SOLVING COMPONENT FAMILY IDENTIFICATION PROBLEMS ON MANUFACTURING SHOP FLOOR

M. Modak, T. Ghosh and P. K. Dan
Department of Industrial Engineering & Management
West Bengal University of Technology
BF 142, Salt lake City, Kolkata, 700064, India
Email: tamal.31@gmail.com

ABSTRACT
This article demonstrates effective techniques for component/part family formation problem in the vicinity of cellular manufacturing system. Past investigations reported that part family formation techniques are typically grounded on production flow analysis (PFA) which largely considers operational requirements, sequences and time. Part coding analysis (PCA) is merely counted in cellular manufacturing which is assumed to be the most competent method to identify the part families. In present study different clustering techniques are quantified to develop proficient part families by utilizing Opitz part coding scheme and the techniques are tested on 5 different datasets of size (5×9) to (27×9) and the obtained results are compared with each other. The experimental results reported that the C-Linkage method is more effective in terms of the quality of the solution obtained and has outperformed SLCA and K-means techniques.

Keywords: Part family formation; Group technology; Cellular manufacturing; Similarity metric; Clustering algorithms

1. INTRODUCTION
Group technology (GT) in cellular manufacturing systems (CMS) is substantial in improving productivity for the manufacturing companies. Group technology (GT) might be considered as a simplified methodology which groups standardized similar entities such as parts, assemblies, process plans, tools, instructions, etc. to minimize the time and effort and to improve the overall productivity for batch type production (Burbidge, 1963). As reported in the literature (Guerrero et al., 2000) a successful implementation of GT could eventually minimize the engineering costs, facilitate cellular manufacturing, quicken product development, enhance costing accuracy, simplify process planning, minimize tooling cost and simplify the overall purchasing process. A major prerequisite in implementing GT is the identification of part families (Kaparthi and Suresh, 1991). A part family is a group of parts sharing homogeneous design and manufacturing attributes. Early research in this domain has been dedicated primarily on the formation of production-oriented part families in which similarities amongst the parts are principally recognized on the fact of their processing requirements, operation time and operation sequences. Though these methodologies are inadequate in achieving the needs of other extents of manufacturing. For example, parts with homogeneous shape, size, dimension or other design characteristics are believed to be clustered in a single family for design justification and elimination of part varieties; however, parts which are clustered on the fact of homogeneous routing and the tooling needs are convenient to resolve the process planning issues.

Therefore, the scope of this domain of research is believed to be expanded and examined to a wider span of part similarities. Part similarities are believed to be identified sooner than the formation of part families. Part attributes such as shape, length/diameter ratio, material type, part function, dimensions, tolerances, surface finishing, process, operations, machine tool, operation sequence, annual production quantity, fixtures needed, lot sizes have been considered as the basis for similarity utilization as specified by Groover and Zimmers (1984). Moon (1992) has stated that the complexity remains in acquiring an appropriate technique which provides an identifying competence of human being, such as identifying patterns in groups, and forming part families with the aid of intelligence.

This article utilized different part family identification
techniques based on Complete Linkage (C-Linkage), Single Linkage (SLCA) and K-means clustering algorithms to investigate the nature of similarities and to describe the effectiveness of the techniques in solving the problem in hand.

This article is further structured in following order. Section 2 explains a brief literature survey; Section 3 presents the problem definition and performance metric. Thereafter the employed methodologies are stated in section 4. Computational results and discussion are demonstrated in section 5 and section 6 is the conclusion of this research.

2. LITERATURE SURVEY

Two different approaches are traced in past literature in order to form part families, first is production flow analysis (PFA) which deals with processing requirements of parts, operational sequences and operational time of the parts on the machines. Second approach is the part coding analysis (PCA) which utilizes predefined coding schemes to facilitate the process using several attributes of parts such as geometrical shapes, materials, design features and functional requirements etc.

PCA is exposed in this study as an essential and effective tool for successful implementation of GT concept. A code may be numbers (numerical) or alphabets (alphabetical) or a hybridization of numbers and alphabets (alphanumerical) which are allotted to the parts to process the information. Parts are categorized based on significant attributes such as dimensions, type of material, tolerance, operations required, basic shapes, surface finishing etc. In this approach, each part is assigned a code which is a string of numerical digits that store information about the part. Singh and Rajamani (1996) demonstrated coding systems in their study. Generally coding systems depict either hierarchical structure (monocode), or chain structure (polycode) or hybrid mode structure mixed with monocode and polycode.

Several coding systems have been developed, e.g. Opitz, MICLASS, DCLASS and FORCOD (Jung and Ahluwalia, 1992). Han and Ham (1986) have claimed that part families could be established more realistically by practicing the PCA due to the advantage of using the manufacturing and design attributes concurrently. Offodile (1992) reported a similarity metric based on the numeric codes for any pair of parts which could be utilized to an appropriate clustering method such as agglomerative clustering algorithm to form efficient part families.

Clustering analysis is practiced in Cellular Manufacturing System (CMS) as a competent methodology to facilitate the machine/part grouping problems. Various machine/part grouping techniques are developed to solve manufacturing cell formation problems since last forty years, these include similarity coefficient methods, clustering analysis, array based techniques, graph partitioning methods etc. The similarity coefficient approach was first suggested by McAuley (1972). The basis of similarity coefficient methods is to calculate the similarity between each pair of machines and then to group the machines into cells based on their similarity measurements. Few studies have been proposed to measure dissimilarity coefficients instead of similarity coefficient for machine-part grouping problems (Prabhakaran et al., 2002). Most of the similarity coefficient methods utilized machine–part mapping chart. Few of them are Single linkage clustering algorithm (McAuley, 1972), Average linkage clustering algorithm etc (Seifoddini and Wolfe, 1986).

Array based methods consider the rows and columns of the machine-part incidence matrix as binary patterns and reconfigure them to obtain a block diagonal cluster formation. The rank order clustering algorithm is the most familiar array-based technique for cell formation (King, 1980). Substantial alterations and enhancements over rank order clustering algorithm have been described by King and Nakornchai (1982) and Chandrasekharan and Rajagopalan (1986). The direct clustering analysis (DCA) has been stated by Chan and Milner (1982), and bond energy analysis is performed by McCormick et al. (1972).

Graph Theoretic Approach depicts the machines as vertices and the similarity between machines as the weights on the arcs. Rajagopalan and Batra (1975) proposed the use of graph theory to form machine cells. Chandrasekharan and Rajagopalan (1986) proposed an ideal seed nonhierarchical clustering algorithm for cellular manufacturing. Graph searching algorithms was demonstrated by Ballakur and Steudel (1987).
which select a crucial machine or part according to a pre-fixed criterion. A non-heuristic network method was stated by Vohra et al. (1990) to construct manufacturing cells with minimum inter-cell moves. Srinivasan (1994) implemented a method using minimum spanning tree (MST) for the machine-part cell formation problem.

During past few decades soft computing techniques are exhaustively practiced by researchers in the vicinity of CMS. Lee-Post (2000) proposed that GT coding system (DCLASS) could be efficiently used with simple GA method to cluster part families which is well suited for part design and process planning in production. A hybrid methodology based on Boltzmann function from simulated annealing and mutation operator from GA was proposed by Wu et al. (2009) to optimize the initial cluster obtained from similarity coefficient method (SCM) and rank order clustering (ROC). Arkat et al. (2007) developed a sequential model based on SA for large-scale problems and compared their method with GA. Atene-Nguema and Dao (2007) investigated an ACO based TS heuristic for cellular system design problem (CSDP) and the methodology proved to be much quicker than traditional methods when considering operational sequence, time and cost. These Authors (2009) further proposed quantized Hopfield network for CFP to find optimal or near-optimal solution and TS was employed to improve the performance and the quality of solution of the network. Durán et al. (2010) reported a modified PSO with proportional likelihood instead of using velocity vector on CF problems where the objectives are the minimization of cell load variation and intercellular parts movement and reported the stability of the method with low variability. A similar study was also performed by Anvari et al. (2010) where a hybrid particle swarm optimization technique for CFP was reported. The initial solutions generated either randomly or using a diversification generation method and the technique also utilized mutation operator embedded in velocity update equation to avoid reaching local optimal solutions. Thereafter with due consideration, a wide variety of machine/part matrices were effectively solved by this approach.

A detailed study on metaheuristic based approaches in CMS could be obtained from Papaioannou and Wilson (2010) and Ghosh et al. (2011). These studies report that PCA has merely been adopted to form part families using the stated methodologies. Therefore this article would explore an unexplored area based on PCA in CMS by employing different clustering techniques.

3. PROBLEM DEFINITION

Opitz classification and coding system is used in this article which was developed by Opitz (1970) at Aachen Technology University in West Germany. The basic code comprises of nine digits that can be extended by additional four digits. The general interpretations of the nine digits are as indicated in Fig. 1. The first 5 digits are called the form code and designate the design or the general form of the part and hence aid in design retrieval. Later, 4 more digits were added to the coding scheme in order to enhance the manufacturing information of the specific work part. These last four digits are also called supplementary code. All four digits are integers, and respectively represent: Dimensions, Material, Original shape of raw stock, and Accuracy of the work part. The extra four digits, A, B, C, and D (not shown in Fig. 1) called the secondary code, are practiced by the particular organization to include those features that are specific to their organization.

The interpretation of first 9 digits are:

Digit 1: General shape of workpiece, otherwise called ‘part-class’. This is further subdivided into rotational and non-rotational classes and further divided by size (length/diameter or length/width ratio.)

Digit 2: External shapes and relevant form. Features are recognized as stepped, conical, straight contours. Threads and grooves are also important.

Digit 3: Internal shapes. Features are solid, bored, straight or bored in stepped diameter. Threads and grooves are integral part.

Digit 4: Surface plane machining, such as internal or external curved surfaces, slots, splines.

Digit 5: Auxiliary holes and gear teeth.

Digit 6: Diameter or length of workpiece.

Digit 7: Material Used.

Digit 8: Shape of raw materials, such as round bar, sheet metal, casting, tubing etc.

Digit 9: Workpiece accuracy.
All the 9 digits are interpreted numerically (0-9). Examples of a mild steel forged round bar is further demonstrated in Fig. 2. The Opitz codes of the round rod is 11103 2302. The attributes are denoted as a1-a9 for the round bar, a1=1 (Rotational parts, 0.5< L/D<3.) a2=1 (External shape element, stepped to one end.) a3=1 (Internal Shape element, smooth or stepped to one end.) a4=0 (No surface machining.) a5=5 (Auxiliary holes, radial.) a6=2 (50 mm. < diameter <=100 mm.) a7=3 (material is mild steel.) a8=0 (Internal form: Round bar.) a9=2 (Accuracy in coding digit.) The part family formation problem stated in this research can be formulated using a part-attribute incidence matrix $B=[b_{ij}]$, of size $m \times n$, where $m$ is the number of parts and $n$ is the number of attributes of that part. $b_{ij}$ represents the coding value (0-9) of $j^{th}$ attribute of $i^{th}$ part. A 10x9 example problem based on Opitz coding system is shown in Fig. 3. According to Opitz coding scheme every part of Figure 4 shows different attributes. First 5 columns are form codes and rest of the 4 columns represent supplementary codes.
The solution to the problem is to form the families of parts in such a way that the sum of similarities among the each pair of parts in a same family would be maximized. Therefore clustering methods are used in this article which groups the parts into families.

4. RESEARCH METHODOLOGY

Similarity coefficient based techniques are massively practiced in formation of manufacturing cells and a comprehensive study can be accomplished by Yin and Yasuda (2005). In this article a similarity measure method based on part family identification technique is utilized (Offodile, 1992). It is presented as,

\[ S_{ij} = \frac{\sum_{k=1}^{K} S_{ijk}}{K} \]  

(1)

Where

\[ S_{ijk} = 1 - \frac{|b_{ik} - b_{jk}|}{R_k} \]  

(2)

Where

- \( S_{ijk} \) is Similarity measured between part \( i \) and part \( j \) on attribute \( k \),
- \( K \) is total number of attributes considered,
- \( b_{ik} \) is part coding for part \( i \) on attribute \( k \),
- \( b_{jk} \) is part coding for part \( j \) on attribute \( k \),
- \( R_k \) is range of possible part codings for all parts on attribute \( k \).

Therefore sum of similarities is derived using the formula,

\[ S_{tot} = \sum_{n=1}^{N} S_n \]  

(3)

Where

\[ S_n = \frac{\sum_{i \in n} \sum_{j \in n} S_{ij}}{0.001 + C_{2}^{P_n}} \]  

(4)

Where \( S_n \) = similarity measure between part \( i \) and part \( j \)

\( C_{2}^{P_n} \) = Number of pair-wise combinations formed in part family \( n \), and \( P_n \) is the number of parts in family \( n \)

(in the denominator a small value of 0.001 is added to avoid the division by zero rule).

In order to facilitate the computation in Matlab the similarity matrix obtained using equation (1) further transformed into distance matrix using equation (5).

\[ d_{ij} = 1 - S_{ij} \]  

(5)

Complete Linkage Clustering Algorithm (C-Linkage), Single Linkage Clustering Algorithm (SLCA) and K-means algorithm are adopted in this study as the solution methodologies.

This abovementioned similarity metric technique is utilized in this study to calculate the similarity coefficient value between pair of parts presented as rows of part attribute incidence matrix as given in Table1. Therefore the similarity matrix generated for (10x9) part attribute incidence matrix using equation (1) is presented in Fig. 5.

4.1 Linkage Clustering Techniques

C-Linkage and SLCA are theoretically and mathematically simple algorithm practiced in hierarchical clustering analysis of data. It delivers informative descriptions and visualization of possible data clustering structures. When there exists hierarchical relationship in data this approach can be more competent. Complete linkage also known as furthest neighbor method which uses the distance between two clusters. The distance between cluster \( r \) and another cluster \( s \) is defined as the maximum distance between \( x_{ri} \), \( i^{th} \) object of cluster \( r \) and \( x_{sj} \), \( j^{th} \) object of cluster \( s \) and given as:

\[ C_{ij} = \max(d_{ij}), i \in (1, ..., n_r), j \in (1, ..., n_s) \]  

(6)

Single linkage also known as nearest neighbor method which uses the distance between two clusters. The distance between cluster \( r \) and another cluster \( s \) is defined as the maximum distance between \( x_{ri} \), \( i^{th} \) object of cluster \( r \) and \( x_{sj} \), \( j^{th} \) object of cluster \( s \) and given as:

\[ E_{ij} = \min(d_{ij}), i \in (1, ..., n_r), j \in (1, ..., n_s) \]  

(7)

Using equation (5) and (6), an intermediate matrix could be obtained which is a \((p-1)x3\) matrix, where \( p \) is the number of parts in the original dataset. Columns

<table>
<thead>
<tr>
<th>a1</th>
<th>a2</th>
<th>a3</th>
<th>a4</th>
<th>a5</th>
<th>a6</th>
<th>a7</th>
<th>a8</th>
<th>a9</th>
</tr>
</thead>
<tbody>
<tr>
<td>p1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>9</td>
<td>1</td>
</tr>
<tr>
<td>p2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>p3</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>6</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>p4</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>4</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>p5</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>8</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>p6</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>7</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>p7</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>8</td>
</tr>
<tr>
<td>p8</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>9</td>
<td>5</td>
</tr>
<tr>
<td>p9</td>
<td>0</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>4</td>
<td>8</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>p10</td>
<td>0</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>4</td>
<td>4</td>
<td>8</td>
</tr>
</tbody>
</table>
of the matrix contain cluster indices linked in pairs to form a binary tree. The leaf nodes are numbered from 1 to p. Leaf nodes are the singleton clusters from which all higher clusters are built. Further the dendrogram could be obtained from the matrix which indicates a tree of potential solutions. An example dendrogram structure obtained for the parts using C-Linkage method is shown in Fig. 5, which shows the clear cluster information of part families. Similar dendrogram could be produced using SLCA method (not shown in the article).

Input: part-attribute incidence matrix $A$
1. Procedure similarity ()
   1.1. Compute similarity values between pair of machines using equation (1)
   1.2. Compute the similarity matrix of the parts
   1.3. transform the similarity matrix into a distance matrix using equation (3)
1.4. End
2. Procedure Cluster ()
   2.1. loop
   2.2. Compute the smallest Euclidian distance between two clusters for SLCA method
   2.3. Compute the furthest Euclidian distance between two clusters for C-Linkage method
   2.4. Construct matrices of size $(m-1)\times3$ to from the hierarchical tree structure for both SLCA and C-Linkage method
   2.5. Construct dendrograms for SLCA and C-Linkage using matrices obtained from 2.2 and 2.3
   2.6. loop
   2.7. create part families for the highest level of similarity coefficient
   2.8. Compute the sum of similarities using equation (3) for SLCA and C-Linkage method
2.9. End
Output: Part family Configurations for SLCA and C-Linkage method

4.2 The K-means clustering algorithms
K-means clustering is an algorithm to classify objects based on attributes into $K$ number of groups (Hartigan and Wong, 1979). The grouping is done by minimizing the sum of squares of distances between data and the corresponding cluster centroid. Thus the purpose of K-mean clustering is to classify the data exploiting the Euclidean distance metric between the data-points. The basic steps of k-means clustering are simple. Number of clusters $K$ is fixed and the centroid of these clusters is assumed randomly or the first $K$ objects in sequence could also be chosen as the initial centroids. Thereafter the K-means algorithm would follow three steps,(i) Iterate until no object is left to be clustered, (ii) Determine the centroid coordinate, (iii) Determine the Euclidean distance of each object to the centroids,(iv) Group the object based on minimum distance. The pseudocode of K-means is given as,

Input: part-attribute incidence matrix, number of clusters "K"

Procedure k-means()
1.1. Loop
1.2. First K parts are marked as K centroids
1.3. flag = false
1.5. loop
1.6. For each cluster calculate minimum distance $d(k)$ between each part and cluster centroids

$p_1 p_2 p_3 p_4 p_5 p_6 p_7 p_8 p_9 p_{10}$

<table>
<thead>
<tr>
<th>p1</th>
<th>p2</th>
<th>p3</th>
<th>p4</th>
<th>p5</th>
<th>p6</th>
<th>p7</th>
<th>p8</th>
<th>p9</th>
<th>p10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.815</td>
<td>0.912</td>
<td>0.815</td>
<td>0.938</td>
<td>0.791</td>
<td>0.747</td>
<td>0.667</td>
<td>0.815</td>
<td>0.728</td>
</tr>
<tr>
<td>1</td>
<td>0.815</td>
<td>0.825</td>
<td>0.778</td>
<td>0.827</td>
<td>0.827</td>
<td>0.778</td>
<td>0.791</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0.84</td>
<td>0.889</td>
<td>0.791</td>
<td>0.747</td>
<td>0.716</td>
<td>0.765</td>
<td>0.728</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0.825</td>
<td>0.825</td>
<td>0.754</td>
<td>0.827</td>
<td>0.679</td>
<td>0.815</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0.754</td>
<td>0.728</td>
<td>0.654</td>
<td>0.825</td>
<td>0.716</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0.825</td>
<td>0.825</td>
<td>0.754</td>
<td>0.791</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0.679</td>
<td>0.852</td>
<td>0.914</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0.63</td>
<td>0.747</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0.815</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig. 4 similarity matrix obtained from $(10\times9)$ part-attribute incidence matrix of Fig. 3

Fig. 5 Dendrogram of part clusters of example problem $(10\times9)$

From Fig. 6 it can be stated that 3 part families are obtained using the hierarchical notations. The part families are, Family 1 {7,9,10}, Family 2 {1,3,4,5}, Family 3 {2,6,8}.

Similarly using the SLCA approach 7 part families could be achieved as, Family 1 {22}, Family 2 {23}, Family 3 {25}, Family 4 {1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,26}, Family 5 {24}, Family 6 {21}, Family 7 {27}

This pseudocode would explain both the hierarchical method SLCA and C-Linkage,
1.7. Find minimum of dk, marked as ‘min’
1.8. assign the part to kth cluster
1.9. flag = true
1.10. stop until all parts are assigned to K clusters
1.11. stop until maximum number of iterations reached

5. RESULTS AND DISCUSSIONS
The proposed techniques are tested on 5 different problem datasets of size 5x9 to 27x9. The largest dataset has been obtained from Haworth (1968) using Optiz coding system. Remaining 4 problems are designed using the aforesaid coding system. Problem datasets are provided in Fig. 3 and Fig. 6 to 9. The algorithms are coded in Matlab 7.0 environment and executed on PIV laptop computer. The obtained results are compared and shown in Table 1.

Table 1 demonstrates that each of the proposed methodologies are competent to attain good solutions and effective in constructing the families of parts. Despite of the fact that the solutions obtained are not identical therefore the sum of similarities are not identical for the test datasets. It depicts that C-Linkage technique outperformed the SLCA and K-means methods for the problems # 2, 3 and 4. However for the smallest dataset #1 (5x9) K-means produced nearly 50% improved result over SLCA and C-Linkage technique. Thereafter for the dataset #5 (27x9) nearly 10% improvement has shown by K-means over the other two. Thus for all the test problems C-linkage has reported an overall improvement of 60% and outpaced other two techniques. Hence it could be stated that although C-Linkage is moderately better than the other methods in terms of the goodness of solutions obtained, but K-means is also capable to produce good results. In terms of computational time all of the proposed methods are equally good and took minimum CPU

<table>
<thead>
<tr>
<th>a1</th>
<th>a2</th>
<th>a3</th>
<th>a4</th>
<th>a5</th>
<th>a6</th>
<th>a7</th>
<th>a8</th>
<th>a9</th>
</tr>
</thead>
<tbody>
<tr>
<td>p1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>9</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>p2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>p3</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>6</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>p4</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>4</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>p5</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>8</td>
<td>0</td>
<td>3</td>
</tr>
</tbody>
</table>

Fig. 6 Problem #1 (5x9) dataset

<table>
<thead>
<tr>
<th>a1</th>
<th>a2</th>
<th>a3</th>
<th>a4</th>
<th>a5</th>
<th>a6</th>
<th>a7</th>
<th>a8</th>
<th>a9</th>
</tr>
</thead>
<tbody>
<tr>
<td>p1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>9</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>p2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>p3</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>6</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>p4</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>4</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>p5</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>8</td>
<td>0</td>
<td>3</td>
</tr>
</tbody>
</table>

Fig. 7 Problem #3 (15x9) dataset

Fig. 8 Problem #4 (20x9) dataset

<table>
<thead>
<tr>
<th>a1</th>
<th>a2</th>
<th>a3</th>
<th>a4</th>
<th>a5</th>
<th>a6</th>
<th>a7</th>
<th>a8</th>
<th>a9</th>
</tr>
</thead>
<tbody>
<tr>
<td>p1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>9</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>p2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>p3</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>6</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>p4</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>4</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>p5</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>8</td>
<td>0</td>
<td>3</td>
</tr>
</tbody>
</table>

Fig. 9 Problem #5 (27x9) dataset

<table>
<thead>
<tr>
<th>a1</th>
<th>a2</th>
<th>a3</th>
<th>a4</th>
<th>a5</th>
<th>a6</th>
<th>a7</th>
<th>a8</th>
<th>a9</th>
</tr>
</thead>
<tbody>
<tr>
<td>p1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>9</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>p2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>p3</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>6</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>p4</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>4</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>p5</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>8</td>
<td>0</td>
<td>3</td>
</tr>
</tbody>
</table>
time (less than 10 seconds) for all the datasets tested. Table 2 reports the percentage of perfection achieved while developing the part groups by each of the methods. Further Fig. 11 establishes the superiority shown by C-Linkage over other two methods.

Table 1 Comparison of performance shown by SLCA, C-Linkage and K-Means

<table>
<thead>
<tr>
<th>Dataset size</th>
<th>Part families obtained</th>
<th>SLCA</th>
<th>C-Linkage</th>
<th>K-Means</th>
<th>Maximum similarities</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 5x9</td>
<td>Family 1 {2}, Family 2 {1,3,4,5}</td>
<td>Family 1 {2}, Family 2 {1,3,4,5}</td>
<td>Family 1 {1,3,5}, Family 2 {2,4}</td>
<td>0.8641 0.8641 1.7108</td>
<td></td>
</tr>
<tr>
<td>2 10x9</td>
<td>Family 1 {1,3,4,5}, Family 2 {2,7,8,9,10}, Family 3 {6}</td>
<td>Family 1 {7,9,10}, Family 2 {1,3,4,5}, Family 3 {2,6,8}</td>
<td>Family 1 {1,5,9}, Family 2 {2,6,7,8,10}, Family 3 {3,4}</td>
<td>1.6492 2.5425 2.4902</td>
<td></td>
</tr>
<tr>
<td>3 15x9</td>
<td>Family 1 {4}, Family 2 {1,3,5}, Family 3 {6,7,9,10,11,12,13,14}, Family 4 {2,8,15}</td>
<td>Family 1 {6,11,12}, Family 2 {2,8,15}, Family 3 {1,3,4,5}, Family 4 {7,9,10,12,13,14}</td>
<td>Family 1 {7,9,10,13,14}, Family 2 {2,6,8,11,12}, Family 3 {1,3,5}, Family 4 {4}</td>
<td>2.5707 3.4338 2.5658</td>
<td></td>
</tr>
<tr>
<td>4 20x9</td>
<td>Family 1 {6,9,11,12}, Family 2 {1,3,5,7,10,13,14,17,18,19,20}, Family 3 {4}, Family 4 {16}, Family 5 {2,8,15}, Family 6 {6,11,12}, Family 7 {7,10,13,14,17,18,19,20}</td>
<td>Family 1 {7,10,14,19}, Family 2 {9,13,17,18,20}, Family 3 {6,11,12}, Family 4 {2,8,15}, Family 5 {1,3,4,5,16}</td>
<td>Family 1 {9,13,14,17,18,20}, Family 2 {2,6,7,8,10,11,12,15,19}, Family 3 {1,3,5}, Family 4 {4}, Family 5 {16}</td>
<td>2.4907 4.2510 2.5323</td>
<td></td>
</tr>
<tr>
<td>5 27x9</td>
<td>Family 1 {22}, Family 2 {23}, Family 3 {25}, Family 4 {1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,26}, Family 5 {24}, Family 6 {21}, Family 7 {27}</td>
<td>Family 1 {6,11,12}, Family 2 {2,8,15}, Family 3 {21}, Family 4 {22,23,24}, Family 5 {27}, Family 6 {1,3,4,5,16,25}, Family 7 {7,9,10,13,14,17,18,19,20,26}</td>
<td>Family 1 {9,17,18,20,23}, Family 2 {2,8,15,26}, Family 3 {1,3,5}, Family 4 {4,22,24,25}, Family 5 {16}, Family 6 {6,11,12}, Family 7 {7,10,13,14,19,21,27}</td>
<td>0.7744 4.1631 4.9185</td>
<td></td>
</tr>
</tbody>
</table>

6. CONCLUSIONS
Three different clustering techniques, SLCA, C-Linkage and K-Means algorithms are employed in this research to form part families. Since part coding and classification techniques are merely adopted in group technology problems, therefore the objective of this study is to utilize the heavily practiced coding system called Opitz part coding. 5 test datasets ranging from 5x9 to 27x9 are tested using the aforementioned techniques. Due to the NP-complete nature of the reported problems these methods are equally efficient to produce optimal solutions. The proposed methods are compared among each other successfully. The objective function utilized in this study is to maximize the sum of similarities among parts in all the part
families formed. From the beginning of this research work the number of part families to be formed is considered as constant. As shown in Table 1 C-Linkage algorithm has outperformed SLCA and K-means in terms of solution quality (sum of similarities value) which is further depicted in Figure 1 in terms of the percentage of perefections achieved while forming the part families by each of the methods. This study has assumed identical weightage for each and every attribute; however in formation of part families some attributes could be more significant than the other attributes. Therefore future work could be done by considering fractional weightage for each of the attributes. This work could also be extended by considering operational time and sequence of each part to develop more effective and robust part families.

Table 2 Comparison among perfection of the obtained results by the algorithms

<table>
<thead>
<tr>
<th>Problem</th>
<th>No. of parts</th>
<th>Sum of similarities, $\sum S_n$</th>
<th>Perfection percentage, $\sum S_n / N \times 10$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dataset</td>
<td>families formed(N)</td>
<td>SLCA</td>
<td>C-Linkage</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>0.8641</td>
<td>0.8641</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>1.6492</td>
<td>2.5425</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>2.5707</td>
<td>3.4338</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>2.4907</td>
<td>4.2510</td>
</tr>
<tr>
<td>5</td>
<td>7</td>
<td>0.7744</td>
<td>4.1631</td>
</tr>
</tbody>
</table>

APPENDICES

A. A numerical example of SLCA approach

The following information is given.

(i) Information about parts (relevant part attributes as explained in this article with 5 form codes of Opitz coding system):

Part1: 1 2 3 4 5
Part2: 0 2 4 3 5
Part3: 2 1 3 4 0
Part4: 3 5 1 4 7
Part5: 8 4 3 5 7

(ii) Information about significant part attributes

When, Attribute 1 is numeric, $R_1 = 9$; Attribute 2 is numeric, $R_2 = 9$
Attribute 3 is numeric, $R_3 = 9$; Attribute 4 is numeric, $R_4 = 9$
Attribute 5 is numeric, $R_5 = 9$.

Step 1: Calculated pairwise similarity coefficients ($S_{ij}$) from equation (1).

$S_{12} = 0.9333$
$S_{13} = 0.8444$  $S_{23} = 0.7778$
$S_{14} = 0.8000$  $S_{24} = 0.7333$
$S_{15} = 0.7333$  $S_{25} = 0.6667$
$S_{34} = 0.6889 \quad S_{35} = 0.6222$

$S_{45} = 0.8000$

$S_{12}$ is determined as follows:-

$S_{12} = |\frac{|b_{12}-b_{22}|}{R_2}| = |\frac{1}{9}| = 0.888$

$S_{13} = |\frac{|b_{13}-b_{33}|}{R_3}| = |\frac{2}{9}| = 1.000$

$S_{14} = |\frac{|b_{14}-b_{44}|}{R_4}| = |\frac{3}{9}| = 0.888$

$S_{15} = |\frac{|b_{15}-b_{55}|}{R_5}| = |\frac{4}{9}| = 1.000$

Hence $S_{ij}$ is determined as follows:

$S_{121} = |\frac{|b_{12}-b_{22}|}{R_2}| = |\frac{1}{9}| = 0.888$

$S_{122} = |\frac{|b_{12}-b_{22}|}{R_2}| = |\frac{1}{9}| = 1.000$

$S_{123} = |\frac{|b_{12}-b_{22}|}{R_2}| = |\frac{1}{9}| = 0.888$

$S_{124} = |\frac{|b_{12}-b_{22}|}{R_2}| = |\frac{1}{9}| = 0.888$

$S_{125} = |\frac{|b_{12}-b_{22}|}{R_2}| = |\frac{1}{9}| = 1.000$

Hence $S_{ij}$ is determined as follows:

All other $S_{ij}$ are determined in the same method.

**Step 2:** Calculate pairwise distances ($d_{ij}$) using equation (5)

$d_{12} = 0.0667$

$d_{13} = 0.1556 \quad d_{23} = 0.2222$

$d_{14} = 0.2000 \quad d_{15} = 0.2667$

$d_{24} = 0.2667 \quad d_{25} = 0.3333$

$d_{34} = 0.3111 \quad d_{35} = 0.3778$

$d_{45} = 0.2000$

**Step 3:** Form initial part family.

Because $d_{12}$ is the smallest distance, therefore PF1 is formed with parts 1 and 2 as its members.

**Step 4:** Calculate minimum distance $E_{ij}$

$E_{1,2,3} = 0.1556$

$E_{1,2,4} = 0.2000 \quad E_{4,5} = 0.3111$

$E_{1,2,5} = 0.2667 \quad E_{1,5} = 0.3778$

$E_{2,4} = 0.2000$

$E_{1,2,3}$ is obtained as,

$E_{1,2,3} = \text{min}(d_{13}, d_{23}) = \text{min}(0.1556, 0.2222) = 0.1556$

Other $E_{1,2,3}$ are calculated in the similar manner. Thereafter $E_{1,2,3}$ has the lowest value, hence part 3 is introduced in family PF1.

**Step 5:** Repeat step 4 for part 4 and 5.

$E_{1,2,4} = 0.2000$

$E_{1,2,5} = 0.2667 \quad E_{4,5} = 0.2000$

$E_{1,2,3}$ is obtained as,

$E_{1,2,4} = \text{min}(d_{14}, d_{44}) = \text{min}(0.2000, 0.3111) = 0.2000$

$E_{4,5}$ has the lowest value therefore part 1, 2 and 3 are grouped in family PF1, and part 4 and 5 are grouped in the 2nd family PF2.

**Step 6:** Thus all the parts are grouped. PF1 contains [parts 1, 2, 3] and PF2 contains [parts 4, 5]. Stop.

Sum of similarities for PF1 is achieved as,

$S_1 = \frac{S_{12}+S_{13}+S_{14}}{0.001+C^2} = 0.9333+0.8444+0.7778 \quad \frac{3}{0.001} = 0.8518$

Sum of similarities for PF2 is achieved as,

$S_2 = \frac{S_{45}}{0.001+C^2} = \frac{0.8000}{1.001} = 0.8000$

Therefore sum of similarities for PF1 and PF2 is achieved as,

$S = S_1 + S_2 = 0.8518 + 0.8000 = 1.6518$

C-Linkage method follows the same method of SLCA. The only difference is instead of calculating minimum distance in step 3 to 5, C-Linkage calculates the maximum distance.

**B. A numerical example of K-means approach**

The following information is given.

(i) Information about parts: (illustrated only with two attributes for four parts)

<table>
<thead>
<tr>
<th>Part</th>
<th>Attr 1</th>
<th>Attr 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Part2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Part3</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>Part4</td>
<td>5</td>
<td>4</td>
</tr>
</tbody>
</table>

1. Initial value of centroids: Suppose we use part1 and part2 as the first centroids. Let $c_1$ and $c_2$ denote the coordinate of the centroids, then $c_1 = (1, 1)$ and $c_2 = (2, 1)$.

2. Parts-Centroids distance: We calculate the distance between cluster centroid to each part. Let us use Euclidean distance, then we have distance matrix at iteration 0 is,

$$D^0 = \begin{bmatrix}
0 & 1 & 3.61 & 5 \\
1 & 0 & 2.83 & 4.24
\end{bmatrix}$$

Each column in the distance matrix symbolizes the part. The first row of the distance matrix corresponds to the distance of each part to the first centroid and the second row is the distance of each part to the second centroid. For example, distance from part3 = (4, 3) to the first centroid $c_1$ is

$$\sqrt{(4-1)^2 + (3-1)^2} = 3.61$$

And to the second centroid $c_2$ is,

$$\sqrt{(4-2)^2 + (3-1)^2} = 2.83$$

3. Objects clustering: We assign each part based on the minimum distance. Thus, part1 is assigned to group 1, part2 to group 2, part3 to group 2 and part4 to group 2.
4. Iteration-1, determine centroids : Knowing the members of each group, now we compute the new centroid of each group based on these new memberships. Group 1 only has one member thus the centroid remains in \( c_1 = (1,1) \). Group 2 now has three members, thus the centroid is the average coordinate among the three members: 
\[
\begin{align*}
    c_2 &= \left( \frac{1+3+5}{3}, \frac{1+3+4}{3} \right) \\
    &= \left( \frac{9}{3}, \frac{7}{3} \right)
\end{align*}
\]

5. Iteration-1, Parts-Centroids distances : The next step is to compute the distance of all objects to the new centroids. Similar to step 2, we have distance matrix at iteration 1 as,
\[
D^1 = \begin{bmatrix}
    0 & 1 & 3.61 & 5 \\
    3.14 & 2.36 & 0.47 & 1.89
\end{bmatrix}
\]

6. Iteration-1, Parts-clustering: Similar to step 3, we assign each part based on the minimum distance. Based on the new distance matrix, we move the part2 to Group 1 while all the other parts remain.

7. Iteration 2, determine centroids: Now we repeat step 4 to calculate the new centroids coordinate based on the clustering of previous iteration. Group1 and group 2 both has two members, thus the new centroids are
\[
c_1 = \left( \frac{1+2+2}{3}, \frac{1+1+1}{3} \right) = \left( \frac{3}{2}, 1 \right) \quad \text{and} \quad c_2 = \left( \frac{4+5+3}{2}, \frac{3+4}{2} \right) = \left( \frac{9}{2}, \frac{7}{2} \right)
\]

8. Iteration-2, Parts-Centroids distances : Repeat step 2 again, we have new distance matrix at iteration 2 as,
\[
D^2 = \begin{bmatrix}
    0.5 & 0.5 & 3.2 & 4.61 \\
    4.3 & 3.54 & 0.71 & 0.71
\end{bmatrix}
\]

9. Iteration-2, Parts clustering: Again, we assign each object based on the minimum distance. Therefore group 1 contains part1 and part2 and group 2 contains part3 and part 4.

10. We obtain, comparison of the grouping of step 6 and step 9 reveals that the parts do not move group anymore. Thus, the computation of the k-mean clustering has reached its stability and no more iteration is needed. We get the final grouping as the results PF1 {parts 1 and 2} and PF2 {parts 3 and 4}.

REFERENCES


