. *An Investigation of Part Family/Machine Group Formation in Designing a Cellular Manufacturing System*. Phd dissertation, University of Wisconsin-Madison, Madison, WI, 1985.

[22] A. Ballakur and H.J. Steudel. A within-cell utilization based heuristic for designing cellular manufacturing systems. *IJPR*, 25(5):639–665, 1987.

[23] D. Ben-Arieh and P.T. Chang. An extension to the p-median group technology algorithm-median group technology algorithm. *Computers and Operations Research*, 21(2):119–125, 1994.

[24] D. Ben-Arieh and E. Triantaphyllou. Quantifying data for group technology with weighted fuzzy features. *IJPR*, 30(6):1285–1299, 1992.

[25] D. Bertsekas and P. Tseng. Relaxation methods for minimum cost ordinary and generalized network flow problems. *Operations Research*, 36(1):93–114, 1988.

[26] M.V. Bhat and A. Haupt. An efficient clustering algorithm. *IEEE Transactions on Systems, Man, and Cybernetics*, 6(1):61–64, 1976.

[27] W. Biles, A.S. Elmaghraby, and I. Zahran. A simulation study of hierarchical clustering techniques for the design of cellular manufacturing systems. *Computers Industrial Engineering*, 21(1-4):267–271, 1991.

[28] R. Billo, D. Tate, and B. Bidanda. A genetic cluster algorithm for the machine-component groupng problem. Technical Report A working paper, University of Pittsburgh, 1995.

[29] R. E. Billo, D. Tate, and B. Bidanda. Comparison of a genetic algorithm and cluster analysis for the cell formation problem: A case study. In *3rd Industrial Engineering Research Conference*, pages 543–548, Atlanta, GA, 1994.

[30] R.E. Billo and B. Bidanda. Representing group technology classification and coding techniques with object oriented modeling principles. *IIE Transactions*, to appear, 1995.

[31] R.E. Billo, B. Bidanda, and P. Kharbanda. Re-engineering process plans for effective manufacturing. *International Journal of Manufacturing Systems Design*, to appear, 1995.

[32] R.E. Billo, F. Dearborn, C.J. Hostick, G.E. Spanner, E.J. Sthalman, and AUrand S.S. A group technology model to assess consolidation and reconfiguration of multiple industrial operations-a shared manufacturing solution. *International Journal of Computer Integrated Manufacturing*, 6(5):311–322, 1993.

[33] R.E. Billo, R. Rucker, and D.L. Shunk. Integration of a group technology classification coding system with an engineering database. *JMS*, 6(1):37–45, 1987.

[34] R.E. Billo, R. Rucker, and D.L. Shunk. Enhancing group technology modeling with database abstractions. *JMS*, 7(2):95–106, 1988.

[35] JT Black. Cellular manufacturing systems reduce setup time, make small lot production economical. *Industrial Engineering*, 29(10):36–48, 1983.

[36] F. Boctor. A linear formulation of the machine-part cell formation problem. *IJPR*, 29(2):343–356, 1991.

[37] W. Boe and C.H. Cheng. A close neighbor algorithm for designing cellular manufacturing systems. *IJPR*, 29(10):2097–2116, 1991.

[38] J.L. Burbidge. Production flow analysis. *Production Engineer*, 42:742–752, 1963.

[39] J.L. Burbidge. An introduction of group technology. In *Seminar on Group Technology*, Turin, 1969.

[40] J.L. Burbidge. Production flow analysis. *Production Engineer*, 50(4-5):139–152, 1971.

[41] J.L. Burbidge. Production flow analysis on the computer. In *Third Annual Conference of the Institute of Production Engineers*, 1973.

[42] J.L. Burbidge. *The Introduction of Group Technology*. Halster Press and John Wiley, New York, 1975.

[43] J.L. Burbidge. Manual method of production flow analysis. *Production Engineer*, 56(10):34–43, 1977.

[44] J.L. Burbidge. *Group Technology in the Engineering Industry*. Mechanical Engineering Publications Ltd., London, 1979.

[45] J.L. Burbidge. The simplification of material flow systems. *IJPR*, 20:339, 1982.

[46] J.L. Burbidge. Gt in yugoslavia. *Production Engineer*, 66:11–19, 1987.

[47] J.L. Burbidge. Group technology the state of the art. In *4th NCPR Sheffild City Poly*, pages 412–424, 1988.

[48] J.L. Burbidge. Group technology. In *IMC6 Trinity College*, pages 450–471, Dublin, 1989.

[49] J.L. Burbidge. *Production Flow Analysis for Planning Group Technology.* Oxford University Press, Oxford, 1989.

[50] J.L. Burbidge. Production flow analysis for planning group technology. *Journal of Operations Management*, 10(1):5–27, 1991.

[51] J.L. Burbidge. Change to group technology: process organization is obsolete. *IJPR*, 30(5):1209–1219, 1992.

[52] J.L. Burbidge. Comment on clustering methods for finding gt groups and families. *JMS*, 12(5):428–429, 1993.

[53] J.L. Burbidge. A reply to 'a note on a "change to group technology"'. *IJPR*, 31(4):1001–1002, 1993.

[54] A.G. Burgess, I. Morgan, and T.E. Vollmann. Cellular manufacturing: its impact on the total factory. *IJPR*, 31(9):2059–2077, 1993.

[55] L. I. Burke and S. Kamal. Fuzzy art and cellular manufacturing. In *Artificial Neural Networks in Engineering*, pages 779–784, St Louis, 1992.

[56] G.A. Carpenter and S. Grossberg. A massively parallel architecture for a self-orgainzing nerual pattern recognition machine. *Computer Vision, Graphics and Image Processing*, 37:54–115, 1987.

[57] A.S. Carrie. Numerical taxonomy applied to group technology and plant layout. *IJPR*, 11(4):399–416, 1973.

[58] A.M. Chakravarty and A. Shtub. An integrated layout for group technology with in-process inventory costs. *IJPR*, 22(3):431–442, 1984.

[59] H.M. Chan and D.A. Milner. Direct clustering algorithm for group formation in cellular manufacture. *JMS*, 1(1):65–74, 1982.

[60] C. Chandra, S.A. Irani, and S.R. Arora. Clustering effectiveness of permutation generation heuristics for machine-part matrix clustering. *JMS*, 12(5):388–407, 1993.

[61] M.P. Chandrasekharan and R. Rajagopalan. An ideal seed non-hierarchical clustering algorithm for cellular manufacturing. *IJPR*, 24(2):451–464, 1986.

[62] M.P. Chandrasekharan and R. Rajagopalan. Modroc: an extension of rank order clustering for group technology. *IJPR*, 24(5):1221–1233, 1986.

[63] M.P. Chandrasekharan and R. Rajagopalan. Zodiac-an algorithm for concurrent formation of part families and machine cells. *IJPR*, 25(6):835–850, 1987.

[64] M.P. Chandrasekharan and R. Rajagopalan. Groupability: Analysis of the properties of binary data matrices for group technology. *IJPR*, 27(6):1035–1052, 1989.

[65] M.P. Chandrasekharan and R. Rajagopalan. A multidimensional scaling algorithm for group layout in cellular manufacturing. *International Journal of Prduction Economics*, 32:65–76, 1993.

[66] O.E. Charles-Owaba and B.K. Lambert. Sequence dependent machine setup times and similarity of parts: A mathematical model. *IIE Transactions*, pages 12–21, 1988.

[67] C.Y. Chen and S.A. Irani. A cluster first-sequence last heuristic for machine-part grouping. *IJPR*, 31(11):2623–2647, 1993.

[68] H. G. Chen and H. H. Guerrero. A general search algorithm for cell formation in group technology. *International Journal of Production Research*, 32(11):2711–2724, 1994.

[69] S.-J. Chen and C.-S. Cheng. A neural network-based cell formation algorithm in celullar manufacturing. *International Journal of Production Research*, 33(2):293–318, 1995.

[70] C.H. Cheng. Algorithms for grouping machine groups in group technology. *International Journal of Management Science*, 20:493–501, 1992.

[71] C.H. Cheng. A branch and bound clustering algorithm. *IEEE Tranactions on Systems, Man, and Cybernetics*, 25(5):895–898, 1995.

[72] F. Choobineh. A framework for the design of cellular manufacturing systems. *IJPR*, 26(7):1161–1172, 1988.

[73] W.S. Chow. Discussion: A note on a linear cell clustering algorithm. *IJPR*, 29(1):215–216, 1991.

[74] W.S. Chow and O. Hawaleshka. An efficient algorithm for solving the machine chaining problem in cellular manufacturing. *Computers in Industrial Engineering*, 20(1):95–100, 1992.

[75] W.S. Chow and O. Hawaleshka. A novel machine grouping and knowledge-based approach for cellular manufacturing. *EJOR*, 69:357–372, 1993.

[76] W.S. Chow and O. Hawaleshka. Minimizing intercellular part movements in manufacturing cell formation. *IJPR*, 31(9):2161–2170, 1994.

[77] N. Christofides. Worst case analysis of a new heuritc for the traveling salesman problem. Technical Report Report 388, Graduate School of Industrial Administration, Cranegie-Mellon University, 1976.

[78] C. Chu, C. Tsai, and T. Barta. Fuzzy linear programming approach to the cell formation problem. In *Third IEEE International Conference on Fuzzy Systems*, pages 1406–1411, Orlando, FL, 1994.

[79] C.-H. Chu. Cluster analysis in manufacturing cellular formation. *International Journal of Management Science*, 17(3):289–295, 1989.

[80] C.-H. Chu and J.C. Hayya. A fuzzy clustering approach to manufacturing cell formation. *IJPR*, 29(7):1475–1487, 1991.

[81] C.-H. Chu and W. Lee. An efficient heuristic for grouping part families. In *Midwest Decision Sciences Conference*, pages 62–64, Georgia, 1990.

[82] C.-H. Chu and P. Pan. The use of clustering techniques in manufacturing cellular formation. In *International Industrial Engineering Conference*, pages 495–500, Orlando, Florida, 1988.

[83] C.-H. Chu and M. Tsai. A comparison of three array-based clustering techniques for manufacturing cell formation. *IJPR*, 28(8):1417–1433, 1990.

[84] C.H. Chu. Manufacturing cell formation by competitive learning. *IJPR*, 31(4):829–843, 1993.

[85] C.H. Chu and W. Lee. A comparison of 0-1 integer programming models for manufacturing cell formation. In *National Decision Sciences Institute Conference*, pages 181–184, New Orleans, 1989.

[86] Y. Chung and A. Kusiak. Grouping parts with a neural network. *JMS*, 13(4):262–275, 1994.

[87] H.C. Co and A. Arrar. Configuring cellular manufacturing systems. *IJPR*, 26(9):1511–1522, 1988.

[88] J.R. Crookall and K.I. Baldwin. An investigation into application of grouping principles and cellular manufacturing using monte carlo simulation. *CIRP*, 1(3), 1972.

[89] C. Dagli and R. Huggahali. Machine-part family formation with the adaptive resonance theory paradigm. *International Journal of Production Research*, 33(4):893–913, 1995.

[90] C. Dagli and R. Huggahalli. *Neural network approach to group tecnology*, pages 213–228. Elsvier, New York, 1991.

[91] C. Dagli and C.-F. Sen. *ART1 Neural Netowrk Approach to Large Scale Group Technology Problems*, volume 4, pages 787–792. ASME Press, New York, 1992.

[92] N.E. Dahel and S.B. Smith. Designing flexibility into cellular manufacturing systems. *IJPR*, 31(4):933–945, 1993.

[93] B. G. Dale and F. Dewhurst. Simulation of a group technology product cell. *Engineering Costs and Production Economics*, 8(1):45–54, 1984.

[94] V. Damodarn, R. S. Lashkari, and N. Singh. A production planning model for cellular manufacturing systems with refixturing consideration. *IJPR*, 30(7):1603–1615, 1992.

[95] V. Damodarn, N. Singh, and R. S. Lashkari. Design of cellular manufacturing systems with refixturing and material handling considerations. *Applied Stochastic Models and Data Analysis*, 9(2):97–109, 1993.

[96] M.S. Daskin. An overview of recent research on assigning products to groups for group technology production problems. Technical Report Working Paper, Northwestern University, 1991.

[97] C. De Beer, R. Van Gerwen, and J. De Witte. Analysis of engineering production systems as a base for product-oriented reconstruction. *CIRP*, 25:439–441, 1976.

[98] J. De Witte. The use of similarity coefficients in production flow analysis. *IJPR*, 18(4):502–514, 1980.

[99] J. Dekieva and D. Menart. Extensions of production flow analysis. *JMS*, 6(2):93–104, 1986.

[100] A. G. Del Valle, S. Balarezo, and J. Tejero. A heuristic workload-based model to form cells by minimizing intercellular movements. *International Journal of Production Research*, 32(10):2275–2283, 1994.

[101] M. B. Durmusoglu. Analysis of the converison from a job shop system to a cellular manufacturing system. *International Journal of Production Economics*, 30-31:427–436, 1993.

[102] S.P. Dutta, R.S. Lashkari, G. Nadoli, and T. Ravi. A heuristic procedure for determining manufacturing families from design-based grouping for flexible manufacturing systems. *Computers Industrial Engineering*, 10(3):193–201, 1986.

[103] I. El-Essay and J. Torrance. Component flow analysis-an effective approach to productions systems' design. *Production Engineer*, 51(5):165–170, 1972.

[104] H.A. El Maraghy and P. Gu. Feature based expert parts assignment in cellular manufacturing. *JMS*, 8(2):139–152, 1989.

[105] Z. Faber and M.W. Carter. *A New Graph Theory Approach for Forming Machine Cells in Cellular Production Systems*, pages 301–318. North-Holland, New York, 1986.

[106] M. Faraji and R. Batta. Forming cells to elminate vehicle interference and system locking in an agvs. *International Journal of Production Research*, 32(9):2219–2241, 1994.

[107] G. M. Fazakerley. A research report on the human aspects of group technology and cellular manufacture. *International Journal of Production Research*, 14(1):123–134, 1976.

[108] J.F. Ferreira Ribeiro and B. Pradin. A methodology for cellular manufacturing design. *IJPR*, 31(1):235–250, 1993.

[109] B. Flynn and F. Jacobs. A simulation comparison of group technology with traditional job shop manufacturing. *IJPR*, 24(5):1171–1192, 1986.

[110] B.B. Flynn and F.R. Jacobs. An experimental comparison of cellular (group technology) layout with process layout. 18:562–581, 1987.

[111] G.V. Frazier and N. Gaither. Seed selection procedures for cell formation heuristics. *IJPR*, 29(11):2227–2237, 1991.

[112] G.V. Frazier, N. Gaither, and D. Olson. A procedure for dealing with multiple objectives in cell formation decisions. *Journal of Operations Management*, 9(4):465–480, 1990.

[113] N. Gaither, G. Frazier, and J. Wei. From job shops to manufacturing cells. *Production and Inventory Management Journal*, Fourth Quarter:33–36, 1990.

[114] M.V. Ganesh and G. Srinivasan. Heuristic algorithm for the cell formation problem. *Computers Industrial Engineering*, 26(1):193–201, 1994.

[115] O. Garza and T. L. Smunt. Reducing flow between manufacturing cells: a sensitivity analysis. *International Journal of Production Research*, 32(9):2131–2147, 1994.

[116] O. Garza and T.L. Smunt. Countering the negative impact of intercell flow in cellular manufacturing. *Journal of Operations Mangement*, 10(1):92–118, 1991.

[117] O.O. Geoffrey, M. Chen, C. Chanagchit, and L.S. Richard. Manufacturing system cell formation and evaluation using a new inter-call reduction heuristic. *IJPR*, 30(5):1101–1118, 1992.

[118] F. Glover and E. Woolsey. Further reduction of zero-one polynomial programming problems to 0-1 linear program. *Operations Research*, 21(1):156–161, 1973.

[119] R.E. Gomory and T. C. Hu. Multi-terminal network flows. *SIAM Journal of Applied Mathematics*, 9:551–571, 1971.

[120] T.A. Gongaware and I. Ham. Cluster analysis application for group technology manufacturing systems. *Manufacturing Engineer Transactions*, pages 503–508, 1981.

[121] T.A. Gongaware and I. Ham. Cluster analysis applications for group technology manufacturing systems. In *SME 9th North American Metalworking Research Conference*, pages 503–508, Dearborn, Michigan, U.S.A., 1981.

[122] T.A. Gongaware and I. Ham. *Cluster analysis application for group technology manufacturing systems*, pages 131–136. 1984.

[123] P. Gu. Process-based machine grouping for cellular manufacturing systems. *Computers in Industry*, 17(1):9–17, 1991.

[124] P. Gu and H. A. El Maraghy. Formation of manufaturing cells by cluster-seeking algorithms. *Journal of Mechinical Working Technology*, 20:403–413, 1989.

[125] K. Gunasingh and R. Lashkari. The cell formation problem in cellular manufacturing systems- a sequential modeling approach. *Computers Industrial Engineering*, 16(4):469–476, 1989.

[126] K. Gunasingh and R. Lashkari. Machine grouping problem in cellular manufacturing systems- an integer programming approach. *IJPR*, 27(9):1465–1473, 1989.

[127] K. Gunasingh and R. Lashkari. Simultaneous grouping of parts and machines in cellular manufacturing systems - an integer programming approach. *Computers Industrial Engineering*, 20(1):111–117, 1990.

[128] R.M. Gupta and J.A. Tompkins. An examination of the dynamic behavior of the part-familes in group technology. *International Journal of Production Research*, 20(1):73–86, 1982.

[129] T. Gupta. Clustering algorithms for the design of a cellular manufacturing system-an analysis of their performance. *Computers Industrial Engineering*, 20(4):343–353, 1991.

[130] T. Gupta. Design of manufacturing cells for flexible environment considering alternative routeing. *IJPR*, 31(6):1259–1273, 1993.

[131] T. Gupta and H. Seifoddini. Clustering algorithms for the design of a cellular manufacturing system - an analysis for their performance. *Computers Industrial Engineering*, 19(1):432–435, 1990.

[132] T. Gupta and H. Seifoddini. Production data based similarity coefficient for machine-component grouping decisions in the design of a cellular manufacturing system. *IJPR*, 28(7):1247–1269, 1990.

[133] Y. Gupta, M. Gupta, A. Kumar, and C. Sundram. Minimizing total intercell and intracell moves in cellular manufacturing: A genetic algorihtm approach. *International Journal of Computer Integrated Manufacturing*, 8(2):92–101, 1995.

[134] J. Gznar, A. Mehrez, and O. F. Offodile. Formulation of the machine cell grouping problem with capacity and material movement constraints. *JMS*, 13(4):241–250, 1994.

[135] I. Ham, K. Hitomi, and T. Yoshida. *Group Technology*. Kluwer-Nijhoff, Boston, 1985.

[136] I.Y. Ham and C. Han. Multiobjective cluster analysis for part family formations. *JMS*, 5(4):223–229, 1986.

[137] G. Harhalakis, G. Ioannou, I. Minis, and R. Nagi. Manufacturing cell formation under random product demand. *IJPR*, 32(1):47–64, 1994.

[138] G. Harhalakis, I. Minis, and R. Nagi. Development of application of a knowledge based system for cellular manufacturing. In *Third International Conference on Expert Systems and the Leading Edge in Production and Operations Management*, pages 343–355, University of South Carolina, 1989.

[139] G. Harhalakis, R. Nagi, and J.M. Proth. An efficient heuristic in manufacturing cell formation for group technology applications. *IJPR*, 28(1):185–198, 1990.

[140] G. Harhalakis, J.M. Proth, and X.L. Xie. Manufacturing cell disgn using simulated annealling: an industrial application. *Journal of Intelligent Manufacturing*, 1:18–, 1990.

[141] N. Harvey. Socio-technical organization of cell manufacturing and production islands in the metal manufacturing industry in germany and the usa. *International Journal of Production Research*, 32(11):2669–2681, 1994.

[142] S. Heragu and Y.P. Gupta. A heuristic for designing cellular manufacturing facilities. *IJPR*, 32(1):125–140, 1994.

[143] S. Heragu and Kakuturi. Grouping and placement of machine cells. Technical Report A working Paper, Rensselaer Polytechnic Institute, 1994.

[144] S. S. Heragu. A heuristic algorithm for identifying machine cells. *Information and Decision Technologies*, Forthcoming, 1992.

[145] S. S. Heragu. *Design problems and techniques in cellular manufacturing systems*, volume 60, pages 379–416. Academic Press, New York, 1994.

[146] S. S. Heragu. Group technology and cellular manufacturing. *IEEE Transactions on Systems, Man, and Cybernetics*, 24(2):203–215, 1994.

[147] S.S. Heragu. Knowledge based approach to machine cell layout. *Computers in Industrial Engineering*, 17(1-4):37–42, 1989.

[148] A. Hertz, B. Jaumard, and C. Ribeiro. A graph theory approach to subcontracting, machine duplication, and intercell moves in cellular manufacturing. *Discrete Applied Mathematics*, 50:255–265, 1994.

[149] A. Hertz, B. Juamard, and C. C. Riberio. A multi-criteria tabu search approach to cell formation problems in group technology with multiple objectives. *RAIRO Rech. Opr.*, Forthcomming, 1994.

[150] P. Y. Huang and Houck B. L. W. Cellular manufacturing: An overview and bibliography. *Production and Inventory Management*, (4):83–93, 1985.

[151] V. Huber and N. Hyer. The human impact of cellular manufacturing. *Journal of Operations Management*, 5(2):213–228, 1985.

[152] F. Hug. Labor issues in the implementation of group technology cellular manufacturing. *Production and Inventory Management Journal*, 33(4):15–36, 1992.

[153] F. Huq, M. Kurpad, and M.K. Raja. The use of relational database management systems(dbms) for information retrieval in a group technology (gt) environment. *Computers Industrial Engineering*, 26(2):253–266, 1994.

[154] S. Hwang. A constraint-directed method of solve the part selection problem in flexible manufacturing systems planning stage. In *Second ORSA/TIMS Conference on Flexible Manufacturing Systems*, pages 297–309, Ann Arbor, Michigan: University of Michigan, 1986.

[155] N.L. Hyer and U. Wemmerlov. Group technology in the us manufacturing industry: A survey of current practices. *IJPR*, 27(8):1287–1304, 1989.

[156] S. A. Irani, P. H. Cohen, and T. M. Cavalier. Design of cellular manufacturing systems. *Transactions of the ASME*, 114(August):352–361, 1992.

[157] S.A. Irani, T.M. Cavalier, and P.H. Cohen. Virtual manufacturing cells: Exploiting layout design and intercell flows for the machine sharing problem. *IJPR*, 31(4):791–810, 1993.

[158] S.K. Irani and S.K. Khator. A microcomputer based design of a cellular manufacturing system. In *8th Annual Conference on Computers and Industrial Engineering*, 1986.

[159] A.K. Jain, R.G. Kasilingam, and S.D. Bhole. Cell formation in flexible manufacturing systems under resource constraints. *Computers Industrial Engineering*, 19(1-4):437–441, 1990.

[160] J.A. Joines. Manufacturing cell design using genetic algorithms. Ms thesis, North Carolina State University, Raleigh, NC, 1993.

[161] J.A. Joines, C.T. Culbreth, and R.E. King. Manufacturing cell design: An integer programming model employing genetic algorithms. Technical Report NCSU-IE Technical Report 93-16, North Carolina State University, 1993.

[162] J.A. Joines, C.T. Culbreth, and R.E. King. A genetic algorithm based integer program for manufacturing cell design. In *International Conference on Flexible Automation and Integrated Manufacturing*, Stuttgart, Germany, 1995.

[163] J.A. Joines, C.T. Culbreth, and R.E. King. Manufacturing cell design: An integer programming model employing genetic. *IIE Transactions*, 28(1):69–85, 1996.

[164] J.A. Joines and C.R. Houck. On the use of non-stationary penalty functions to solve constrained optimization problems with genetic algorithms. In *1994 IEEE International Symposium Evolutionary Computation*, pages 579–584, Orlando, Fl, 1994.

[165] J. Jung and R. Ahluwalia. Forcod: A coding and classification system for formed parts. *JMS*, 10(3):223–232, 1991.

[166] S. V. Kamarthi, S.R.T. Kumara, F.T.S. Yu, and I. Ham. Neural networks and their application in componnent design data retrieval. *Journal of Intelligent Manufacturing*, 1(2):125–140, 1990.

[167] A. Kamrani and H. Parsaei. A methodology for forming manufacturing cells using manufacturing and design attributes. *Computers Industrial Engineering*, 23(1-4):73–76, 1992.

[168] L. Kandiller. A comparative study on cell formation in cellular manufacturing systems. *International Journal of Production Research*, 32(10):2395–2429, 1994.

[169] S. Kang and U. Wemmerlov. A work load-oriented heuristic methodology for manufacturing cell formation allowing reallocation of operations. *EJOR*, 69(3):292–311, 1993.

[170] Y. Kao and Y.B. Moon. Learning part families by the backpropagation rule of neural networks. In *1st International Conference on Automation Technology*, pages 819–824, Hsinchu, Taiwan, 1990.

[171] Y. Kao and Y.B. Moon. A unified group technology implementation using the backpropagation learning rule of neural networks. *Computers Industrial Engineering*, 20(4):425–437, 1991.

[172] S. Kaparthi. An improved neural network leader algorithm for part-machine grouping in group technology. *EJOR*, 69(3):342–356, 1993.

[173] S. Kaparthi and N.C. Suresh. A neural network system for shape-based classification and coding of rotational parts. *IJPR*, 29(9):1771–1784, 1991.

[174] S. Kaparthi and N.C. Suresh. Machine-component cell formation in group technology: A neural network approach. *IJPR*, 25(6):1353–1367, 1992.

[175] J.A. Kasilingam and S.D. Bhole. Cell formation in flexible manufacturing systems under resource constraints. *Computers in Industrial Engineering*, 19:437–441, 1990.

[176] K. Kato, F. Oba, and F. Hashimoto. Gt-based heuristic approach for machine loading and batch formation in flexible manufacturing systems. *Control Engineering Practice*, 1(5):845–850, 1993.

[177] M. Kazerooni. Cell formation using genetic algorithms. In *International Conference on Flexible Automation and Integrated Manufacturing*, Stuttgart, Germany, 1995.

[178] G.M. Kern and J.C. Wei. The cost of eliminating exceptional elements in group technology cell formation. *IJPR*, 29(8):1535–1547, 1991.

[179] S.K. Khator and S.K. Irani. Cell formation in group technology: A new approach. *Computers Industrial Engineering*, 12(2):131–142, 1987.

[180] J.R. King. Machine-component grouping formation in group technology. *International Journal of Management Science*, 8(2):193–199, 1980.

[181] J.R. King. Machine-component grouping in production flow analysis: An approach using a rank order clustering algorithm. *IJPR*, 18(2):213–232, 1980.

[182] J.R. King and V. Nakornchai. Machine-component group formation in group technology: Review and extension. *IJPR*, 20(2):117–133, 1982.

[183] K.R. Kumar and M.P. Chandrasekharan. Grouping efficacy: a quantitative criterion for goodness of block diagonal forms of binary matrices in group technology. *IJPR*, 28(2):233–243, 1990.

[184] K.R. Kumar, A. Kusiak, and A. Vannelli. Grouping of parts and components in flexible manufacturing systems. *EJOR*, 24:387–397, 1986.

[185] K.R. Kumar and A. Vannelli. Design of flexible production systems: Capacity balancing and subcontracting strategies. In *Second ORSA/TIMS Conference on Flexible Manufacturing Systems: Operations Research Models and Applications*, pages 203–208, Amsterdam, 1986.

[186] K.R. Kumar and A. Vannelli. Strategic subcontracting for efficient disaggregated manufacturing. *IJPR*, 25(12):1715–1728, 1987.

[187] A. Kusiak. Part families selection model for flexible manufacturing systems. *IJPR*, 28(2):233–243, 1983.

[188] A. Kusiak. The generalized group technology concept. *IJPR*, 25(4):561–569, 1987.

[189] A. Kusiak. Exgt-s: A knowledge based system for group technology. *IJPR*, 26:887–905, 1988.

[190] A. Kusiak. *Computational Experience with the knowledge-based system for group technology(KGBT)*, pages 107–118. North-Holland, Amsterdam, 1990.

[191] A. Kusiak. *Intelligent Manufacturing Systems.* Prentice Hall, Englewood Cliffs, N.J., 1990.

[192] A. Kusiak. Branching algorithms for solving the group technology problem. *JMS*, 10(4):332–343, 1991.

[193] A Kusiak. Group technology: Models and solution approaches. In *First Industrial Engineering Research Conference*, pages 349–352, 1992.

[194] A. Kusiak. *Intelligent Design and Manufacturing.* John Wiley and Sons, New York, 1992.

[195] A. Kusiak, W. Boe, and C. Cheng. Designing cellular manufacturing systems: branch-and-bound and a* approaches. *IIE Transactions*, 25(4):46–56, 1993.

[196] A. Kusiak and C.H. Cheng. A branch-and-bound algorithm for solving the group technology problem. *Annals of Operations Research*, 26:415–431, 1990.

[197] A. Kusiak and C.H. Cheng. *Group Technology: Analysis of selected models and algorithms*, volume 53, pages 99–114. ASME, New York, 1991.

[198] A. Kusiak and M. Cho. Similarity coefficient algorithms for solving the group technology problem. *IJPR*, 30(11):2633–2646, 1992.

[199] A. Kusiak and W. Chow. Efficient solving of the group technology problem. *JMS*, 6(2):117–124, 1987.

[200] A. Kusiak and W. Chow. Decomposition of manufacturing systems. *RA*, 4(5):457–471, 1988.

[201] A. Kusiak and Y. Chung. Gt/art: Using neural networks to form machine cells. *Manufacturing Review*, 4:293–301, 1991.

[202] A. Kusiak and S.S. Heragu. Group technology. *Computers Industrial Engineering*, 9:83–91, 1987.

[203] A. Kusiak, A. Vannelli, and K.R. Kumar. Cluster analysis: Models and algorithms. *Control and Cybernetics*, 15(2):139–154, 1986.

[204] A. Kusiak and I. Wadodd. Knowledge based system for group technology(kbgt). In *1st International Conference on CIM*, pages 184–193, Troy, New York, 1988.

[205] R.S. Lashkari and K.R. Gunnasingh. A lagrangian relaxation approach to machine allocation in cellular manufacturing systems. *Computers Industrial Engineering*, 19:442–446, 1990.

[206] H. Lee and A. Garcia-Diaz. A network flow approach to solve clustering problems in group technology. *IJPR*, 31(3):603–612, 1993.

[207] H. Lee, C.O. Malave, and S. Ramachandran. A self-orgainizing neural network approach for the design of cellular manufacturing systems. *Journal of Intelligent Manufacturing*, 3:314–320, 1992.

[208] R.J.V. Lee. Design considerations of automated guided vehicles in a cellular manufactuirng environment. *International Journal of Operations and Production Managment*, 13(1):35–70, 1993.

[209] S. Lee and H.P. Wang. Manufacturing cell formation: A dual-objective simulated annealing approach. *International Journal of Advanced Manufacturing Technology*, 7:314–320, 1992.

[210] S. Lee, C. Zhang, and H. Wang. Fuzzy set-based procedures for machine cell formation. In *Design, Analysis and Control of Manufacturing Cells*, pages 31–45, New York, 1991.

[211] Y. Lemonie and B. Mutel. *Automatic recognition of production cells and part families*, page 239. North-Holland Publish Company, Amsterdam, 1983.

[212] J.K. Lenstra. Clustering a data array and the traveling-salesman problem. *Operations Research*, 22:413–414, 1974.

[213] J. Li and Z. Ding. Fuzzy cluster analysis and fuzzy pattern recognition methods for formation of part families. In *16th North American Manufacturing Research Conference (NAMRC)*, pages 588–563, 1988.

[214] M. Liang and S. Taboun. Part selection and part assignment in flexible manufacturing systems with cellular layout. *Computers Industrial Engineering*, 23(1-4):63–67, 1992.

[215] T. W. Liao. Design of line-type cellular manufacturing systems for minimum operation and material-handling costs. *International Journal of Production Research*, 32(2):387–397, 1994.

[216] T.W. Liao and L.J. Chen. An evaluation of art1 neural models for gt part family and machine cell forming. *JMS*, 12(4):282–290, 1993.

[217] C. Liu and J. Wu. Machine cell formation: using the simulated annealing algorithm. *Interational Journal of Comupter Integrated Manufacturing*, 6(6):335–349, 1993.

[218] R. Logendran. A workload based model for minimizing total intercell and intracell moves in cellular manufacturing. *IJPR*, 28(5):913–925, 1990.

[219] R. Logendran. Impact of sequence of operations and layout of cells in cellular manufacturing. *IJPR*, 29(2):375–390, 1991.

[220] R. Logendran. A model for duplicating bottleneck machines in the presence of budgetary limitations in cellular manufacturing. *IJPR*, 30(3):683–694, 1992.

[221] R. Logendran. Simultaneous machine-part grouping approach in manufacturing cells. *Computers Industrial Engineering*, 23(1-4):77–80, 1992.

[222] R. Logendran. A binary integer programming approach for simultaneous machine-part grouping in cellular manufacturing systems. *Computers Industrial Engineering*, 24(3):329–336, 1993.

[223] R. Logendran, P. Ramakrishna, and C. Sriskandarajah. Tabu search-based heuristics for cellular manufacturing systems in the presence of alternative process plans. *IJPR*, 32(2):273–297, 1994.

[224] R. Logendran and P. Ramkrishna. Manufacturing cell formation in the presence of lot splitting and multiple units of the same machine. *International Journal of Production Research*, 33(3):675–693, 1995.

[225] R. Logendran and T.M. West. A machine-part based grouping algorithm in cellular manufacturing. *Computers Industrial Engineering*, 19(1-4):57–61, 1990.

[226] R. Logendran and T.M. West. A comparison of methodologies for efficient part-machine cluster formation. *Computers Industrial Engineering*, 21:285–289, 1991.

[227] L.H.S. Luong. A cellular similarity coefficient algorithm for the design of manufacturing cells. *IJPR*, 31(8):1757–1766, 1993.

[228] D. Luzzatto and M. Perona. Cell formation in pcb assembly based on production quantitative data. *EJOR*, 69(3):312–329, 1993.

[229] J.B. MacQueen. Some methods for classification and analysis of multivariate observations. In *5th Symposium on Mathematical Statistics and Probability*, page 281, University of California, Berkley, 1967.

[230] B.B. Malakooti and Z. Yang. Group formation by multiple criteria neural network clustering. In *2nd International Engineering Research Conference*, pages 822–826, Los Angeles, CA, 1993.

[231] C. O. Malave and S. Ramachandran. A neural network-based design of cellular manufacturing system. *Journal of Intellignet Manufacturing*, 2:305–314, 1991.

[232] J. McAuley. Machine grouping for efficient production. *Production Engineer*, 51(2):53–57, 1972.

[233] W. T. McCormick, Jr., P.J. Schweitzer, and T.W. White. Problem decomposition and data reorganization by a cluster technique. *Operations Research*, 20(5):993–1009, 1972.

[234] Z. Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs.* AI Series. Springer-Verlag, New York, 3rd edition, 1996.

[235] J. Miltenburg and A.R. Montazemi. Revisiting the cell formation problem: assigning parts to production systems. *IJPR*, 31(11):2727–2746, 1993.

[236] J. Miltenburg and W. Zhang. A comparative evaluation of nine will-known algorithms for solving the cell formation in group technology. *Journal of Operations Management*, 10(1):44–72, 1991.

[237] Hokey Min and Dooyoung Shin. Simultaneous formation of machine and human cells in group technology: a multiple objective approach. *IJPR*, 31(10):2307–2318, 1993.

[238] I. Minis, G. Harhalkis, and S. Jajodia. Manufacturing cell formation with multiple, functionally identical machines. *Manufacturing Review*, 3(4):252–261, 1990.

[239] S.P. Mitrofanov. Scientific principles of group technology. Technical report, Leningrad, 1959.

[240] S.P. Mitrofanov. Scientific principles of group technology. Technical report, London: National Lending Library, 1966.

[241] S. Miyamoto. *Fuzzy Sets in Information Retrieval and Cluster Analysis.* Kluwar Academic Publishers, Boston, 1990.

[242] Y.B. Moon. Forming part families for cellular manufacturing: a neural network approach. *International Journal of Advanced Manufacturing Technology*, 5:278–291, 1990.

[243] Y.B. Moon. An interactive and competition model for machine-part family formation in group technology. In *International Joint Conference on Neural Networks*, pages 667–670, Washington, D.C., 1990.

[244] Y.B. Moon and S.C. Chi. Generalized part family formation using neural network techniques. *JMS*, 11(3):149–159, 1992.

[245] Y.B. Moon and U. Roy. Learning group technology part families from solid models by parallel distributed processing. *International Journal of Advanced Manufacturing Technology*, 7:109–118, 1992.

[246] J. S. Morris and R. J. Tersine. Simulation analysis of factors influencing the attractiveness of group technology cellular layoiuts. *Management Science*, 36(12):1567–1578, 1990.

[247] J.S. Morris and R.J. Tersine. A comparison of cell loading practices in group technology. *Journal of Manufacturing and Operations Management*, 2:299–313, 1989.

[248] C. Mosier and L. Taube. The facets of group technology and their impacts on implementation - a state-of-the-art survey. *International Journal of Management Science*, 13:381–391, 1985.

[249] C.T. Mosier. An experiment investigating the application of clustering procedures and similarity coefficients to the gt machine cell formation problem. *IJPR*, 27(10):1811–1835, 1989.

[250] C.T. Mosier, D.A. Elvers, and D. Kelly. Analysis of group technology heuristics. *IJPR*, 22:857–875, 1984.

[251] C.T. Mosier and L. Taube. A weighted similarity coefficient for use in addressing the group technology part-machine clustering problem. In *American Institute of Decision Sciences Annual Meeting*, pages 812–815, Las Vegas, 1985.

[252] C.T. Mosier and L. Taube. Weighted similarity measure heuristics for the group technology machine clustering problem. *International Journal of Management Science*, 13(6):577–583, 1985.

[253] S. Mukhopadhyay, A. Gopalakrishnan, and M.K. Kripalani. Moments-based clustering techniques for manufacturing cell formation. *International Journal of Production Research*, 33(4):1091–1115, 1995.

[254] S.K. Mukhopadhyay, P. Sarkar, and R. P. Panda. Machine-component grouping in cellular manufacturing by multidimensional scaling. *IJPR*, 32(2):457–478, 1994.

[255] J. Mulvey and H. Crowder. Cluster analysis: An application of lagrangian relaxation. *Management Science*, 25:329–340, 1979.

[256] R. Nagi, G. Harhalakis, and J.M. Proth. Multiple routings and capacity considerations in group technology applications. *IJPR*, 28(12):2243–2257, 1990.

[257] S. Ng. Bond energy, rectilinear distance and a worst-case bound for the group technology problem. *Journal of the Operational Research Society*, 42(7):571–578, 1991.

[258] S. Ng. Characterizing the independent cells in group technology. Technical Report Working Paper, 1992.

[259] S. Ng. On the measures of cell formation in group technology. In *Proceedings of the First Industrial Engineering Research Conference*, pages 353–356, 1992.

[260] S. Ng. Worst-case analysis of an algorithm for cellular manufacturing. *EJOR*, 69(3):384–398, 1993.

[261] None. Tips on tackling gt-based cells. *Manufacturing Enigneering*, 106(2):46–51, 1991.

[262] F. Oba, K. Kato, K. Yasuda, and T. Tsumura. Review and extension of cell formation problems in flexible manufacturing systems. In *Second IFIP Conference*, pages 61–78, Copenhagen, Denmark, 1987.

[263] O. Offodile. Assignment model formulation of the machine cell formation problem in cellular manufacturing. *International Journal of Operations and Production Managment*, 13(10):49, 1993.

[264] O. F. Offodile. Machine grouping in cellular manufacturing. *International Journal of Management Science*, 21(1):35–52, 1993.

[265] O. F. Offodile, A. Mehrez, and J. Grznar. Cellular manufacturing: A taxonomic review framework. *JMS*, 13(4):196–220, 1994.

[266] O.F. Offodile. Application of similarity coefficient method to parts coding and classification analysis in group technology. *JMS*, 10(6):442–448, 1991.

[267] E. Olivia-Lopez and G.F. Purcheck. Load balancing for group technology planning and control. *International Journal of Machine Tool Design and Research*, 19:259, 1979.

[268] H. Opitz. *A Classification to Describe Workpieces*. Pergamon Press, Oxford, 1970.

[269] H. Opitz, W. Eversheim, and H.P. Wiendahl. Workpiece classification and its industrial application. *International Machine Tool Design and Research*, 9:39, 1969.

[270] N.E. Ozdemirel. A generic simulation module architecture based on clustering group technology model codings. *Simulation*, 60(6):421–, 1993.

[271] N.E. Ozdemirel, G.T. Mackulak, and J.K. Cochran. A group technology classification and coding scheme for discrete manufacturing simulation models. *International Journal of Production Research*, 31(3):579–603, 1993.

[272] R.D. Pullen. A survey of cellular manufacturing cells. *Production Engineer*, 56:431–454, 1976.

[273] G. Purcheck. Machine-component group formation: An heuristic method for flexible production cells and flexible manufacturing systems. *IJPR*, 23(5):991–993, 1985.

[274] G.F.K. Purcheck. A mathematical classification as a basis for the design of group technology production cells. *Production Engineer*, 54(1):35–48, 1974.

[275] G.F.K. Purcheck. Combinatorial grouping–a lattice theoretic method for the design of manufacturing systems. *Journal of Cybernetics*, 1975.

[276] G.F.K. Purcheck. A linear programming method for the combinatorial grouping of an incomplete power set. *Journal of Cybernetics*, 5:51–76, 1975.

[277] G.F.K. Purcheck. Combinatorial analysis in planning for cellular manufacture. In *9th CIRP International Seminar on Manufactory Systems*, Bedford: Cranfield Institute of Technology, 1977.

[278] R. Rajagopalan and J. Batra. Composite components through graphs and fuzzy clusters. In *18th International Machine Tool Design and Research Conference*, New York, 1978.

[279] R. Rajagopalan and J.L. Batra. Design of cellular production systems: A graph-theoretic approach. *IJPR*, 13(6):567–579, 1975.

[280] D. Rajamani, N. Singh, and Y. Aneja. Integrated design of cellular manufacturing systems in the presence of alternative process plans. *IJPR*, 30(6):1541–1554, 1990.

[281] D. Rajamani, N. Singh, and Y. Aneja. A model for cell formation in manufacturing systems with sequence dependence. *IJPR*, 28(8):1227–1235, 1992.

[282] D. Rajamani, N. Singh, and Y.P. Aneja. Selection of parts and machines for cellularization: A mathematical programming approach. *EJOR*, 62:47–54, 1992.

[283] G.M. Ransom. *Group Technology.* McGraw Hill, New York, 1972.

[284] H. A. Rao and Gu P. Expert self-organizing neural network for the design of cellular manufacturing systems. *JMS*, 13(5):346–358, 1994.

[285] H.A. Rao and P. Gu. Design of a cellular manufacturing systems: A neural net approach. *International Journal of Systems Automation: Research and Applications*, 2:407–424, 1992.

[286] H.A. Rao and Gu P. A mulit-constraint neural network for the pragmatic design of cellular manufacturing systems. *Internation Journal of Production Research*, 33(4):1049–1070, 1995.

[287] H. Rikken and J. Slomp. Testing the neural network approach of kaparthi and suresh. Technical report, University of Twente, The Netherlands, 1992.

[288] G.T. Ross and R.M. Soland. A branch and bound algorithm for the generalized assignment problem. *Computers and Operations Research*, 8:91–, 1977.

[289] R. S. Russell, P. Y. Huang, and Y. Leu. A study of labor allocation strategies in cellular manufacturing. *Decision Sciences*, 22:594–611, 1991.

[290] S. Sankaran. Multiple objective decision making approach to cell formation: A goal programming model. *Mathematical Computer Modeling*, 13:71–82, 1990.

[291] S. Sankaran and R.G. Kasilingam. An integrated approach to cell formation and part routing in group technology manufacturing systems. *Engineering Optimization*, 16:235–245, 1990.

[292] S. Sankaran and R.G. Kasilingam. On cell size and machine requirements planning in group technology systems. *EJOR*, 69(3):373–383, 1993.

[293] B. R. Sarker and J. Yu. A two-phase procedure for duplicating bottleneck machines in a linear layout, cellular manufacturing system. *International Journal of Production Research*, 32(9):2049–2066, 1994.

[294] F. Sassani. A simulation stdy on performance imporvement of group technology cells. *International Journal of Production Research*, 28(2):293–300, 1990.

[295] H. Seifoddini. Improper machine assignment in machine component grouping in group technology. In *Fall Industrial Engineering Conference*, pages 406–409, 1986.

[296] H. Seifoddini. Duplication process in machine cell formation in group technology. *IIE Transactions*, 21(4):382–388, 1989.

[297] H. Seifoddini. A note on the similarity coefficient method and the problem of improper machine assignment in group technology applications. *IJPR*, 27(7):1161–1165, 1989.

[298] H. Seifoddini. Single linkage versus average linkage clustering in machine cell formation applications. *Computers Industrial Engineering*, 16:419–426, 1989.

[299] H. Seifoddini. Machine-component group analysis versus the similarity coefficient method in cellular manufacturing applications. *Computers in Industrial Engineering*, 18(3):333–339, 1990.

[300] H. Seifoddini. A probabilistic model for machine cell formation. *JMS*, 9(1):69–75, 1990.

[301] H. Seifoddini. Performance evaluation of hybird cells. *IIE Transactions*, 24:84–88, 1992.

[302] H. Seifoddini and M. Djassemi. The production data-based similarity coefficient versus jaccard's similarity coefficient. *Computers Industrial Engineering*, 21(1-4):263–266, 1991.

[303] H. Seifoddini and M. Djassemi. Merits of the producion volume based similarity coefficient in machine cell formation. *JMS*, 14(1):35–44, 1995.

[304] H. Seifoddini and C. Hsu. Comparative study of similarity coefficients and clustering algorihtms in cellular manufacturing. *JMS*, 13(2):119–127, 1994.

[305] H. Seifoddini and P.M. Wolfe. Application of the similarity coefficient method in group technology. *IIE Transactions*, 18(3):271–277, 1986.

[306] H. Seifoddini and P.M. Wolfe. Selection of a threshold value based on material handling cost in machine-component grouping. *IIE Transactions*, 19(3):266–270, 1987.

[307] A. Selim, A. J. Vakharia, and R. G. Askin. Mathematical models of cell formation: Review and extensions. In *ORSA/TIMS National Meeting*, Nashville, Tennessee, 1991.

[308] H.M. Selim. *A Flexible Cell Formation Approach for Cellular Manufacturing*. Phd dissertation, University of Arizona, Tuscon, Tuscon, AZ, 1993.

[309] H.M. Selim, R.G. Askin, and A.J. Vakharia. Cell formation in group technology: Review, evaluation and directions of future research. Technical Report Working Paper, University of Arizona, Tucson, 1994.

[310] H.M. Selim, A.J. Vakharia, and R.G. Askin. Flexibility in cellular manufacturing: A framework and measures. Technical report, University of Arizona, Tucson, 1993.

[311] R.P. Selvam and K.N. Balasubramanian. Algorithmic grouping of operation sequences. *Engineering Costs and Production Economics*, 9:125–134, 1985.

[312] S. Shafer, G. Kern, and J. Wei. A mathematical programming approach for dealing with exceptional elements in cellular manufacturing. *IJPR*, 30(5):1029–1036, 1987.

[313] S.M. Shafer and J.M. Charnes. A simulation analyses of factors influencing loading practices in cellular manufacturing. *International Journal of Production Research*, 33(1):279–290, 1995.

[314] S.M. Shafer and J.R. Meredith. A comparison of selected manufacturing cell formation techniques. *IJPR*, 28(4):661–673, 1990.

[315] S.M. Shafer and J.R. Meredith. An empirically-based simulation study of functional versus cellular layouts with operations overlapping. *International Journal of Operations and Production Management*, 13:47–62, 1993.

[316] S.M. Shafer and D.F. Rogers. A goal programming approach to the cell formation problem. *Journal of Operations Management*, 1991.

[317] S.M. Shafer and D.F. Rogers. Similarity and distance measures for cellular manufacturing part i. a survey. *IJPR*, 31(5):1133–1142, 1993.

[318] S.M. Shafer and D.F. Rogers. Similarity and distance measures for cellular manufacturing part ii. an extension and comparison. *IJPR*, 31(6):1315–1326, 1993.

[319] S.M. Shafer, B.J. Tepper, and R. Marsh. Comparing the effects of cellular and functional manufacturing on employees' perceptions and attitudes. *Journal of Operations Management*, 12(2):63–75, 1995.

[320] T. Shaffer and R. Billo. A demand-based method for manufacturing cell design and replication. *International Journal of Manufacturing System Design*, 1(2), 1994.

[321] M. Shargal, S. Shekhar, and S.A. Irani. Evaluation of search algorithms and clustering efficiency measures for machine-part matrix clustering. *IIE Transactions*, to appear, 1994.

[322] S.M. Shenoy and R.G. Kasilingam. Performance analysis of machine cell configurations using simulation. *Computers Industrial Engineering*, 21(1-4):279–283, 1991.

[323] G. Shiko. A process planning-oriented approach to part family formation in group technology applications. *IJPR*, 30(8):1739–1752, 1992.

[324] A. Shtub. Modeling group technology cell formation as a generalized assignment problem. *IJPR*, 27(5):775–782, 1989.

[325] N. Singh. Design of cellular manufacturing systems: An invited review. *EJOR*, 69(3):284–291, 1993.

[326] N. Singh, Y.P. Aneja, and S.P. Rana. A bicriterion framework for operations assignment and routing flexibility analysis in cellular manufacturing systems. *EJOR*, 60(2):200–210, 1992.

[327] J.L. Slagle, C.L. Chang, and S.R. Heller. A clustering and data reorganization algorithm. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-5(2):125–128, 1974.

[328] S. Song and K. Hitomi. Gt cell formation for minimizing the intercell parts flow. *IJPR*, 30(12):2737–2753, 1992.

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

[330] G. Srinivasan, T. Narendran, and B. Mahadevan. An assignment model for the part-families problem in group technology. *IJPR*, 28(1):145–152, 1990.

[331] G. Srinivasan and T.T. Narendran. Grafics-a non hierarchical clustering algorithm for group technology. *IJPR*, 29(3):463–478, 1991.

[332] L.E. Stanfel. Machine clustering for economic production. *Engineering Costs and Production Economics*, 9:73–81, 1985.

[333] L.E. Stanfel. A successive approximations method for a cellular manufacturing problem. *Annals of Operations Research*, 17:13–30, 1989.

[334] K.E. Stecke. Formulation and solution of nonlinear integer production problems for flexible manufacturing systems. *Management Science*, 29:273–288, 1983.

[335] H. Steudal and A. Ballakur. A dynamic programming based heuristic for machine grouping in manufacturing cell formation. *Computers Industrial Engineering*, 12(4):215–222, 1987.

[336] H.J. Steudel. Simshop: A job shop/cellular manufacturing simulator. *JMS*, 5:181–189, 1986.

[337] R. M. Sundaram and K. Doshi. Formation of part families to design cells with alternative routing considerations. *Computers Industrial Engineering*, 23(1-4):59–62, 1992.

[338] R. M. Sundaram and S. Fu. Group technology cell formation - some new insights. *Computers Industrial Engineering*, 13(1-4):267–276, 1987.

[339] C. Sundarm, Y.P. Gupta, and M.C. Gupta. Minimizing total intercell and intracell moves in cellular manufacturing: a genetic algorithm approach. *International Journal of Computer Integrated Manufacturing*, 8(2):92–101, 1995.

[340] N.C. Suresh. Partitioning work centers for group technology: Insights from an analytical model. *Decision Sciences*, 22(4):772–791, 1991.

[341] N.C. Suresh. Partitioning work centers for group technology: Analytical extension and shop-level simulation investigation. *Decision Sciences*, 23:267–290, 1992.

[342] N.C. Suresh and S. Kaparthi. Performance of fuzzy art neural network for group technology cell formation. *International Journal of Production Research*, 32(7):1693–1713, 1994.

[343] S.M. Taboun, S. Sankaran, and S. Bhole. Comparison and evaluation of similarity measures in group technology. *Computers Industrial Engineering*, 20(3):343–353, 1991.

[344] S.M. Taboun and A. Sharma. A weighted index for the design of cellular manufacturing systems. *Computers Industrial Engineering*, 21(1-4):273–277, 1991.

[345] M.T. Tabucanon and R. Ojha. Icrm-a heuristic approach for intercell flow reduction in cellular manufacturing. *Material Flow*, 4:189–197, 1987.

[346] K. Y. Tam. Linguistic modeling of flexible manufacturing systems. *JMS*, 8(2):127–137, 1989.

[347] K.Y. Tam. An operation sequence based similarity coefficient for part families formations. *JMS*, 9:55–68, 1990.

[348] M. Tarsuslugil and J. Bloor. The use of similarity coefficients and cluster analysis in production flow analysis. In *20th International Machine Tool Design and Research Conference*, pages 525–531, 1979.

[349] M.V. Tatikonda and U. Wemmerlov. Adoption and implementation of group technology classification and coding systems: Insights from seven case studies. *IJPR*, 30(9):2087–2110, 1992.

[350] A. J. Vakharia and U. Wemmerlov. An investigation of hierarchical clutering techniques and dissimilarity measures applied to the cell formation problem. Technical Report Working Paper, University of Arizona, Tucson, 1994.

[351] A.J. Vakharia. Methods of cell formation in group technology: A framework for evaluation. *Journal of Operations Management*, 6(3):257–271, 1986.

[352] A.J. Vakharia, Y. Chang, and H.M. Selim. Cell formation in group technology: A combinatorial search approach. Technical Report Working Paper, University of Arizona, Tucson, 1994.

[353] A.J. Vakharia and B.K. Kaku. Redisigning a cellular manufacturing system to handle long-termed demand changes: A methodology and investigation. *Decision Sciences*, 24(5):909–930, 1993.

[354] A.J. Vakharia and U. Wemmerlov. A new similarity index and clustering methodology for formation of manufacturing cells. In *Decision Sciences*, pages 1075–1077, Las Vegas, NE, 1988.

[355] A.J. Vakharia and U. Wemmerlov. Designing a cellular manufacturing system: A material flow approach based on operation sequences. *IIE Transactions*, 22(1):84–97, 1990.

[356] A. Vannelli and R.G. Hall. An eigenvector solution methodology for finding part-machine families. *IJPR*, 31(2):325–349, 1993.

[357] A. Vannelli and K.R. Kumar. A method for finding minimal bottle-neck cells for grouping part-machine families. *IJPR*, 24(2):387–400, 1986.

[358] J. Ventura, F.F. Chen, and C.H. Wu. Grouping parts and tools in flexible manufacturing systems production planning. *IJPR*, 28(6):1039–1056, 1990.

[359] V. Venugopal and T. T. Narendran. A neural network approach for designing cellular manufacturing systems. *Advances in Modeling and Analysis*, 32(2):13–26, 1992.

[360] V. Venugopal and T. T. Narendran. Design of cellular manufacturing systems based on asymptotic forms of a boolean matrix. *EJOR*, 67:405–417, 1993.

[361] V. Venugopal and T.T. Narendran. Cell formation in manufacturing systems through simulated annealing: An experimental evaluation. *EJOR*, 63(3):409–422, 1992.

[362] V. Venugopal and T.T. Narendran. A genetic algorithm approach to the machine-component grouping problem with multiple objectives. *Computers in Industrial Engineering*, 22(4):469–480, 1992.

[363] S. Viswanathan. Configuring cellular manufacturing systems: A quadratic integer programming formulation and a simple interchange heuristic. *International Journal of Production Research*, 33(2):361–376, 1995.

[364] T. Vohra, D. Chen, J. Chang, and H. Chen. A network approach to cell formation in cellular manufacturing. *IJPR*, 28(11):2075–2084, 1990.

[365] P.H. Waghodekar and S. Sahu. Machine-component cell formation in group technology: Mace. *IJPR*, 22(6):937–948, 1984.

[366] J.C. Wei and N. Gaither. A capacity constrained multi-objective cell formation method. *JMS*, 9:222–232, 1990.

[367] J.C. Wei and N. Gaither. An optimal model for cell formation decisions. *Decision Sciences*, 21(2):416–433, 1990.

[368] J.C. Wei and G.M. Kern. Commonality analysis: A linear cell clustering algorithm for group technology. *IJPR*, 27(12):2053–2062, 1989.

[369] J.C. Wei and G.M. Kern. Reply to 'a note on a linear cell clustering algorithm'. *IJPR*, 29(1):217–218, 1991.

[370] U. Wemmerlov. Comments on direct clustering algorithm for group formation in cellular manufacture. *JMS*, 3:vii–ix, 1984.

[371] U. Wemmerlov and N.L. Hyer. Procedures for the part family/machine group identification problem in cellular manufacturing. *Journal of Operations Management*, 6(2):125–147, 1986.

[372] U. Wemmerlov and N.L. Hyer. Cellular manufacturing in the u.s. industry: a survey of users. *IJPR*, 27(9):1511–1530, 1989.

[373] J.D.E. White. The use of similarity coefficient in production flow analysis. *IJPR*, 18(4):504–514, 1980.

[374] H. Wu, M. Venugopal, and M. Barash. Design of a cellular manufacturing system: A syntactic pattern recognition approach. *JMS*, 5(2):81–88, 1986.

[375] N. Wu and G. Salvendy. A modified network approach for the design of cellular manufacturing systems. *IJPR*, 31(6):1409–1421, 1993.

[376] S. P. Wu and P. L. Chang. The synthetic index algorithm: an improved cluster analysis procedure for machine cell formation. *International Journal of Comupter Integrated Manufacturing*, 3(5):299–313, 1990.

[377] X. Xie. Manufacturing cell formation under capacity constraints. *Applied Stochastic Models and Data Analysis*, 9(2):87–96, 1993.

[378] H. Xu and H.P. Wang. Part family formation for gt applications based on fuzzy mathematics. *IJPR*, 27(9):1637–1651, 1989.

[379] C. Zhang and H. Wang. Concurrent formation of part families and machine cells based on the fuzzy set theory. *JMS*, 11(1):61–67, 1992.

[380] H.-C. Zhang and S.H. Huang. Applications of neural networks in manufacturing: A state-of-the-art survey. *International Journal of Production Research*, 33(3):705–728, 1995.