Group Technology and Cellular Manufacturing

Sunderesh S. Heragu

Abstract— **A number of survey papers on group technology and cellular manufacturing system design have been published. Many of them focus primarily on clustering techniques that manipulate rows and columns of the part-machine processing indicator matrix to form a block diagonal structure. Since the last survey paper was published, there have been some developments in cellular manufacturing system design. A number of papers that consider practical design constraints while designing cellular manufacturing systems have been published. The purpose of this paper is to provide a thorough survey of papers on group technology and cellular manufacturing system design. Its purpose is also to state some important design factors that cannot be ignored.**

I. Introduction

GROUP TECHNOLOGY (GT) is a management philosophy that attempts to group products with similar design and/or manufacturing characteristics. Cellular manufacturing (CM) can be defined as an application of GT and involves grouping machines or processes on the basis of parts or part families they process [71]. It is a relatively recent concept and has been applied in many manufacturing environments successfully. There are significant benefits that can be achieved as a result of implementing a CMS. A number of companies surveyed in [70] have witnessed the following:

- Setup time reduction
- Work-in-process inventory reduction
- Material handling cost reduction
- Equipment cost and direct/indirect labor cost reduction
- Improvement in quality
- Improvement in material flow
- Improvement in machine utilization
- Improvement in space utilization
- Improvement in employee morale

The main difference between a traditional jobshop environment and a cellular manufacturing environment is in the grouping and layout of machines. In a jobshop environment, machines are typically grouped based on their functional similarities (see Fig. 1). On the other hand, in a cellular manufacturing environment, machines are grouped into cells so that each cell is dedicated to the manufacture of a specific part family (Fig. 2). Typically, machines in each cell are dissimilar in their functions. Such an arrangement in which sets of machines are dedicated to specific part families allows easier control of a CMS.

Manuscript received August 14, 1992; revised April 21, 1993. This work was supported by grants from the New York State Science and Technology Foundation and Welch Allyn Medical Division, New York.

The author is with the Department of Decision Sciences and Engineering Systems, Rensselaer Polytechnic Institute, Troy, NY 12180-1390.

IEEE Log Number 9214536.

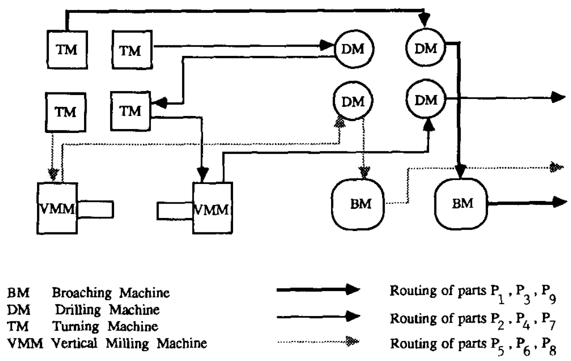


Fig. 1. Arrangement of cells in a jobshop environment.

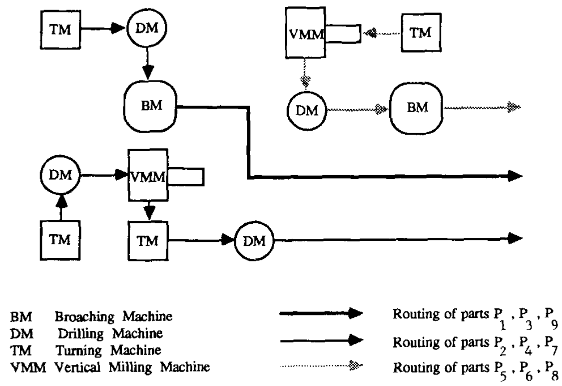


Fig. 2. Arrangement of cells in a cellular manufacturing system.

The main objective in the design of a CMS is to create machine cells, identify part families, and allocate part families to machine cells so that the inter-cellular movement of parts of part families is minimized. Of course, while doing so, a number of other constraints need to be considered. For example, the allocation of part families to machine cells must be such that available capacity of machines in each cell is not exceeded; safety and technological requirements pertaining to the location of equipment and process must be met; size of a cell and number of cells must not exceed a user specified number; etc.

A majority of the papers on GT and CMS design ignore some or all the constraints listed above and focus only on the identification of machine cells and corresponding part families. Given a part-machine processing indicator matrix such as the one shown in Fig. 3, they basically attempt to find a

$$
[a_{ij}] = \begin{array}{cc|ccccccc|}
 & & M & a & c & h & i & n & e \\
 & & M_1 & M_2 & M_3 & M_4 & M_5 & M_6 & M_7 \\
\hline
 & P_1 & 1 & & & 1 & & 1 & \\
P & P_2 & & 1 & 1 & & 1 & & \\
a & P_3 & & & & 1 & & 1 & \\
r & P_4 & & 1 & 1 & & & & \\
t & P_5 & & & 1 & & & & 1 \\
 & P_6 & & 1 & & & 1 & & 1 \\
\hline
\end{array}
$$

Fig. 3. Sample part-machine processing indicator matrix.

$$
[a_{ij}] = \begin{array}{cc|ccc|cccc|}
 & & M & a & c & h & i & n & e \\
 & & M_1 & M_4 & M_6 & M_2 & M_3 & M_5 & M_7 \\
\hline
 & P_1 & 1 & 1 & 1 & & & & \\
P & P_3 & & 1 & 1 & & & & \\
\hdashline
a & P_2 & & & & 1 & 1 & 1 & \\
r & P_4 & & & & 1 & 1 & & \\
t & P_5 & & & & & 1 & & 1 \\
 & P_6 & & & & 1 & & 1 & 1 \\
\hline
\end{array}
$$

Fig. 4. Rearranged processing indicator matrix.

rearrangement of the rows and columns of the matrix that will provide a block diagonal form as shown in Fig. 4. Typically, the part-machine processing indicator matrix consists of 0, 1 entries only; a 1 entry in row i, column j indicates that the part corresponding to the ith row is processed by the machine corresponding to the jth column. A 0 entry means that the part is not processed by the corresponding machine. Typically, zeros are left blank in the matrix as in Figs. 3 and 4.

The clusters of 1's around the diagonal of the matrix in Fig. 4 indicate three groups or machine cells $MC_1 = \{M_1, M_4, M_6\}$ and $MC_2 = \{M_2, M_3, M_5, M_7\}$; the corresponding part families are $PF_1 = \{P_1, P_3\}$ and $PF_2 = \{P_2, P_4, P_5, P_6\}$. It should be noted that not all part-machine processing indicator matrices can be rearranged to fit a block diagonal form such as the one indicated by dashed lines in Fig. 4. For many matrices, such a form may not even exist. For example, assume that part P_2 requires processing on machine M_1 in addition to M_2, M_3 and M_5. The corresponding part-machine processing indicator matrix is shown in Fig. 5. For such a problem, no rearrangement of the rows and columns will produce a block diagonal form such that no 1's lie outside the block indicated by dashed lines.

The rows (parts) corresponding to the 1's that lie outside the diagonal block are usually referred to as exceptional parts because when these parts are removed, a block diagonal structure can be easily identified. Of course, some rearrangement of the remaining rows and columns must still be done to uncover the block diagonal structure.

In the example considered (see Fig. 5), the exceptional part is P_2. If this part is removed from the matrix, two mutually separable part family/machine cell combinations can be easily identified. Some authors have suggested that exceptional parts be subcontracted out. On the other hand, if this part is left

$$
[a_{ij}] = \begin{array}{cc|ccc|cccc|}
 & & M & a & c & h & i & n & e \\
 & & M_1 & M_4 & M_6 & M_2 & M_3 & M_5 & M_7 \\
\hline
 & P_1 & 1 & 1 & 1 & & & & \\
P & P_3 & & 1 & 1 & & & & \\
\hdashline
a & P_2 & 1 & & & 1 & 1 & 1 & \\
r & P_4 & & & & 1 & 1 & & \\
t & P_5 & & & & & & 1 & 1 \\
 & P_6 & & & & 1 & & 1 & 1 \\
\hline
\end{array}
$$

Fig. 5. Revised processing indicator matrix.

as is and machine cells are formed as indicated in Fig. 5, it will be necessary for part P_2 to visit both the cells MC_1 and MC_2. For such types of problems, there are a number of methods that rearrange rows and columns (i.e., attempt to identify part families/machine cells) so as to minimize the intercellular movement of parts.

Just as some problems may have exceptional parts, other problems may have bottleneck machines (columns) which when removed will result in identification of the block diagonal structure. These bottleneck machines are so named because two or more part families share this resource. For example, for the matrix shown in Fig. 5, if the column corresponding to machine M_1 is removed, again two mutually separable clusters of machine cells and part families can be identified. It is well known that additional copies of exceptional machines can be purchased and placed in appropriate cells so as to eliminate or at least minimize the intercellular movement of parts; of course, this must be done only if the cost of bolttleneck machines is not high. A systematic procedure that uses cost information for evaluating the elimination of bottleneck machines and exceptional parts can be found in [32]. Shafer *et al.* [58] also present a mathematical programming approach for dealing with exceptional parts and bottleneck machines.

Zhang and Wang [72] suggest that it is useful to use nonbinary values between 0 and 1 to indicate the degree of match between each part-machine pair in the processing indicator matrix $[a_{ij}]$. They show how the degree of match can be calculated using fuzzy set theory. Compared to the binary matrix approach, in which the only information available is whether or not a part is processed on a machine, this method is very flexible as it provides a mechanism to capture a number of other relationships between each part-machine pair (e.g., cost of processing a part on a machine processing time, etc). Ben-Arieh and Triantaphyllou [5] also use fuzzy set theory and show how qualitative data and quantitative data that have a subjective meaning can be considered while identifying machine cells and part families. They also provide an algorithm that is capable of handling such fuzzy data.

The part-machine processing indicator matrix may be modified to handle other pieces of valuable information that are used in identifying machine cells and part families. For example, two additional columns can be created to indicate the number of parts to be manufactured and the batch size for each part [29]. Similarly, the sequence of machines visited by a part may also be recorded in the matrix [48]. This can be

$$
[a_{ij}] = \begin{array}{cc} & \\ & \\ \mathrm{P} & P_1 \\ \mathrm{a} & P_3 \\ \mathrm{r} & P_2 \\ \mathrm{t} & P_4 \\ & P_5 \\ & P_6 \end{array}
\begin{array}{ccccccc} \mathrm{M} & \mathrm{a} & \mathrm{c} & \mathrm{h} & \mathrm{i} & \mathrm{n} & \mathrm{e} \\ M_1 & M_4 & M_6 & M_2 & M_3 & M_5 & M_7 \\ \hline 2 & 3 & 1 & & & & \\ & 1 & 2 & & & & \\ 3 & & & 1 & 4 & 2 & \\ & & & 2 & 1 & & \\ & & & & & 1 & 2 \\ & & & 1 & & 2 & 3 \end{array}
$$

Fig. 6. Processing indicator matrix indicating sequence of operations.

accomplished simply by defining x_{ij} to be the following:

$$x_{ij} = \begin{cases} k & \text{if part } i \text{ visits machine } j \\ & \text{for the } k\text{th operation} \\ 0 & \text{otherwise.} \end{cases}$$

where k is an integer representing the operation for which part i visits machine j. Such a modified matrix is shown in Fig. 6.

The advantage of such a representation is that available clustering techniques such as the one in [27] can be used. The above representation enables us to capture routing sequence information which is useful in determining material flow between machine pairs. As long as each machine is visited by a part for any number of consecutive operations, the representation shown in Fig. 6 is adequate. All the consecutive operations on the same machine may be treated as a single operation. However, if the same machine is visited for two or more nonconsecutive operations, then the above representation is not adequate. In such a case, each entry in the above matrix will have to modified as a vector representing all the operations for which the corresponding machine is visited by the part. For example, if part P_5 visits machine M_2 for the second and fourth operation, then this information will have to be stored in the form of a vector—(2,4) in the 5th row and 2nd column of the matrix. This obviously increases the memory requirements for data storage. Hence, Heragu and Kakuturi [29] suggest an alternate way of representing nonconsecutive opeations on the same machine by use of dummy columns. In general, it can be argued that it is advantageous to modify the part design or select alternate process plans so that the same machine is visited more than once for consecutive operations only; otherwise there will be an unnecessary increase in set-up time.

The above modifications suggested are useful in determining the flow between each machine pair. The flow information is used in some algorithms for identifying machine cells (see [28] for example).

To date, a number of survey papers on GT have been published. However, all of them have primarily surveyed clustering techniques (i.e., techniques that are concerned with the rearrangement of rows and columns of the binary part-machine processing indicator matrix so as to identify a block diagonal structure). Moreover, since the survey paper in 1988, a number of developments have taken place. A number of algorithms that are concerned with practical CMS design problems have been published. It is therefore necessary to develop a new framework over which research attention can be focused. The focus of this paper is on GT *and* CMS design. Consequently, techniques for GT and CMS design are surveyed in detail with emphasis on papers pertaining to the latter category. Also, a list of some of the important constraints that msut be considered while designing a CMS are presented. Each constraint is discussed in detail. Some general concluding remarks are also provided.

II. Various Classifications of GT Techniques

Mitrofanov [46] first introduced the GT philosophy and the machine grouping problem in the late 1950's in the U.S.S.R. The other early pioneers in the field of GT are Burbidge [9] and Ham [23]. Since then, a number of researchers have developed techniques for solving the GT and CM problems. These techniques have been classified and reclassified a number of times.

Burbidge [9] classified the techniques as follows:

1) *Rule of Thumb techniques*: These techniques, sometimes referred to as eyeball methods, use some simple rules of thumb to identify part families and machine cells [6]. Clearly such techniques are not useful in solving large-scale problems, but are relatively easy to use. In other words, they can be used to obtain "quick-and-dirty" solutions to small-scale problems. Because there are a number of efficient techniques available today, these are not preferred.
2) *Classification and coding techniques*: Unlike the Production Flow Analysis (PFA) techniques (discussed below) which group parts solely based on their processing characteristics, classification and coding techniques group parts on the basis of a number of attributes. Examples of such attributes are shape of part; dimensions of part; material composition of part; tolerance requirement and operations requirement. Typically, each part is assigned a ten- to thirty-digit code with each code representing an attribute of the part. In some classification and coding systems, a hierarchical relationship exists amongst the digits; i.e., information stored in digits with higher numbers is related to the information stored in digits with lower numbers. For example, digit 2 may represent the part shape attribute, digit 3 may represent rotational parts, digit 4 may respresent rotational parts with diameter greater than 3 inches, and so on. In some other classification and coding systems, such a hierarchical relationship may exist partially or may not exist at all. Thus, depending upon whether or not a hierarchical relationship exists, classification and coding systems have been classified as [41]

   •hierarchical codes;
   •non-hierarchical codes; and
   •hybrid codes.

Ham *et al.* [24] provided a survey of classification and coding techniques.

3) *Production Flow Analysis (PFA) techniques*: PFA techniques involve the systematic listing of information

contained in route cards and identification of part families and machine cells by careful inspection. Some of the latter forms of PFA techniques use a part-machine processing indicator matrix to specify the machining requirements on parts and then attempt to manipulate the rows and columns of this matrix to identify clusters.

While forming machine cells and part families, they do not take into consideration design, shape or other part characteristics. Burbidge [8] argued that these characteristics are not useful in GT; processing requirement for each part is the only information that is needed. He also argued that GT is suitable for all manufacturing systems. Citing personal consulting experiences, Burbidge mentioned there is sufficient flexibility in the part routing (or process planning) that allows PFA to be used to identify mutually separable machine cells and corresponding part families. However, a survey of many companies in the U.S. that use GT found that only in rare circumstances were companies able to identify mutually separable clusters of machine cells and part families [70]. Also, Wei [68] pointed out that the studies by Morris and Tersine [47] and Flynn and Jacobs [20] indicate that GT provides operational advantages only under certain conditions. Through a simulation study, Suresh [63] and [64] provided other conditions under which functional systems may be preferable to CMS's.

King and Nakornchai [34] used the following four categories:

1) *Similarity coefficient algorithms*: These are algorithms that are derived from numerical taxonomy and attempt to measure the similarity coefficient between each pair of machines or parts. For example, the similarity coefficient for a pair of machines may be given by the number of parts which visit both machines divided by the number of parts which visit at least one machine. Most of the similarity coefficient algorithms use the Jaccard similarity coefficient [59]. Examples of algorithms that measure the similarity between machines are, the single linkage clustering algorithm in [44], its modification in [53], the algorithm in [57], etc. Algorithms that measure the similarity between parts are presented in [10], [17], etc.
2) *Set-theoretic algorithms*: These techniques build supersets of machines and parts that can be represented as a path along the edges of a lattice diagram using the union operation. These techniques were first suggested by Purcheck [50] and later refinements can be found in [51] and [52]. King and Nakornchai [34] mention that the lattice diagram grows exponentially and hence such techniques will not be useful for practical problems where the number of parts and machines are typically large.
3) *Evaluative algorithms*: These techniques are basically the same as PFA techniques described above.
4) *Other analytical techniques*: These are techniques that were defined as clustering techniques in Section I. They examine the part-machine processing indicator matrix and attempt to rearrange its rows and columns so as to uncover clusters of machine cells and corresponding part families.

Han and Ham [25] classified the GT algorithms as follows:

1) *Peripatetic and ocular techniques*: These techniques are similar to the "rule-of-thumb" or "eyeballing" techniques described above. Knowledge concerning the parts and the manufacturing system are used in determining machine cells and part families. These methods are also not useful in practice as better techniques exist.
2) *PFA techniques.*
3) *Classification and coding techniques.*
4) *Mathematical programming techniques*: These techniques use fuzzy mathematics, pattern recognition, cluster analysis, etc. to identify part family/machine cell combinations.

Vakharia [65] used the following classification:

1) *Descriptive techniques*: Descriptive techniques include the PFA techniques discussed above and other component (or part) flow analysis (CFA) techniques discussed in [34]. King and Nakornchai [34] pointed out that there are a number of similarities between the PFA and CFA techniques. Some minor differences between the two are also noted in that paper.
2) *Block diagonal techniques*: These techniques are those that were previously defined as clustering techniques.
3) *Similarity coefficient techniques.*
4) *Other analytical techniques:* The set-theoretic based techniques such as the one discussed in King and Nakornchai [34] come under this classification.

Wemmerlov and Hyer [71] did an extensive analysis of some of the above classifications and also various techniques. They pointed out that most other classifications do not help us in understanding the process of cell formation. Hence they classified the GT techniques as follows:

1) *Techniques that identify part families without the help of machine routing.*
2) *Techniques that identify part families using routing.*
3) *Techniques that identify machine groups only.*
4) *Techniques that identify part families and machine groups simultaneously.*

The following categories were used to classify GT techniques in [39]:

1) *PFA techniques.*
2) *Similarity coefficient techniques.*
3) *Sorting-based techniques.*
4) *Bond-energy techniques.*
5) *Cost-based techniques.*
6) *Cluster-identification techniques.*

The sorting-based techniques, bond-energy techniques, and cluster identification techniques are specific clustering techniques. Cost-based techniques attempt to identify part family/machine cell combinations that minimize machining, setup, inventory and material handling costs.

In addition to the above, there are a few other authors that have classified GT techniques. These are listed in [71] and the interested reader is referred to that source for more details.

III. Design and Planning Issues in Cellular Manufacturing Systems

As mentioned in the previous section, a number of techniques have been developed for GT and CMS design. Most of them are primarily concerned with the identification of part families and machine cells and do not take into consideration other practical design constraints. These constraints are listed below in order of their importance. The list is an extension of the one in [27].

- Available capacity of machines must not be exceeded.
- Safety and technological requirements must be met.
- Number of machines in a cell and number of cells must not exceed an upper bound.
- Inter-cell and intra-cell cost of handling material between machines must be minimized.
- Machine utilization rate must be as high as possible.
- Machine purchase and operating costs and work-in-process inventory costs must be minimized.

In addition to the above design constraints, there are other planning issues that need to be addressed [65]. Two of the more important ones include

a) scheduling of jobs in individual cells; and
b) throughput rate maximization.

A. Machine Capacity

It is obvious that machine capacity is more important than the other constraints. It is therefore necessary to first ensure that adequate capacity (in machine hours) is available to process all the parts. Thus, the number of each type of machine must be known *a priori*. Otherwise, it must be determined. For this purpose, we use the model presented in [28]. The model and notation used are provided below.

- type of operations to be performed on the various parts; $i = 1, \cdots, p$
- types of machines available; $j = 1, \cdots, m$
- types of parts to be manufactured; $k = 1, \cdots, n$
- number of units of part type k to be manufactured; $NP_k, k = 1, \cdots, n$
- cost of performing operation i on part type k using machine type j; $c_{ijk}, i = 1, \cdots, p, j = 1, \cdots, m, k = 1, \cdots, n$
- time required to perform operation i on part type k using machine type j; $t_{ijk}, i = 1, \cdots, p, j = 1, \cdots, m, k = 1, \cdots, n$
- number of machines of type j selected; $NM_j, j = 1, \cdots, m$
- purchase cost of machine type j prorated over the planning period; $CM_j, j = 1, \cdots, m$
- time for which machine type j is available; $\tau_j, j = 1, \cdots, m$

Model

$$\text{Minimize} \quad \sum_{i=1}^{p}\sum_{j=1}^{m}\sum_{k=1}^{n} x_{ijk}c_{ijk} + \sum_{j=1}^{m} CM_j NM_j$$

$$\text{subject to} \quad \sum_{j=1}^{m} x_{ijk} = NP_k \quad \text{for each } i, k$$

$$\sum_{i=1}^{p}\sum_{k=1}^{n} x_{ijk}t_{ijk} \le \tau_j NM_j \quad \text{for each } j$$

$$x_{ijk}, NM_j \ge 0 \text{ and integer for each } i, j, k.$$

x_{ijk} in the above model may be used to detemine the number of times operation i is to be performed on part type k using machine type j. Note that it is implicitly assumed that each machine type can perform all the operations on each part type. If a machine type j cannot perform an operation i on part type k, then that particular x_{ijk} can be set to 0. This will decrease the number of x_{ijk} variables. The objective function minimizes the machine purchase and operating costs. The first constraint ensures that the required number of each operation is performed on the machines. The second constraint is a capacity constraint and ensures that the time available on each machine type is not exceeded. The last constraint is obvious and does not need any explanation.

The above model only ensures that the required number of each machine type is available. While allocating machines to cells, it is equally important to verify that there is adequate capacity in each cell to completely process all the part families assigned to it. To explain this, assume that

a) Two units of machine M_5 are available and that these two machines process parts P_1, P_4 and P_5 only. (There is just enough capacity on these two machines to process the three parts).
b) High volume parts P_1 and P_4 are in part family PF_1, whereas low volume part P_5 is in part family PF_2.
c) One unit of machine M_5 is in machine cells MC_1 and MC_2.

Since there is just enough capacity on the two units of machine M_5 to process all the three parts, the capacity constraint in cell MC_1 will be violated as a result of the above grouping because the two high volume parts P_1 and P_4 are processed on only one unit of machine M_5. Hence, because of the capacity limitation it may be necessary to process parts P_1 and P_4 on the other machine M_5 in cell MC_2. This will obviously require some units of part P_1 or P_4 to visit the two cells. This hypothetical example illustrates the fact that while the number of pieces of each type of equipment may be sufficient to process all the parts, the machine grouping may be such that there is not enough capacity to process one or more parts entirely within their corresponding cells. Thus, it is necessary to ensure that the capacity constraint in each cell is not violated while allocating machines to cells and parts to part families.

B. Safety and Technological Considerations

Safety or technological considerations may dictate two or more machines to be placed in the same cell, regardless of the number of parts visiting the two machines. An example would be the forging and heat-treatment stations. Due to fire hazards, these two stations must be placed in the same cell even if the number of parts visiting these two stations are relatively small in number. Conversely, two or more work-stations cannot be placed in the same cell, even if a large number of parts are processed by the two work-stations. The painting and welding stations are two such stations. Because of the sparks generated in the welding station, and the possibility of sparks igniting flammable solvents in the painting station, it is desirable to locate the welding and painting stations in different cells, as far apart as possible, although there may be a high interaction between the two stations. Such factors must be considered while allocating each machine to a nonempty cell.

C. Upper Bound on the Number of Machine Cells and Size of a Cell

It has been pointed out that in some CMS's, a set of multi-functional workers may be assigned to oversee the operations of a certain number of cells in order to improve the utilization of employees [6]. Since the availability of such cross trained workers is limited, it is necessary to impose an upper bound on the number of cells.

In many CMS's, there is also an upper bound on the number of machines that can be included in a cell. Askin and Chiu [1] mention that in practice, management can determine such upper bounds based on their experience. There are a number of reasons why such upper limits may be necessary. For example, for control purposes, it may be desirable to assign one operator to a machine cell. Since an operator can attend to a limited number of machines, it may be desirable to place an upper bound on the number of machines in a cell. Furthermore, floor plan dimensions may dictate the size of a cell in some GT problems [29]. In addition to upper bounds on the number of machines in a cell, Askin and Chiu [1] also impose a lower bound. Such a consideration may be necessary if it is desired to ensure that each operator is assigned a minimum workload.

D. Minimization of Inter-Cellular and Intra-Cellular Material Handling Cost

In an ideal CMS, there will be no flow of material between cells. But a number of researchers have pointed out that for practical problems, it is difficult to form machine groups between which there is no flow of material. Given this practical limitation, we often seek to form groups so that the inter-cellular movement of parts in minimized. In fact, one of the main objective of most GT techniques is to do just that (see Fig. 2). An equally important factor that needs to be addressed is that of minimizing intra-cellular cost of transporting parts. Obviously, this requires the layout design to be done for each cell. Heragu and Kakuturi [29] have implemented a 3-stage approach in which machine cells (and corresponding part families) are identified first; layout of machines within each cell is determined next and layout of cells is determined in the third stage. It should be noted that if the maximum number of machines in a cell and maximum number of cells are less than 15, optimal techniques that enable us to solve the two layout problems exist.

Instead of a sequential approach, a simultaneous solution of the grouping and layout problems would be desirable but such problems are difficult to solve. A mathematical model that minimizes the inter-cell material handling costs in addition to other operating costs is presented in [1]. However, this technique does not determine the position of cells or machines within each cell.

A number of researchers, for example, Co and Araar [16], Harhalakis *et al.* [26] and Logendran [42] have recognized that operations sequence for each part must be considered in the formation of machine cells. For, if this is done, it will enable identification of machine cells so that inter-cellular movement of parts is minimized. For example, if a part has one intermediate operation that is performed in a secondary cell, and all the others in a primary cell, then this will necessitate two trips (for each batch) between the two cells. On the other hand, if only the last operation is performed in a secondary cell, then there is only one trip required. A technique to minimize inter-cellular movement of parts is discussed in [27]. Since the product mix changes from time to time and is not known *a priori*, Seifoddini [56] suggests a probabilistic model be used to determine the expected intercellular material handling cost and outlines a mechanism to capture this uncertainty. Representation of the part-machine processing data as shown in Fig. 6 is useful is recording operation sequence information.

E. Machine Utilization

Machine utilization is a very important issue that must be addressed especially when equipment/process selection decision is made. At later stages, the importance of this factor diminishes. This is because costs are already incurred as a result of purchasing equipment in some previous stage and hence there is not much to gain by ensuring high equipment utilization when making planning decisions.

F. Cost Minimization

In Fig. 5, we saw how a bottleneck machine (M_1) may prevent the block diagonal form from occuring. For problems with bottleneck machines, a number of authors have suggested duplication of bottleneck machines in the appropriate cells so that parts belonging to a part family are processed entirely within the cell. For the data in Fig. 5, if an additional unit of machine M_1 is place in cell MC_1 (which consists of machines M_2, M_3, M_5 and M_7), part P_2 may be entirely processed within cell MC_2. (This is illustrated in Fig. 7). However, the benefit of forming mutually separable clusters may be more than offset by the expense incurred as a result of purchasing additional units of bottleneck machines; or the addition of duplicate machines may increase the cell size constraint. Under such circumstances, a physical arrangement in which bottleneck machine(s) are placed between machine cells so that the corresponding part families may share the resource, may be an attractive alternative. A physical arrangement that allows the sharing of an expensive bottleneck machine M_1 between two part families is shown in Fig. 8. For illustration purposes, it is assumed that the first set of machines is served by a robot and the other set by an AGV in both figures.

The example considered above underscores the importance of considering machine procurement costs while designing CMS's. In addition to this cost, there are others that need to be considered. Examples are

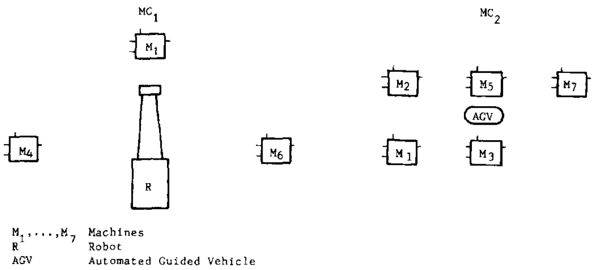


Fig. 7. Duplication of machine M_1 results in the formation of two smaller cells.

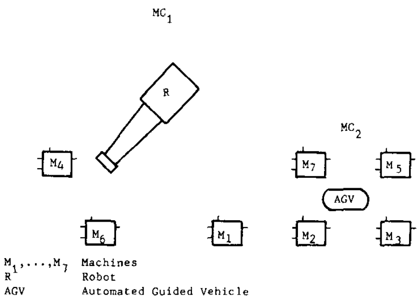


Fig. 8. Physical placement of machine M_1 allows two part families to share the machine.

•work-in-process inventory;
•machine depreciation; and
•machine setup.

If the above costs are considered while identifying machine cells and allocating part families to machine cells, the resulting solution will not only enable better control of the manufacturing system but also minimize operating costs significantly.

G. Scheduling of Jobs in Individual Cells

Vakharia [65] first recognized the need to consider scheduling issues while designing CMS's. He noted that since there are fewer jobs in each cell of CMS when compared to the entire system, solving scheduling problems optimally in a CMS may not be impossible.

H. Throughput Rate Maximization

The primary goal in any manufacturing system is to maximize throughput rate of parts. Maximization of throughput rate should be given more importance than maximization of machine utilization, especially if the machines have already been selected. Throughput rate can be maximized in a number of ways. For example, set-up time for each operation can be reduced. In fact, as a result of implementing GT, set-up times are reduced somewhat as similar parts are manufactured within a cell. If set-up times can be further reduced by redesigning parts or redesigning fixtures, this will enable us to increase the throughput rate even more.

IV. Literature Review of Recent Techniques for GT and CMS Design

In this section, we survey a number of papers published on GT and CM. Before surveying the papers, it is necessary to provide a broad classification of the papers. For purposes of literature review, these techniques are classified as

•clustering techniques; and
•CMS design techniques.

A. Clustering Techniques

In a broad sense, clustering techniques can be defined to be those techniques that, given an input object-object or object-attribute data matrix, attempt to uncover and display similar cluster or groups [45]. As discussed in Section I, they often do so by rearranging rows and columns of the input matrix. Clustering techniques have been applied in a number of areas and are listed in [39]. Since this paper deals with CMS's, clustering techniques are defined rather narrowly. They are defined as techniques that are concerned only with the identification of machine cells and/or corresponding part families; they neither take into account the design constraints listed in Section III nor consider cost factors. Also, most of the techniques discussed use process plan or part routing information in forming machine cells and/or part families. Such techniques have been classified in [71] as discussed in Section II. We modify that classification as shown below.

1) Techniques that identify part families only.
2) Techniques that identify machine cells only.
3) Techniques that identify part families and machine cells simultaneously.

In CMS design, it is clear that techniques belonging to the latter category are more useful than those belonging to the other two. This is because knowledge concerning machine grouping makes the task of the layout analyst easier. Part family identification simplifies the planning process [27].

Techniques that Identify Part Families Only To our knowledge, there are only a few techniques that come under this category. The first one uses a similarity coefficient which specifies the similarity between parts and was developed by Carrie [10]. The similarity coefficient for a part pair $\{i, j\}$ is determined by the ratio of the number of machines visited by parts i and j to the nubmer of machines visited by i or j. This coefficient is somewhat similar to the one used in [44]. Carrie [10] also developed a clustering technique to identify part families. This technique adds a part to a part family provided there is an acceptable level of similarity between the two. The technique incorporates an additional constraint that specifies the minimum number of parts allowed per family. Carrie [10] demonstrated how the same principles can be applied to develop plant layout.

Han and Ham [25] developed a technique that identifies part families only. However, the technique does not use part routing information in generating part families. Rather, it uses a classification and coding system which specifies part design and manufacturing characteristics. The part family identification problem is formulated as an integer programming problem and the absolute Minkowski metric is used to specify the distance between two parts. The objective function of the model lexicographically minimizes the distance between a part and part family. The model is then solved using the goal programming method.

Offodile [49] applied the similarity coefficient method to parts coding and classification analysis. In his approach, a part-part similarity matrix is constructed and used to identify part families. First, the largest element in the matrix is identified and the corresponding row, column indicate the parts to be included in the first part family. Then the rows, columns corresponding to the parts in the just formed part family are replaced by a single row and column indicating the maximum similarity of every other part (that has not been included in any part family) with the parts in existing part families. The resulting matrix is used to determine whether a new part family is to be formed or whether a new part is to be added to an existing part family. This is repeated until each part is added to a part family. Kini *et al.* [35] also presented a new approach to part numbering that has applications in GT.

Currie [17] presented a technique that uses a self-organizing neural network model—the interactive activation and competition (IAC) model, that identifies part families by considering design and manufacturing similarities simultaneously. A part-part similarity matrix is constructed and a bond energy algorithm is used to identify part families. Machine cells are identified only after the part families are formed and they done simply by examining the machines required to process parts in each family. Thus, although machine cells are identified, this algorithm is discussed in this section because its emphasis is on identifying part families only.

Techniques that Identify Machine Cells Only Some of the techniques discussed below, for example, the algorithms in [18] and [57], identify machine cells first and then part families based on the cells identified. Such techniques are also discussed in this section because their primary focus is on identifying machine cells.

McAuley [44] used a similarity coefficient applied in numerical taxonomy and constructed a similarity coefficient matrix $[s_{jk}]$, where s_{jk} is the similarity coefficient for a pair of machines $\{i, j\}$; s_{jk} is the ratio of the number parts that visit both the machines to the number of parts that visit at least one of these two machines. Notice the similarity between this coefficient and the one used in [10]. Whereas the coefficient above is defined for machine pairs, the similarity coefficient used in [10] deals with part pairs.

The single-linkage clustering technique developed by McAuley [44] adds a new machine to an existing cell if the similarity coefficient between the new machines and any existing machine in the cell exceeds a certain threshold level. An obvious disadvantage of this method is that a machine which has a high similarity coefficient with just one other machine already in the cell will automatically be included (in it) even if the similarity coefficient between the new machine and all other machines in the cell is very low. This is referred to as the chaining problem. Some of the deficiencies were remedied by Rajagopalan and Batra [53] who developed a graph theoretic method. Since their method considers some additional design constraints, it is discussed in a later section.

Since the methods in [44] and [53] do not always work well, de Witte [18] proposed another approach in which machine types were divided into three categories:

a) primary (if only one machine of that type is available or if all machines of that type have to be allocated to a single cell);
b) secondary (if more than one machine of that type is available); and
c) tertiary (if enough number of machines of that type is available so this machine type can be allocated to all cells).

In order to analyze the relationship between the three machine types, de Witt [18] used three similarity coefficients. He identified appropriate clusters (or cells). In the last two steps, he allocated corresponding parts to the cells and refined the solution further.

Faber and Carter [19] present a graph theoretical approach in which a machine-machine binary similarity matrix is constructed using a similarity measure. Then a similarity graph (with nodes indicating machines and arcs between machine pairs only if the machine pair has a 1 value in the binary matrix) is drawn. The densest subgraph of this graph is found by solving a min-cut max-flow problem. Machine cells are identified using the densest subgraph by means of a polynomial time algorithm. The technique consider upper bounds on cell size.

The single-linkage clustering algorithm in [44] was modified to overcome the chaining problem by Seifoddini and Wolfe [57]. They also used special data storage and analysis techniques to improve the algorithm's computation performance. Further, some strategies dealing with the duplication of bottleneck machines are incorporated.

Srinivasan *et al.* [61] proposed an assignment based algorithm that identifies machine groups first and then part families. An assignment problem is solved for a machine–machine similarity coefficient matrix. This results in a number of closed loops each corresponding to a machine cell. Parts visiting these cells form a part family. If a disjoint set of part families can be identified, the algorithm stops. If not, it checks whether two machine groups can be merged into one so as to identify a disjoint set of part families. If this is not possible, an assignment problem is solved for a part-part similarity coefficient matrix. The resulting part families are then assigned to machine cells so as to minimize the number of parts visiting multiple cells. This algorithm was further improved in [60].

In addition to the above techniques there are some others that attempt to identify machine cells or groups only. Some of these are discussed in [71].

Techniques that Identify Machine Cells and Part Families Simultaneously There are a number of a techniques that fall under this category. Iri [30] presented a clustering algorithm which starting from any row, masks all columns having a 1 entry in that row. It then masks all rows which have 1 entries in the masked columns. This procedure is repeated until it is not possible to go to new unmasked rows or columns. When this condition occurs, a cluster of machines and corresponding part families is formed. The above procedure is repeated to identify other clusters. Kusiak and Chow [40] presented essentially the same algorithm. Instead of masking rows and columns, they draw horizontal and vertical lines. Modified versions of this basic technique can be found in [39] and [37].

McCormick *et al.* [45] presented a heuristic bond-energy algorithm (BEA) that attempts to maximize the sum of the bond energies for each element $\{i,j\}$ in the part-machine processing indicator matrix $[a_{ij}]$. The bond-energy for element $\{i,j\}$ is given by:

$$a_{ij}[a_{ij+1} + a_{ij-1} + a_{i+1j} + a_{i-1j}].$$

The heuristic technique attempts (in a single pass for the row and column) to determine a permutation of rows and columns of the part-machine processing indicator matrix $[a_{ij}]$ so as to maximize the sum of the bond energies. If a block diagonal structure exists, it is immediately identified. However, if there are one or more bottleneck elements that prevent the formation of such a block diagonal structure, then the technique does not perform well. Gongaware and Ham [21] developed a modified and improved version of the original BEA that eliminated a tedious manual sorting procedure and outperformed the latter [71].

King [33] and King and Nakornchai [34] developed rank-order clustering (ROC) algorithms which determine a "binary value" for each row and column, rearrange rows and columns in descending order of their binary values and then identify clusters. The ROC algorithms have certain deficiencies. For example, the final, stable solution produced by the ROC algorithm is not necessarily the best one [34]. Chandrasekharan and Rajagopalan [13] developed a modified version of the algorithm called MODROC which attempts to overcome some of the deficiencies of ROC. MODROC enables an "objective" identification of machine cells.

Chandrasekharan and Rajagopalan [14] presented a graph theoretic formulation of the clustering problem. They developed expressions for a) the upper limit on the number of groups, and b) grouping efficiency that allows comparison of clustering algorithms. In addition, they presented a 3-stage technique to determine clusters. This method was extended, improved and is presented in [12].

A graph theoretic approach in which the grouping problem is modeled as a graph partitioning problem was presented in [36]. The two-phase approach involves obtaining a good initial solution (graph partition) by solving a transportation problem. In phase 2, another algorithm is used to improve the initial solution. Another network based method may be found in [67].

Askin *et al.* [2] presented a 3-stage hamiltonian path approach for solving the clustering problem. In stage 1, they compute the distance matrix for parts and machines, respectively. Using each distance matrix, they suboptimally solve an associated traveling salesman problem (TSP). This is done in stage 2. In stage 3, the tours of stage 2 are used to find a suboptimal solution to the associated hamiltonian path problem. This method was shown to be superior to the ROC algorithm and its extension using test problems.

Boe and Cheng [7] presented a close neighbor algorithm which rearranges a closeness matrix (the matrix indicates closeness between each pair of machines based on the parts they process) so as to identify clusters of machines. Then the part-machine processing indicator matrix is rearranged such that adjacent rows correspond to machines in a cluster. This gives a preliminary block diagonal form which is further improved using another heuristic procedure. The algorithm's performance was compared with that of a number of other clustering algorithms and found to be superior with respect to solution quality and computation time.

Kaparthi and Suresh [31] and Kusiak and Chung [38] have applied the Carpenter-Grossberg neural network to determine the part family/machine cell combinations. This network has two layers of neurons with each neuron in the input layer having a top-down and bottom-up connection to each neuron in the output layer. The input layer serves as a comparison layer while the output layer is a recognition layer. The basic procedure is an follows. When the first set of data, for example, a row of the part-machine processing indicator matrix, is provided to the network, it is stored as a representative vector for the first cluster. The subsequent sets of data are also read one row at a time, and compared to existing representative vectors. They are either treated as being part of existing clusters (in which case, the corresponding representative vector is suitably updated) or are stored as representative vectors of new clusters. A key parameter called the vigilance parameter dictates whether the input vector is to be stored as a representative vector or whether it is to be treated as being part of existing clusters. Different values of the vigilance parameter generate different clusters.

B. CMS Design Techniques

CMS design techniques are those that not only attempt to find mutually separable clusters of machine cells (and corresponding part families) but also incorporate practical facility design constraints. Such techniques can be further classified into

a) cost based techniques; and
b) design constraint based techniques.

Cost Based Techniques Cost based techniques explicitly consider the various costs listed in Section III while identifying machine cells and corresponding part families. There are relatively fewer techniques in this category and are presented in this section. Although some of the techniques consider design constraints also, they are discussed in this section because their primary emphasis is on forming machine cells such that various costs are minimized.

One of the first models that explicitly considered costs while forming machine cells was presented by Chakravarty and Shtub [11]. However, their model only considers work-in-process inventory cost; further, it assumes that only one machine of each type is available.

The cost model presented [3] considers fixed machine costs, variable processing costs, set-up, cycle inventory, and intracellular material handling costs. The GT problem is solved using a 3-stage approach. This approach is not suitable for large problems because of its complexity [1].

Choobineh [15] also presented a two-stage approach in which part families are determined in the first stage. In the second stage, a mathematical model is used to identify the corresponding machine cells so that the equipment purchase, part production, work-in-process and set-up costs are minimized. Machine capacity constraints are also included. Further, the desired number of cells can be provided as input to the model. The author suggests that this number be kept small; otherwise the model cannot be solved easily.

Three mixed-integer programming models that assume availability of alternate process plans are provided in [55]. The first one provides information to construct a part-machine processing indicator matrix which can then be solved using existing clustering techniques. The second forms cells assuming part families are known whereas the third identifies part families and machine cells simultaneously. The models minimize machine investment and operating costs. But as the authors point out, the models are not suitable for industrial problems because of the presence of integer variables. Another model that considers setup costs and setup time in addition to investment costs is presented in [54]. Again, this model is appropriate only in environments where relatively few parts are manufactured.

Askin and Chiu [1] developed a model that considers machine depreciation, inventory, material handling and set-up costs. The cost model is first divided into two subproblems and solved suboptimally using graph partitioning techniques. The first subproblem determines the assignment of parts to machines, while the second one determines the assignment of machines to cells. This approach was shown to be not only flexible but also successful for larger realistic problems.

Design Constraint Based Techniques To improve the performance of the single-linkage clustering algorithm in [44], Rajagopalan amd Batra [53] formulated the machine grouping problem as a graph-theoretic model and applied a partitioning algorithm to identify clusters. First, a graph with vertices representing machines is drawn. The vertices are initially unconnected. The technique then connects the vertices $\{i, j\}$ only if the similarity coefficient between machines i and j is greater than a threshold value. When this procedure is completed, a clique consisting of several vertices, or alternately, a machine cell consisting of several machines, is formed. Additional factors such as, imposing an upper limit on machine cell size, maintaining an even and high utilization of the machines in each cell, and determination of the number of machines of a given type in each cell, are incorporated in the technique.

Purcheck [50]–[52] presented a lattice theoretic method which was previously discussed under set-theoretic methods. Their method maximized scheduling flexibility and minimized the total cost of forming machine cells while determining the machine-part grouping.

Ballakur and Steudel [4] presented a heuristic technique for identifying part family/machine cell combinations simultaneously. The technique adds machines to a cell based on cell work load and cell size constraints. The assignment of a part to a part family is done so that a majority of the operations on the part are performed within the cell corresponding to the part family to which the part is assigned. Thus the technique indirectly attempts to minimize the inter-cellular movement of parts. Its performance was compared with other well known clustering techniques and in all cases produces solutions that were at least as good as the ones reported in the literature. The technique is flexible as it allows the user to change two key parameters—cell admission factor (which controls the admission of a machine to a cell) and cell size factor. Thus a number of alternative solutions can be generated for various values of these two parameters.

In addition to the above there are a number of other techniques that address only one or two design constraints listed in Section III. Some of them are briefly reviewed here. Co and Araar [16] presented a hierarchical procedure for configuring CMS's that involves three steps. The only design constraint discussed in this paper is the capacity constraint. In the first step, a mathematical model is used to determine the assignment of jobs to machines; the objective is to minimize deviation between workload assigned to machines and the available capacity. Solution of the model is used to construct a binary job-machine processing indicator matrix. In the second stage, an extension of King's ROC algorithm [33] is used to identify clusters of machines and corresponding jobs. In stage 3, a direct search algorithm is employed to determine the number of cells as well as the composition of each. The objective of this algorithm is a) to minimize the number of uncompleted jobs in each cell and b) to maximize the number of cells. Note that the first objective minimizes the number of cells visited by each job and hence indirectly puts more machines in each cell, whereas the second objective attempts to do the opposite. Co and Araar [16] discussed a simple procedure for achieving this; it involves applying two key steps repeatedly until all machines are assigned to cells.

Gupta and Seifoddini [22] presented a similarity coefficient that considers part processing requirement, production volume, routing sequence and operation time in determining the coefficient for each machine-pair. They also presented clustering techniques that identify machine cells and part families so that the design constraints mentioned in Section III are satisfied to the extent possible.

A heuristic that attempts to minimize inter-cellular movement of parts is discussed in [26]. It is a two-stage algorithm in which a bottom-up aggregation procedure is used to minimize inter-cellular movement of parts. The second step is a local refinement procedure that attempts to improve the first stage solution. Nagi *et al.* [48] formulated the GT problem as a linear programming problem which addresses two subproblems: a) routing selection and b) cell formation. The algorithm in [26] is used to solve the second subproblem. The solution approach iterates between the two subproblems.

As was previously mentioned, the sequence of operations is of utmost importance in determining the flow of material between machines and machine cells. Logendran [42] presented a four phase approach that was an improvement over his previous algorithm in [43] and incorporates machine utilization considerations. In the first phase, a certain number of key machines, one for each cell, is selected based on workload consideration. The number of cells is predetermined. The key machines are selected so that they are dissimilar with respect to the parts they process. The remaining machines are added, one at a time, to a cell that results in minimum total move. An improvement algorithm is used to improve the solution in phase 2, if possible. Finally, parts are assigned to their respective cells based on the cumulative processing time.

Heragu [27] presented an algorithm that considers many of the constraints listed in Section III. This algorithm identifies a preliminary set of mutually separable machine groups and part families (simultaneously) in such a way that safety and technological constraints are satisfied. Then it ascertains whether the number of cells formed is less than the upper limit specified by the user. If this constraint is violated, it attempts to group two cells at a time so that cells having the greatest interaction are combined into one. This is repeated until the number of cells do not exceed the upper bound. Next, the algorithm determines if the cells size constraint has been violated for any of the cells formed. If so, it attempts to identify bottleneck machines. If a bottleneck machine (which costs less than a threshold value set by the user) cannot be identified, the algorithm determines if there are two bottleneck machines. If bottleneck machines cannot be identified, it forms smaller cells of the required size by retaining the required number of machines and placing the remaining in a temporary cell. However, the machines retained are the ones that together have the maximum flow between them. The above procedure is repeated for machines in the temporary cell also. In a final step, the algorithm determines if any part visits multiple cells. For such parts, it further determines if the processing is done on only one machine in any of the cells. If so, it duplicates that machine in another cell visited by the part. Heragu and Gupta [28] provide a modified version of this algorithm that also incorporates capacity considerations.

A three stage approach aimed at identifying not only machine cells (and corresponding part families) but also determining a layout of machines within each cell and the cells themselves is presented in [29]. Using real world data, Heragu and Kakuturi [29] demonstrate that the three-stage approach which considers practical design constraints can be used for industrial prolems. For performing the machine grouping, the technique in [28] is modified significantly and used. For determining a layout of cells and machines within each cell, simulated annealing based algorithms are used. The approach allows the user to change key data items and thereby perform some sensitivity analysis. The algorithm's solution quality and performance are demonstrated using two grouping measures. It is shown to produce good quality solution without requiring excessive computation time.

In addition to the above, there are other methods that explicitly consider some of the design constraints listed in section III. Examples are Wei and Gaither [69], Vakharia and Wemmerlov [66], Sule [62].

V. Conclusion

Thus far, research on the GT and CMS design problem has focused primarily on identifying mutually separable part family/machine cell combinations without considering practical design constraints that are frequently encountered. However, there have been some efforts toward incorporating design constraints in the past few years. In this paper, some of the more important ones were listed and discussed. In addition, some planning considerations were also discussed. A number of papers on GT and CMS design were surveyed. A new classification scheme was also provided. Since the last survey paper was published, there have been a number of algorithms that have been developed for the practical CMS design problem. It appears that research on clustering algorithms (as defined in section I) has reached a saturation point and further efforts must be focused on using existing efficient clustering algorithms to solve CMS design problem that incorporate important design constraints. We feel that such techniques will find more use in practice.

References

[1] R. G. Askin and K. S. Chiu, "A graph partitioning procedure for machine assignment and cell formation in group technology," *Int. J. Production Res.*, vol. 28, no. 8, pp. 1555–1572, 1990.

[2] R. G. Askin, S. H. Creswell, J. B. Goldberg and A. J. Vakharia "A Hamiltonian path approach to reordering the part-machine matrix for cellular manufacturing," *Int. J. Production Res.*, vol. 29, pp. 1081–1100, 1991.

[3] R. G. Askin and S. P. Subramanian, "A cost based heuristic for group technology configuration," *Int. J. Production Res.*, vol. 25, pp. 101–113, 1987.

[4] A. Ballakur and H. J. Steudel, "A within-cell utilization based heuristic for designing cellular manufacturing systems," *Int. J. Production Res.*, vol. 25, no. 5, pp. 639–665, 1987.

[5] D. Ben-Arieh and E. Triantaphyllou, "Quantifying data for group technology with weighted fuzzy features," *Int. J. Production Res.*, vol. 30, no. 6, pp. 1285–1299, 1992.

[6] J. T. Black, "Cellular manufacturing systems reduce setup time, make small lot production economical," *Indust. Eng.*, Nov., pp. 36–48, 1983.

[7] W. J. Boe and C. H. Cheng, "A close neighbor algorithm for designing cellular manufacturing systems," *Int. J. Production Res.*, vol. 29, no. 10, pp. 2097–2116, 1991.

[8] J. L. Burbidge, "Change to group technology: Process organization is obsolete," *Int. J. Production Res.*, vol. 30, no. 5, pp. 1209–1219, 1992.

[9] J. L. Burbidge, "Production flow analysis," *The Production Engineer*, vol. 42, no. 12, pp. 742, 1963.

[10] A. S. Carrie, "Numerical taxonomy applied to group technology and plant layout," *Int. J. Production Res.*, vol. 11, no. 4, pp. 399–416, 1973.

[11] A. K. Chakravarty and A. Shtub, "An integrated layout for group technology with in-process inventory cost," *Int. J. Production Res.*, vol. 22, pp. 431–442, 1984.

[12] M. P. Chandrasekharan and R. Rajagopalan, "ZODIAC: An algorithm for concurrent formation of part-families and machine-cells," *Int. J. Production Res.*, vol. 25, no. 6, pp. 835–850, 1987.

[13] M. P. Chandrasekharan and R. Rajagopalan, "MODROC: An extension of rank order clustering for group technology," *Int. J. Production Res.*, vol. 24, no. 5, pp. 1221–1233, 1986.

[14] M. P. Chandrasekharan and R. Rajagopalan, "An ideal seed non-hierarchical clustering algorithm for cellular manufacturing," *Int. J. Production Res.*, vol. 24, no. 2, pp. 451–464, 1986.

[15] F. Choobineh, "A framework for the design of cellular manufacturing systems," *Int. J. Production Res.*, vol. 26, pp. 1161–1172. 1988.

[16] H. C. Co and A. Araar, "Configuring cellular manufacturing systems," *Int. J. Produciton Res.*, vol. 26, no. 9, pp. 1511–1522, 1980.

[17] K. R. Currie, "An intelligent grouping algorithm for cellular manufacturing," presented at 14th Ann. Computers and Industrial Engineering Conf., Orlando, FL, March 9–11, 1992.

[18] J. de Witte, "The use of similarity coefficients in production flow analysis," *Int. J. Production Res.*, vol. 18, no. 4, pp. 503–514, 1980.

[19] Z. Faber and M. W. Carter, "A new graph theoretic approach for forming machine cells in cellular production systems," in *Flexible Manufacturing Systems: Methods and Studies*, A. Kusiak, ed. New York: North Holland, pp. 301–318, 1986.

[20] B. B. Flynn and F. R. Jacobs, "A simulation comparison of group technology with traditional jobshop manufacturing," *Int. J. Production Res.*, vol. 24, pp. 1171–1192, 1986.

[21] T. Gongaware and I. Ham, "Cluster analysis applications for group technology manufacturing systems," in *Proc. Ninth Amer. Metal Working Res. Conf.*, Society of Manufacturing Engineers, 1981.

[22] T. Gupta and H. Seifoddini, "Production data based similarity coefficient in the design of a cellular manufacturing system," *Int. J. Production Res.*, vol. 28, pp. 1247–1269, 1990.

[23] I. Ham, "Introduction to Group Technology," SME Tech. Paper MMR 76–093, Society of Manufacturing Engineers, Dearborn, MI, 1976.

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

[25] C. Han and I. Ham, "Multiobjective cluster analysis for part family formations," *J. Manufacturing Systs.*, vol. 5, no. 4, pp. 223–230, 1986.

[26] G. Harhalakis, R. Nagi and J. M. Proth, "An efficient heuristic in manufacturing cell formation for group technology applications," *Int. J. Production Res.*, vol. 28, no. 1, pp. 185–198, 1990.

[27] S. S. Heragu, "A heuristic algorithm for identifying machine cells," *Information and Decision Technologies*, vol. 18, pp. 171–184, 1992.

[28] S. S. Heragu and Y. P. Gupta, "A heuristic for designing cellular manufacturing facilities," *Int. J. Production Res.*, vol. 32, no. 1, pp. 125–140, 1994.

[29] S. S. Heragu and S. R. Kakuturi, "Grouping and placement of machine cells," DSES Tech. Rep. #37-92-341, Rensselaer Polytechnic Inst., Troy, NY 12180, 1992.

[30] M. Iri, "On the synthesis of loop and cutset matrices and related problems," in *RAAG Memoirs*, K. Kondo, ed., Research Association of Applied Geometry, Japan, 4 (A-XII), pp. 376–410, 1968.

[31] S. Kaparthi and N. C. Suresh, "Machine-component cell formation in group technology: A neural network approach," *Int. J. Production Res.*, vol. 30, no. 6, pp. 1353–1367, 1990.

[32] G. M. Kern and J. C. Wei, "The cost of eliminating exceptional elements in group technology cell formation," *Int. J. Production Res.*, vol. 29, no. 8, pp. 1535–1547, 1980.

[33] J. R. King, "Machine-component grouping in production flow analysis: An approach using a rank order clustering algorithm," *Int. J. Production Res.*, vol. 18, pp. 213–219, 1980.

[34] J. R. King and V. Nakornchai, "Machine-component group formation in group technology: Review and extension," *Int. J. Production Res.*, vol. 20, no. 2, pp. 117–133, 1982.

[35] R. B. Kini, L. R. Taube, and C. T. Mosier, "Part identification and group technology: A new approach," *J. Manufacturing Systs.*, vol. 10, no. 2, pp. 134–145, 1991.

[36] K. R. Kumar, A. Kusiak and A. Vannelli, "Grouping of parts and components in flexible manufacturing systems," *European J. Op. Res.*, vol. 24, pp. 387–397, 1986.

[37] A. Kusiak, "Branching algorithms for solving the group technology problem," *J. Manufacturing Systs.*, vol. 10, no. 4, pp. 332–343, 1991.

[38] A. Kusiak and Y. K. Chung, "GT/ART: Using neural networks to form machine cells," *Manufacturing Review*, vol. 4, no. 4, pp. 293–301, 1991.

[39] A. Kusiak and W. S. Chow, "Decomposition of manufacturing systems," *IEEE Trans. Robotics and Automation*, vol. 4, no. 5, pp. 457–471, 1988.

[40] A. Kusiak and W. S. Chow, "An efficient cluster identification algorithm," *IEEE Trans. Syst., Man, Cybern.*, vol SMC-17, no. 4, pp. 696–699, 1987.

[41] A. Kusiak and S. S. Heragu, "Group technology," *Computers in Industry*, vol. 9, no. 2, pp. 83–91, 1987.

[42] R. Logendran, "Impact of sequence of operations and layout of cells in cellular manufacturing," *Int. J. Production Res.*, vol. 29, no. 2, pp. 375–390, 1991.

[43] R. Logendran, "A workload based model for minimizing total intercell and intracell moves in cellular manufacturing," *Int. J. Production Res.*, vol. 28, no. 5, pp. 913–925, 1990.

[44] J. McAuley, "Machine grouping for efficient production," *The Production Engineer*, vol. 51, no. 2, pp. 53–57, 1972.

[45] W. T. McCormick, P. J. Schweitzer, and T. W. White, "Problem decomposition and data reorganization by a clustering technique," *Op. Res.*, vol. 20, pp. 992–1009, 1972.

[46] S. P. Mitrofanov, *Group Technology in Industry*, vols. 1 and 2, Leningrad, USSR: Mashinostroienie, (in Russian) 1983.

[47] J. S. Morris and R. J. Tersine, "A simulation analysis of factors influencing the attractiveness of group technology and cellular layouts," *Management Science*, vol. 36, no. 12, pp. 1567–1578, 1990.

[48] R. Nagi, G. Harhalakis and J. M. Proth, "Multiple routeings and capacity considerations in group technology applications," *Int. J. Production Res.*, vol. 28, no. 12, pp. 2243–2257, 1990.

[49] O. F. Offodile, "Application of similarity coefficient method to parts coding and classification analysis in group technology." *J. Manufacturing Systs.*, vol. 10, no. 6, pp. 442–448, 1991.

[50] G. F. K. Purcheck, "Combinatorial grouping—a lattice-theoretic method for the design of manufacturing systems," *J. Cybern.*, vol. 4, no. 3, pp. 27–60, 1974.

[51] G. F. K. Purcheck, "A mathematical classification as a basis for the design of group technology production cells," *Production Engineer*, vol. 54, pp. 35–48, 1975.

[52] G. F. K. Purcheck, "A linear programming method for the combinatoric grouping of an incomplete power set," *J. Cybern.*, vol. 5, no. 4, pp. 51–76, 1975.

[53] R. Rajagopalan and J. L. Batra, "Design of cellular production systems—A graph theoretic approach," *Int. J. Production Res.*, vol. 13, no. 6, pp. 567–579, 1975.

[54] D. Rajamani, N. Singh and Y. P. Aneja, "A model for cell formation in manufacturing systems with sequence dependence," *Int. J. Production Res.*, vol. 30, no. 6, pp. 1227–1235, 1992.

[55] D. Rajamani, N. Singh and Y. P. Aneja, "Integrated design of cellular manufacturing systems in the presence of alternative process plans," *Int. J. Production Res.*, vol. 28, no. 8, pp. 1541–1554, 1990.

[56] H. Seifoddini, "A probabilistic model for machine cell formation," *J. Manufacturing Systs.*, vol. 9, no. 1, pp. 69–75, 1990.

[57] H. Seifoddini and P. M. Wolfe, "Application of the similarity coefficient method in group technology," *IIE Transactions*, pp. 271–277, 1987.

[58] S. M. Shafer, G. M. Kern and J. C. Wei, "A mathematical programming approach for dealing with exceptional elements," *Int. J. Production Res.*, vol. 30, no. 5, pp. 1029–1036, 1992.

[59] P. H. A. Sneath and R. R. Sokal, *Numerical Taxonomy*. San Francisco, CA: Freeman and Co., 1973.

[60] G. Srinivasan and T. T. Narendran, "GRAFICS—A nonhierarchical clustering algorithm for group Technology," *Int. J. Production Res.*, vol. 29, no. 3, pp. 463–478, 1991.

[61] G. Srinivasan, T. T. Narendran and B. Mahadevan, "An assignment model for the part-families problem in group technology," *Int. J. Production Res.*, vol. 28, no. 2, pp. 145–152, 1990.

[62] D. R. Sule, "Machine capacity planning in group technology," *Int. J. Production Res.*, vol. 29, no. 9, pp. 1909–1922, 1991.

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

[64] N. C. Suresh, "Partitioning work centers for group technology: Insights from an analytical model," *Decision Sciences*, vol. 22, pp. 772–791, 1991.

[65] A. J. Vakharia, "Methods of cell formation in group technology: A framework for evaluation," *J. Op. Management*, vol. 6, no. 3, pp. 257–271, 1986.

[66] A. J. Vakharia and U. Wemmerlov, "Designing a cellular manufacturing system: A materials flow approach based on operation sequence," *IIE Transactions*, vol. 22, no. 1, pp. 84–97, 1990.

[67] T. Vohra, D. Chen, J. C. Chang and H. C. Chen, "A Network Approach to Cell Formation in Cellular Manufacturing," *Int. J. Production Res.*, vol. 28, no. 11, pp. 2075–2084, 1990.

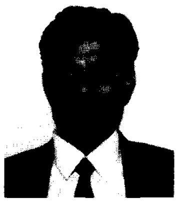
[68] J. C. Wei, "A note on 'Change to group technology,' " *Int. J. Production Res.*, vol. 30, no. 5, pp. 1221–1222, 1992.

[69] J. C. Wei and N. Gaither, "A capacity constrained multiobjective cell formation method," *J. Manufacturing Systs.*, vol. 9, no. 3, pp. 222–232, 1990.

[70] U. Wemmerlov and N. L. Hyer, "Cellular manufacturing in the U. S. industry: A survey of users," *Int. J. Production Res.*, vol. 27, no. 9, pp. 1511–1530, 1989.

[71] U. Wemmerlov and N. L. Hyer, "Procedures for the part-family/machine group identification problem in cellular manufacturing," *J. Op. Management*, vol. 6, no. 2, pp. 125–147, 1986.

[72] C. Zhang and H. Wang, "Concurrent formation of part families and machine cells based on the fuzzy set theory," *J. Manufacturing Systs.*, vol. 11, no. 1, pp. 61–67, 1992.



Sunderesh Heragu received the B.Eng. degree in mechanical engineering from the University of Mysore, India, in 1982; the M.B.A. degree from the University of Saskatchewan, Canada, in 1985; and the Ph.D. degree in industrial engineering from the University of Manitoba, Canada, in 1988.

He is an Assistant Professor in the Department of Decision Sciences and Engineering Systems at Rensselaer Polytechnic Institute. Previously he was Assistant Professor at the State University of New York, Plattsburgh, and Adjunct Professor at the University of Manitoba, Winnipeg, Canada. His current research interests are in the integration of design and planning activities in manufacturing systems. His previous research has focused on the application of knowledge-based systems in automated manufacturing systems, cellular manufacturing, facility location, layout, and network design. He has authored or coauthored over fifty articles and technical reports, many of which have appeared in journals, magazines, books, and conferences. He has edited a special issue of *European Journal of Operational Research* on facility layout and currently serves on the editorial board of *International Journal of Industrial Engineering*. He is a senior member of IIE and SME, and member of ORSA, POMS, and TIMS.