Management and Production Engineering Review
Volume 1 • Number 4 • December 2010 • pp. 26–37

A HYBRID HEURISTIC BASED CLUSTERING ALGORITHM TO DESIGN MANUFACTURING CELL

Tamal Ghosh, Sourav Sengupta, Pranab K Dan

West Bengal University of Technology, Department of Industrial Engineering and Management

Corresponding author:
Tamal Ghosh
West Bengal University of Technology
Department of Industrial Engineering and Management
BF 142, Salt Lake City, Kolkata 700064 India
phone: +91 33-2334-1014
e-mail: tamal.31@gmail.com

Received: 9 September 2010
Accepted: 10 November 2010

Abstract

The manufacturing cell formation problem consists of designing a subclass of machine cells and their corresponding part families with an objective to minimize the inter-cell and intra-cell moves of the items while enhancing the machine utilization. This paper demonstrates a hybrid heuristic algorithm namely HACCF (**H**euristic based **A**gglomerative **C**lustering for **C**ell **F**ormation) exploiting the centroid linkage clustering method and the Minkowski distance metric as dissimilarity coefficient. An exhaustive heuristic technique is developed and combined with the proposed clustering method to form manufacturing cells. 15 widely practiced datasets are obtained from past literature and tested with the proposed technique. The computational results are exhibited using the grouping efficacy as performance measure for the abovementioned test problems. The proposed technique is shown to outperform the standard and published techniques such as ZODIAC, GRAFICS, TSP based genetic algorithm and simple genetic algorithm and attained 73.33% improved result by exceeding the solution quality on the test problems. Therefore the proposed technique could be extensively used and could be hybridized with intelligent approaches to obtain more improved result in the vicinity of future cellular manufacturing system.

Keywords

cellular manufacturing, group technology, cell formation, centroid linkage clustering technique, dissimilarity measure, heuristic, hybrid approach.

Introduction

As reported in cellular manufacturing systems, group technology could be anticipated as a manufacturing philosophy which recognises similar parts, therefore associating them into part families depending on its manufacturing designs, characteristics and geometric shapes which was first presented by Burbidge [1–3]. Group technology is employed in cellular manufacturing systems to bring in front a replacement of conventional manufacturing system. Constructing manufacturing cell has been called cell formation problem (CF/CFP), it consists of the following approaches: usually similar parts are clustered into part families following their processing requirements or exploiting the part coding systems such as Opitz, MICLASS and DCLASS [4] and dissimilar machines are grouped into manufacturing cells and subsequently part families are assigned to cells. The problem encountered in CMS is the construction of cells irrespective of its nature [5]. Not mandatorily the aforestated steps are carried out in the abovementioned order or even sequentially. Depending on the procedures involved in CFP, three methods of achieving solutions are identified [6]: (1) recognizing part families first and consequently machines are grouped into cells depending on the processing requirements of the part families, (2) recognizing manufacturing cells by grouping dissimilar machines and then the part families are allocated to the cells, (3) part families and machine cells are developed concurrently. Due to the NP-Complete nature of the problem [7], many recent computational techniques are proposed which are heavily practised to achieve improved solution to the CFP [8].

Throughout the last few decades many hierarchical and non-hierarchical clustering techniques are adopted by researchers in the aforementioned area such as ZODIAC [9], GRAFICS [10], MST [11], K-Harmonic Mean algorithm [7], Fuzzy C-Means algorithm [12] etc. The present research reports an effective hybrid clustering technique namely HACCF to obtain improved cell configuration in cellular manufacturing.

This article is further structured in following order. Section 2 presents a brief literature survey based on cell formation problems and its solution methodologies, Sec. 3 presents the problem definition, Sec. 4 discusses the research methodology, Sec. 5 depicts the details about the experimentation and the computational results. The conclusion of this research is provided in Sec. 6.

Literature survey

Various techniques are developed to solve cell formation problems since last forty years, these include similarity coefficient methods, clustering analysis, array based techniques, graph partitioning methods and soft computing approach etc.

The similarity coefficient approach was first suggested by McAuley [13]. The basis of similarity coefficient methods is to measure the similarity between each pair of machines and then to group the machines into families based on their similarity measurements. Some studies have proposed to measure dissimilarity coefficients instead of similarity coefficient for cell formation. Reference [14] have used dissimilarity coefficients for generalized cell formation considering the operation sequences and production volumes of parts. Most similarity based methods used machine-part mapping chart. Some of the techniques, which utilized this approach, are Single linkage clustering algorithm [13], Average linkage clustering algorithm [15] etc.

Clustering methods can be categorized as hierarchical and non-hierarchical methods. Standard or modified clustering techniques could be used to construct groups of either parts or machines. Among these, [10, 13, 15–30] are recognized as familiar techniques in the past literature. Machine–part group analysis (MPGA) is based on production flow analysis. In MPGA based methods the machine-part groups are constructed by rearranging rows and columns of the machine-part mapping chart in the form of a binary incidence matrix. Some of the MPGA methods are Rank order clustering [18], Bond energy algorithm [15] etc. Reference [31] has reported a hierarchical algorithm based on genetic programming in CMS which efficiently forms cells.

Array based techniques utilize the rows and columns of the binary incidence matrix as binary words and reorder them to achieve a block diagonal formation. The rank order clustering algorithm is the most popular array-based method for group technology [18]. Substantial modifications and enhancements over rank order clustering algorithm have been described in [19] and [25]. The direct clustering analysis has been implemented in [20], and bond energy analysis in [18].

Graph Theoretic Approach represents the machines as vertices and the similarity between machines as the weights of the arcs. Reference [32] suggested the use of graph theory to form machine groups. Reference [25] proposed an ideal seed non-hierarchical clustering algorithm for cellular manufacturing. Reference [33] developed graph searching algorithms which select a key machine or component based on a pre-fixed criterion. Reference [34] depicted a non-heuristic network method to construct manufacturing cells to minimize intercellular traffic. Reference [11] reported a new method based on Minimum Spanning Tree (MST) for the cell formation problem. Reference [35] proposed a polynomial-time algorithm using a graph theoretic approach to construct optimal cluster namely vertex-tree graphic matrices.

Soft computing approaches are exclusively practiced for the NP-complete problems [7] and adopted for the cell formation problems since past two decades [8]. DCLASS coding system based genetic algorithm technique was proposed by Lee-Post [4] to form efficient part families. Tabu search, simulated annealing and genetic algorithms are utilized to capture the dynamic conditions of cell formation problems in [36]. References [48, 49] stated a graph partitioning formulation of CFP which utilized a binary Genetic Algorithm (GA) and thereafter a Branch & Bound method is combined to enhance the GA and the authors further considered dynamic production factors such as input data with realistic constraints and tried to avoid assumptions such as static number of cells, hence they proposed an improved GA based methodology with the help of fuzzy logic. Defersha and Chen [50–52] developed a mathematical model which incorporated dynamic cell configuration, alternative routings, sequence of operations, multiple units of identical machines, machine capacity, workload balancing among cells, operation cost, subcontracting cost, tool consumption cost, set-up cost and other practical constraints and a two-phase

GA based heuristic technique was proposed. The authors further experimented with parallel GA model for dynamic-CFP considering various parameters such as connection topology, migration policy, migration frequency and migration rate and thereafter attempted to minimize production and quality related costs by incorporating a number of manufacturing attributes and practical constraints. Reference [53] designed a multiobjective- optimization model using GA, where fitness evaluation was performed via simulation of CMS. while reference [54] presented dynamic production condition considering product mix, demand of parts during some period, machine movement, addition of new equipment, by providing flexibility in CM. A hybrid methodology based on Boltzmann function from simulated annealing (SA) and mutation operator from GA was proposed by Wu et al. [55] to optimize the initial cluster obtained from similarity coefficient method (SCM) and rank order clustering (ROC). Reference [56] reported a modified Particle Swarm method with proportional likelihood instead of using velocity vector on CF problems where the objectives are the minimization of cell load variation and inter cellular parts movement and reported the stability of the method with low variability. A similar study was also performed in [57] where a hybrid particle swarm optimization technique for CFP was reported. The technique utilized mutation operator embedded in velocity update equation to avoid reaching local optimal solutions. Reference [58] developed a sequential model based on SA for large-scale problems and compared their method with GA. However reference [59] investigated an Ant Colony Optimization (ACO) based Tabu Search (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. Further they extended the work in [60] and 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.

Therefore the past literature demonstrates that various techniques are being developed to solve CFP. Although different methods have their own merits and demerits, but the objective is to obtain more improved solution for CFP. Thus our article proposes a new hybrid heuristic method to solve the aforementioned problem to achieve near optimal solutions.

Problem definition

The cell formation problem in group technology begins with two fundamental tasks, namely, machine cell formation and part family identification. Similar machines are grouped to form machine cells and dedicated for the manufacture of one or more part families. In part family formation, parts with similar design features, attributes, shapes are grouped, so that the group of parts can be manufactured within a cell. Generally the cell formation problems are presented using a matrix namly machine-part incident matrix in which all the elements are either 0 or 1. Parts are arranged in cloumns and machines are in rows of the incidence matrix. An example matrix is presented in Table 1.

Table 1
Machine-part incidence matrix (3×5).

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

In this matrix a 0 indicates no mapping or no processing and an 1 indicates mapping or processing. The Block diagonal structure is shown in Table 2. Therefore it depicts that machine 1 processes parts 2, 3, 5, machine 2 processes parts 1, 4, 5 and machine 3 processes parts 3, 5.

Table 2
Block diagonal matrix (3×5).

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

Table 2 presents the formation of cells as diagonal square boxes. Once some appropriate technique is employed to CFP this solution matrix is believed to be obtained. It can be interpreted that cell 1 contains {machines 1, 3 || parts 2, 3, 5} and cell 2 contains {machine 2 || parts 1, 4}. An 1 outside the block means a part is processed through some machine which does not belong to any machine cell, therefore the intercellular move cost will be added. This element is known as an 'exceptional element' (EE) and a 0 inside a cell means an unutilized space in cell, therefore lesser utilization of space and increased intracell move cost. It is known as a 'void'. The objective of cell formation is to minimize the EEs and voids.

Performance measure

Two widely accepted performance measures to assess the goodness of CFP solutions are grouping efficiency [26] which incorporates machine utilization and intercell moves, and Grouping efficacy [38], intends to minimize the number of exceptional elements and the number of voids in the diagonal blocks. A detailed description regarding various performance measure could be obtained from a critical survey of reference [37]. In this study grouping efficacy measure is used as the evaluation criterion to test the effectiveness of the proposed HACCF algorithm. Grouping efficacy measure is stated as,

$$\tau = \frac{E - E_e}{E + E_v} = 1 - \frac{E_v + E_e}{E + E_v}, \tag{1}$$

where E – total number of 1s in incidence matrix, E_e – total number of exceptional elements, E_v – total number of voids.

Solution methodology

Agglomerative clustering is conceptually and mathematically simple algorithm practiced in clustering analysis of data [39, 46, 47]. It delivers informative descriptions and helps visualizing potential data clustering structures. If there exists hierarchical relationship in data this approach can be more competent. In this article a 6-step algorithm namely HACCF is proposed using agglomerative clustering which is further combined with an exhaustive heuristic technique to facilitate the cell formation procedure.

STEP 1: Formation of input dataset

An input data set for HACCF is a machine–part incidence matrix. machines are the items that should be grouped based on their similarities. Parts are the components which contains routing information. The type of input datasets can be classified into binary data (contains only 0 or 1 i.e. the routing information) and ratio data (contains information about production volume, operation time). In this study binary datasets are considered.

Step 2: Computing pair-wise distance of machines and forming dissimilarity matrix

This function computes the distance between each pairs of machines of the given input data matrix. It produces an output vector of length $m^*(m-1)/2$ where m is the number of rows of the input matrix. This output is commonly used to form the dissimilarity matrix and employed in clustering or in multidimensional scaling. The distance or dissimilarity is computed using Minkowski distance metric. The computation is performed using the following formula,

$$d_{ij} = \sqrt[p]{\sum_{k=1}^{n} |m_{ik} - m_{jk}|^p}. \tag{2}$$

This metric is a generalized form of Euclidian, Chebychev and City block distance metrics. 'm' is denoted as machines of the incidence matrix. The machine-machine distance matrix for (5×7) test problem #1 [21] is computed using equation (2) and presented in Table 3.

Table 3
Machine-machine distance matrix.

	m1	m2	m3	m4	m5
m1	0.000	2.449	2.000	2.449	2.000
m2		0.000	1.414	1.414	1.414
m3			0.000	2.000	1.414
m4				0.000	2.000
m5					0.000

STEP 3: Applying centroid linkage technique and dendrogram formation

This function is developed on the basis of hierarchical cluster formation. If cell r is formed from cell p and q, and n_r is the number of machines in cell r, x_{ri} is the i^{th} machine of cell r, then centroid linkage is computed using the formula,

$$d(r,s) = \|\overline{x}_r - \overline{x}_s\|_2, \tag{3}$$

which is the Euclidean distance between the centroids of two cells where,

$$\overline{x}_r = \frac{1}{n_r} \sum_{i=1}^{n_r} x_{ri}. \tag{4}$$

This function is applied on the distance matrix of Table 3. The output matrix is generated using this function is a $(m-1) \times 3$ matrix namely z,

$$z = \begin{bmatrix} 3 & 5 & 1.414 \\ 2 & 6 & 1.225 \\ 4 & 7 & 1.633 \\ 1 & 8 & 2.000 \end{bmatrix}.$$

In present computation m is the number of machines in the original dataset. Columns of the matrix contain cluster indices linked in pairs to form a binary tree. The leaf nodes are numbered from 1 to m. Leaf nodes are the singleton clusters from which all higher clusters are built. Therefore for the problem in hand, the initial cluster is formed by machines 3 and 5 with dissimilarity value 1.414 and combined with a single cluster index 6. Thereafter cluster index 6 is combined with singleton machine 2 with dissimilarity value 1.225 and cluster index 7 is generated. Further singleton machine 4 and cluster index 7 are combined with dissimilarity value 1.633

and cluster index 8 is generated. At the end singleton machine 1 and cluster index 8 are combined with maximum dissimilarity value 2. The dendrogram is obtained from the problem matrix presented in Fig. 1 which indicates the hierarchical structure and relationship between clusters. From the dendrogram two clusters or cells could be identified,

Cell 1: contains machine 1 and 4

Cell 2: contains machine 2, 3 and 5

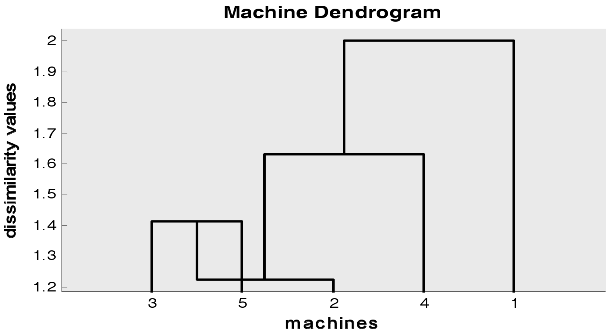

Fig. 1. Dendrogram obtained for machine cluster.

STEP 4: Constructing agglomerative clustering from centroid linkage

This routine constructs clusters from the agglomerative hierarchical cluster tree, as generated by the linkage function. A threshold value is used for cutting the linkage matrix obtained from previous step into clusters. Clusters are formed when a node and all of its subnodes have inconsistent value less than the threshold value. All leaves at or below the node are grouped into a cluster. Output is a vector of size m containing the cluster assignments of each machine row. Therefore the vector obtained could be presented as,

$$T = [1, 2, 2, 1, 2].$$

STEP 5: Part grouping method

Once the machine clusters are formed, the part family should be assigned to the machine cells. In order to obtain the part family, step 1–4 are repeated on part-machine incidence matrix which is a transpose matrix of machine-part incidence matrix. The dendrogram obtained for part cluster is depicted in Fig. 2. The vector containing the cluster information for parts is presented as,

$$T1 = [1, 2, 2, 2, 2, 2, 1].$$

This presents that part 1 and 7 are in cell 1 and part 2 to 6 are in cell 2. Once the machines and parts are assigned to the corresponding cells, the goodness of the solution is measured using the grouping efficacy metric given in equation (1). In present scenario this obtained feasible solution is not essentially be the best solution, therefore the scope remains in improving the quality of the solution obtained.

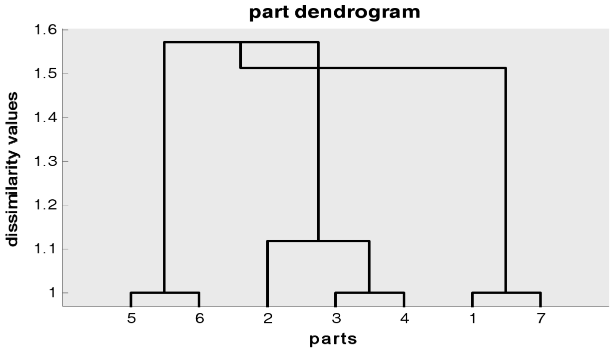

Fig. 2. Dendrogram obtained for part cluster.

STEP 6: Proposed heuristic method

In order to improve the quality of solution with an aim to reduce the number of exceptional elements and voids from the configuration i.e. to minimize the intercell and intracell moves a novel heuristic approach is proposed in this research which is utilized to hybridize the HACCF algorithm.

This heuristic is an iterative procedure which investigates the grouping efficacy value in every step of the technique. It tries to maximize the value of grouping efficacy by randomly generating solutions or by randomly swapping a machine or a part from one cell to another in the generated solution.

This technique subsequently searches for better solution and tries to improve the quality of solution. It eventually stops after a fixed number of iterations or if the solution has not been improved after a certain number of consecutive iterations. The final output obtained is the block diagonal solution matrix.

In the abovementioned heuristic method the initial input is a solution string to the problem in hand which is generated from centroid linkage clustering technique. Therefore the initial input string is $T2$ which is obtained from T and $T1$,

$$T2 = [1, 2, 2, 1, 2, 1, 2, 2, 2, 2, 2, 1].$$

The size of the solution string is $m + p$. where m is the number of machines and p is the number of parts. Each bit of the string represents cell number of the corresponding machine or part (string indices). Therefore $T2$ states machine 1 and 4 are placed in cell 1, and machine 2, 3, 5 are placed in cell 2, and part 1 and 7 are placed in cell 1, and part 2 to 6 are placed in cell 2.

The proposed HACCF is given in next subsection, Proposed HACCF Pseudocode

Input:*Machine-part incidence matrix A*

Procedure *pair_wise_distance()*

1.1. compute distance values between pair of machines using equation (2)
1.2. compute the distance or dissimilarity matrix of the machines

Procedure *cluster()*

2.1. loop
2.2. Compute the Euclidian distance between the centroids of two cells using equation (3)
2.3. Construct matrix of size $(m-1)\times 3$ to from the hierarchical tree structure
2.4. construct dendrogram from the binary matrix computed using centroid linkage rule
2.5. create machine cells for the highest level of dissimilarity coefficient
2.6. assign parts to machine cells using the step 2.1–2.4 on the transpose of the input matrix

Procedure *improvement_heuristic()*

3.1. input the initial solution string s_0
3.2. calculate the objective value 'f_0' for input string using equation (1)
3.3. set *iter = 1*
3.4. *best_fval* $= f_0$
3.5. *best_sol* $= s_0$
3.6. while *iter <= iter_max*
3.7. create initial set 'S' of randomly generated strings $(s_i \in S,\ i = 1, 2, 3, \ldots, n)$
3.8. for *i = 1 to n*
3.9. calculate objective value 'f_i' for solution string s_i
3.10. compute $\theta = (f_i - f)$
3.11. if $\theta>$ small random no. (1.0000e-006)
3.12. *best_sol* $= s_i$
3.13. *best_fval* $= f_i$
3.14. else
3.15. pick a machine/part randomly and put it to another cell in current solution string (single move)
3.16. check for the objective function value 'f_i'
3.17. repeat 3.10 to 3.15 until $f_i > f$
3.18. accept the arrangement
3.19. else eliminate the solution string
3.20. stop if maximum iteration has been reached or best solution does not change in prefixed number of iterations

Output: *Block diagonal matrix*

Convergence analysis are almost equivalent for all the problem datasets. Problem #1 (Waghodekar and Sahu [21]) of size 5×7 is selected as an example to illustrate the convergence curve during iterations of the heuristic technique. For the first iteration the objective value ($\tau\times 100$) obtained is to be 50 initially. Since the computer program is designed to maximize the objective value therefore It increases when the number of iteration increases. At the 10^{th} iteration it attains the value of 62.5, an increase of 25%. The final optimal solution is obtained during the 53^{rd} iteration having the objective value of 69.56, an increase of 39.12%. Based on the experimentation for all the problems reported in Table 4, it is observed that

Table 4
Comparison of HACCF with published results from literature.

#	dataset	size	cell	EE	voids	ZODIAC	GRAFICS	GATSP	GA	HACCF
1	Waghodekar and Sahu [21]	5×7	2	4	3	56.52	60.87			**69.56****
2	Seifoddini [28]	5×18	2	7	3			77.36	77.36	**79.59**
3	Kusiak and Chow [40]	7×11	3	6	8	39.13	53.12	46.88	50	**53.13**
4	Chandrasekharan and Rajagopalan [25]	8×20	2	28	17	58.33	58.13	58.33	55.91	58.33
5	Chandrasekharan and Rajagopalan [26]	8×20	3	8	0	85.24	85.24	85.24	85.24	**86.67**
6	Chan and Milner [20]	10×15	3	0	4	92	92	92	92	92
7	Mosier and Taube [23]	20×20	6	54	25	21.63	38.26	37.12	34.16	**39.23**
8	Carrie [17]	20×35	5	7	33	75.14	75.14	75.28	66.3	**75.9**
9	Boe and Cheng [41]	20×35	5	43	36			55.14	44.44	**56.21**
10	Chandrasekharan and Rajagopalan [42]	24×40	7	0	0	100	100	100	100	100
11	Chandrasekharan and Rajagopalan [42]	24×40	7	10	11	85.1	85.1	85.11	85.11	85.11
12	Kumar and Vannelli [43]	30×41	11	38	23	33.46	55.43	53.8	40.96	**60.13**
13	Stanfel [24]	30×50	11	25	42	46.06	56.32	56.61	48.28	**60.12**
14	Stanfel [24]	30×50	11	63	44	21.11	47.96	45.93	37.55	**48.06**
15	Chandrasekharan and Rajagopalan [9]	40×100	10	37	37	83.92	83.92	84.03	83.9	**84.15**

** improved results are shown in boldface the objective value is increased when the number of iterations increased until it reaches the best objective value at some iteration and thereafter the objective value continues to remain constant even though the number of iterations is increased. Since the proposed heuristic gives the same pattern of convergence for all the tested problems therefore the convergence property is verified. Figure 3 could be observed to experience the convergence characteristics. The solution matrix of the problem is presented in Fig. 5.

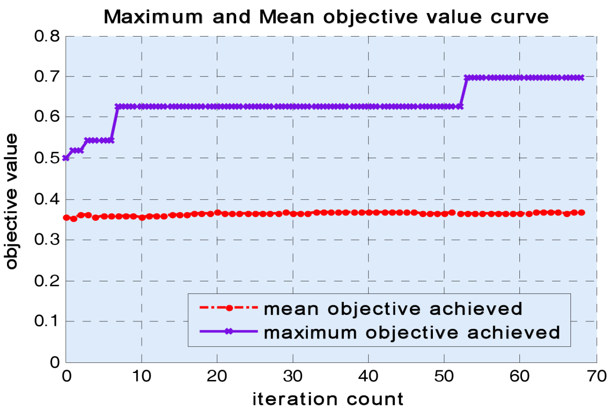

Fig. 3. Convergence curve obtained from the heuristic approach for problem #1

For the example problem the heuristic approach is executed for 68 iterations. The size of the set of the generated solutions is considered as 300. The maximum objective value obtained is 69.56. This proposed method is computationally efficient. It took 18.26 sec. to attain the best solution.

Computational results

The HACCF is tested with a set of 15 problems that have been published in the literature and have been widely used in many comparative studies. All the data sets were transcribed from the original articles to avoid the inconsistencies in data. The sources of the problems are shown in Table 4. The HACCF is coded in Matlab 7.0 environment and is tested on a laptop with a 2.1 GHz processor and 2 GB of RAM.

For the problems solved with HACCF to obtain optimal solution, the grouping efficacy value is better or equal in most instances. This observation indicates that this hybrid clustering technique is very efficient and moderately complex because of its simplicity in implementation (execution time $<$ 60 seconds for the largest (40×100) dataset tested). Therefore it indicates lesser computational resource utilization which in turn proves its computational efficiency. Therefore this technique is comparable with complex clustering heuristic and metaheuristic techniques.

Comparisons of the HACCF technique against other algorithms from the literature are given in Table 4. These other algorithms include ZODIAC [9], GRAFICS [10], GATSP-Genetic Algorithm [44], GA-Genetic Algorithm [45]. The HACCF is shown to outperform the standard techniques in 11 instances, and equal in 4 instances, which further depicts 73.33% improved result which is significant in terms of solution quality, time and space complexities. Figure 4 demonstrates the sharp dominance of HACCF over the aforementioned established methods (x-axis denotes grouping efficacy values and y-axis presents 15 test problems) and some of the improved solution matrices obtained by HACCF are presented in Fig. 5 to Fig. 9 for problem datasets #1, 2, 7, 12, and 13 which depict the reduction in number of EEs and voids.

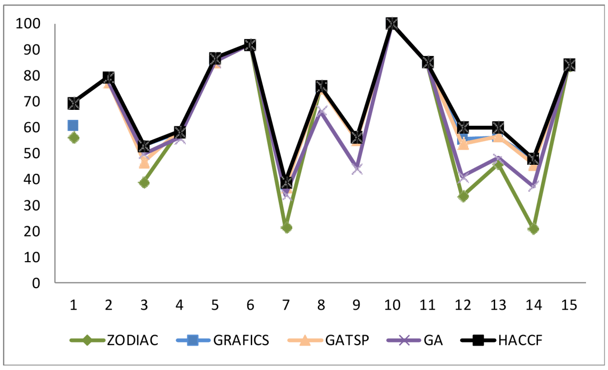

Fig. 4. Improvement shown by HACCF over other GA, GATSP, GRAFICS, ZODIAC for the 15 test problems.

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

Fig. 5. Solution matrix of dataset #1 (**grouping efficacy value = 69.56**).

	m1	m4	m2	m3	m5
p1	1	1	1		
p3	1	1	1		
p6	1	1	1		
p8	1	1	1		
p11	1	1	1		
p12	1	1	1		
p13	1	1	1		
p2	1	1			
p5	1	1			
p14	1	1			
p16	1	1			
p17	1	1			
p4			1	1	1
p7			1	1	
p10			1	1	1
p15			1	1	1
p18			1	1	1
p9					1

Fig. 6. Solution matrix of dataset #2 (**grouping efficacy value = 79.59**).

	m2	m3	m5	m1	m7	m8	m11	m12	m16	m17	m20	m4	m15	m10	m14	m18	m19	m9	m13	m6
p15		1	1		1				1									1		
p2	1	1	1			1				1										
p20		1		1		1						1					1			
p18				1	1	1			1											
p1				1			1		1									1	1	1
p12							1	1		1		1			1	1				
p13	1						1		1	1				1		1			1	
p16			1				1		1						1					
p17			1		1				1			1	1		1					
p5		1											1						1	
p8											1	1	1	1	1	1		1		1
p4														1				1		1
p10							1						1	1			1		1	
p19														1	1					
p6			1												1					1
p3		1				1	1					1			1	1			1	
p7				1	1				1		1				1	1	1	1		1
p11			1									1	1		1	1	1		1	
p9									1						1		1	1		1
p14										1					1	1	1		1	

Fig. 7. Solution matrix of dataset #7 (**grouping efficacy value = 39.23**).

	p10	p33	p41	p31	p32	p39	p12	p23	p2	p16	p27	p34	p36	p40	p11	p18	p19	p24	p1	p3	p21	p22	p30	p13	p6	p8	p38	p20	p15	p25	p28	p7	p35	p17	p9	p14	p29	p5	p4	p26	p37
m1	1	1	1	1	1	1																																			
m11	1	1	1			1																																			
m2		1	1		1	1	1								1	1																									
m3				1	1	1	1	1						1																											
m10						1	1	1	1																			1													
m22	1			1	1	1	1	1						1																											
m23				1		1	1	1	1																																
m7										1	1	1	1																												
m17										1	1		1																			1									
m18										1	1	1																								1					
m26										1	1	1	1																												
m12						1	1	1						1	1	1	1	1																							
m21						1	1	1						1	1	1	1	1																							
n19																			1	1	1	1	1																		
m20																			1	1	1	1																			
m29																			1	1	1		1	1											1						
m9																				1	1			1																	
m30																			1	1	1	1																			
m6																									1															1	
m16																									1						1										
m24																											1														
m27																										1			1												
m13																1											1			1											
m28																										1							1				1				
m4																		1											1	1	1	1	1								
m15																																		1							1
m25																																							1		
m5												1	1																					1				1			
m8																			1							1									1	1	1				
m14																1																						1	1	1	1

Fig. 8. Solution matrix of dataset #12 (**grouping efficacy value = 60.13**).

	p1	p4	p3	p6	p7	p11	p12	p13	p14	p10	p2	p5	p8	p9	p15	p16	p17	p18	p19	p21	p22	p27	p25	p23	p20	p24	p26	p28	p29	p30	p31	p32	p33	p34	p35	p36	p37	p38	p39	p40	p41	p42	p43	p44	p45	p46	p47	p48	p49	p50
m1	1	1	1	1											1																																			
m11	1	1	1		1																																													
m5			1		1	1	1	1	1																																									
m7			1			1	1	1	1	1		1																																						
m10				1			1	1	1			1																																						
m3	1											1			1																																			
m4	1											1	1	1																																				
m6					1	1				1	1																																							
m9	1										1	1			1																																			
m2		1		1							1			1																																				
m8		1			1						1	1	1	1	1																																			
m14																1			1	1			1																											
m15																		1	1		1	1					1																							
m20																1	1		1	1			1	1																										
m19																1	1	1			1	1	1																											
m12																																																		
m13																																																		
m17																	1	1		1						1	1																							
m16																								1	1		1																							
m18																1					1				1	1																								
m21																												1	1	1																				
m22																												1	1	1																				
m23																															1	1	1	1																
m24																															1	1	1	1																
m25																																			1	1	1	1	1											
m26																																			1		1	1												
m27																																								1	1	1	1	1	1	1				
m28																																									1	1	1			1				
m29																																															1	1	1	
m30																																															1	1	1	1

Fig. 9. Solution matrix of dataset #13 (**grouping efficacy value = 60.12**).

Conclusions

This study proposes a hybrid clustering technique called HACCF that combines a simple but efficient heuristic with a hierarchical agglomerative clustering technique. Performance measure utilized in this research is the grouping efficacy measure with an objective to minimize the intercell and intracell material flow. Computational results are presented in previous section demonstrate that the HACCF outperforms not only the standard clustering techniques, but also several other well-known soft computing based cell formation solution methodologies such as simple genetic algorithms, GA-TSP from the literature. This article depicts that, juxtaposition of a simple heuristic algorithm into the traditional hierarchical clustering technique could improve the solution quality substantially. The HACCF attains better quality solutions by consuming lesser computational time and resources than that of the traditional complex soft computing based methodologies. Further work can be done to utilize the technique in more complex and generalized cell formation problem which deals with ratio data for production volume, operational time, worker assignment and other multi-objectives. Future modification can also be incorporated by introducing similarity coefficient method in pair-wise similarity computation and also improving the part grouping section of the algorithm by practicing classification and coding approach with some recent evolutionary techniques such as Particle Swarm Optimization (PSO) or Bees Algorithm.

References

[1] Burbidge J.L., "Production flow Analysis", in Production Engineer, 42, 12 742–752, 1963.

[2] Burbidge J.L., "Production flow Analysis", in Production Engineer, 50, 139–152, 1971.

[3] Burbidge J.L., "Production Flow Analysis For Planning Group Technology", Oxford University Press, USA, pp. 192, 1996.

[4] Lee-Post A., "Part family identification using a simple genetic algorithm", International Journal of Production Research, 38, 793–810, 2000.

[5] Selim M.S., Askin R.G., and Vakharia A.J., "Cell formation in group technology: review evaluation and directions for future research", in Computers & Industrial Engineering, 34, 1, 3–20, 1998.

[6] Papaioannou G. and Wilson J.M., "The evolution of cell formation problem methodologies based on recent studies (1997–2008): Review and directions for future research", in European Journal of Operational Research, 206, 3, 509–521, 2010.

[7] Unler A. and Gungor Z., "Applying K-harmonic means clustering to the part-machine classification problem", in Expert Systems with Applications, 36, 1179–1194, 2009.

[8] Ghosh T., Sengupta S., Chattopadhyay M., and Dan P.K., "Meta-heuristics in cellular manufacturing: A state-of-the-art review", in International Journal of Industrial Engineering Computations, 2, 1, 87–122, 2011.

[9] Chandrasekharan M.P. and Rajagopalan R., "ZODIAC: an algorithm for concurrent formation of part-families and machine-cells", in International Journal of Production Research, 25, 6, 835–850, 1987.

[10] Srinivasan G. and Narendran T.T., "GRAFICS: a non-hierarchical clustering algorithm for group technology", in International Journal of Production Research, 29, 3, 463–478, 1991.

[11] Srinivasan G., "A clustering algorithm for machine cell formation in group technology using minimum spanning trees", in International Journal of Production Research, 32, 2149–2158, 1994.

[12] Lozano S., Dobado D., Larraneta J., and Onieva L., "Modified fuzzy C-means algorithm for cellular manufacturing", in Fuzzy Sets and Systems, 126, 23–32, 2002.

[13] McAuley J., "Machine grouping for efficient production", in Production Engineer, 51, 2, 53–57, 1972.

[14] Prabhakaran G., Janakiraman T.N., and Sachithanandam M., "Manufacturing data based combined dissimilarity coefficient for machine cell formation", in International Journal of Advanced Manufacturing Technology, 19, 889–897, 2002.

[15] Seifoddini H., Wolfe P.M., "Application of the similarity coefficient method in group technology", in IIE Transactions, 183, 271–277, 1986.

[16] McCormick W.T., Schweitzer J.P.J., and White T.W., "Problem decomposition and data reorganization by a clustering technique", in Operations Research, 20, 5, 993–1009, 1972.

[17] Carrie A.S., "Numerical taxonomy applied to group technology and plant layout", in International Journal of Production Research, 11, 4, 399–416, 1973.

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

[19] King J.R. and Nakornchai V., "Machine-component group formation in group technology: review and

extension", in International Journal of Production Research, 20, 117–133, 1982.

[20] Chan H.M. and Milner D., "Direct Clustering algorithm for group formation in Cellular Manufacturing", in Journal of Manufacturing Systems, 1, 65–75, 1982.

[21] Waghodekar P.H. and Sahu S., "Machine-component cell formation in group technology", in International Journal of Production Research, 22, 6, 937–948, 1984.

[22] Kusiak A., "The part families problem in flexible manufacturing systems", in Annals of Operations Research, 3, 279–300, 1985.

[23] Mosier C.T. and Taube L., "Weighted similarity measure heuristics for the group technology machine clustering problem", in OMEGA, 13, 6, 577–583, 1985.

[24] Stanfel L.E., "Machine clustering for economic production", in Engineering Costs and Production Economics, 9, 73–81, 1985.

[25] Chandrasekharan M.P. and Rajagopalan R., "An ideal seed non-hierarchical clustering algorithm for cellular manufacturing", in International Journal of Production Research, 24, 2, 451–464, 1986a.

[26] Chandrasekharan M.P. and Rajagopalan R., "MODROC: An extension of rank order clustering for group technology", in International Journal of Production Research, 24, 5, 1221–1233, 1986b.

[27] Kusiak A., "The generalized group technology concept", in International Journal of Production Research, 25, 4, 561–569, 1987.

[28] Seifoddini H., "Duplication process in machine cells formation in group technology", in IIE Transactions, 21, 382–388, 1989.

[29] Chu C.H. and Tsai M., "A Comparison of three array-based clustering techniques for manufacturing cell formation", in International Journal of Production Research, 28, 8, 1417–1433, 1990.

[30] Shafer S.M. and Rogers D.F., "Similarity and distance measures for cellular manufacturing", in International Journal of Production Research, 31, 5, 1131–1142, 1993.

[31] Dimopoulos C. and Mort N., "A hierarchical clustering methodology based on genetic programming for the solution of simple cell-formation problems", in International Journal of Production Research, 39, 1, 1–19, 2001.

[32] Rajagopalan R. and Batra J.L., "Design of Cellular production system – A graph theoretic approach", in International Journal of Production Research, 13, 6, 567–579, 1975.

[33] Ballakur A. and Steudel H.J., "A with-in cell utilization based heuristic for designing cellular manufacturing systems", in International Journal of Production Research, 25, 5, 639–655, 1987.

[34] Vohra T., Cheng D. and Chang J.L., "A network model for cell formation in CMS", in International Journal of Production Research, 28, 11, 2075–2084, 1990.

[35] Veeramani D. and Mani K., "A polynomial-time algorithm for optimal clustering in a special class of $\{0, 1\}$-matrices", in International journal of production research, 34, 9, 2587–2611, 1996.

[36] Tavakkoli-Moghaddam R., Aryanezhad M.B., Safaei N. and Azaron A., "Solving a dynamic cell formation problem using meta-heuristics", Applied Mathematics and Computation, 170, 761–780, 2005.

[37] Sarker B.R. and Mondal S., "Grouping efficiency measures in cellular manufacturing: a survey and critical review", in International Journal of Production Research, 37, 2, 285–314, 1999.

[38] Kumar S.C. and Chandrasekharan M.P., "Grouping efficacy: a quantitative criterion for goodness of block diagonal forms of binary matrices in group technology", in International Journal of Production Research, 28, 2, 233–243, 1990.

[39] Anderberg M.R., "Cluster Analysis for Applications", Academic Press Inc., New York, 1973.

[40] Kusiak A. and Chow W.S., "Efficient solving of the group technology problem", in Journal of Manufacturing Systems, 6, 2, 117–124, 1987.

[41] Boe W.J. and Cheng C.H., "A close neighbor algorithm for designing cellular manufacturing systems", in International Journal of Production Research, 29, 10, 2097–2116, 1991.

[42] Chandrasekharan M.P. and Rajagopalan R., "GROUPABILITY: an analysis of the properties of binary data matrices for group technology", in International Journal of Production Research, 27, 6, 1035–1052, 1989.

[43] Kumar K.R. and Vannelli A., "Strategic subcontracting for efficient disaggregated manufacturing", in International Journal of Production Research, 25, 12, 1715–1728, 1987.

[44] Cheng C.H., Gupta Y.P., Lee W.H. and Wong K.F., "A TSP-based heuristic for forming machine groups and part families", in International Journal of Production Research, 36, 5, 1325–1337, 1998.

[45] Onwubolu G.C. and Mutingi M., "A genetic algorithm approach to cellular manufacturing systems", in Computers & Industrial Engineering, 39, 125–144, 2001.

[46] Davidson I. and Ravi S.S., "Agglomerative Hierarchical Clustering with Constraints: Theoretical and Empirical Results", in Knowledge Discovery in Databases: PKDD 2005, Lecture Notes in Computer Science, 3721, 59–70, 2005.

[47] Tzacheva A.A., "Summaries of Action Rules by Agglomerative Clustering", in Advances in Intelligent Information Systems, Studies in Computational Intelligence, 265, 259–271, 2010.

[48] Boulif M. and Atif K., "A new branch-&-bound-enhanced genetic algorithm for the manufacturing cell formation problem", Computers & Operations Research, 33, 2219–2245, 2006.

[49] Boulif M. and Atif K., "A new fuzzy genetic algorithm for the dynamic bi-objective cell formation problem considering passive and active strategies", International Journal of Approximate Reasoning, 47, 141–165, 2008.

[50] Defersha F.M. and Chen M., "Machine cell formation using a mathematical model and a genetic-algorithm-based heuristic", International Journal of Production Research, 44, 12, 2421–2444, 2006.

[51] Defersha F.M. and Chen M., "A parallel genetic algorithm for dynamic cell formation in cellular manufacturing systems", International Journal of Production Research, 46, 22, 6389–6413, 2008a.

[52] Defersha F.M. and Chen M., "A linear programming embedded genetic algorithm for an integrated cell formation and lot sizing considering product quality", European Journal of Operational Research, 187, 46–69, 2008b.

[53] Neto A.R.P. and Filho E.V.G., "A simulation-based evolutionary multiobjective approach to manufacturing cell formation", Computers & Industrial Engineering, 59, 64–74, 2010.

[54] Deljoo V., Al-e-hashem S.M.J.M., Deljoo F. and Aryanezhad M.B., "Using genetic algorithm to solve dynamic cell formation problem", Applied Mathematical Modelling, 34, 1078–1092, 2010.

[55] Wu T.H., Chung S.H. and Chang C.C., "Hybrid simulated annealing algorithm with mutation operator to the cell formation problem with alternative process routings", Expert Systems with Applications, 36, 3652–3661, 2009.

[56] Durán O., Rodriguez N. and Consalter L.A., "Collaborative particle swarm optimization with a data mining technique for manufacturing cell design", Expert Systems with Applications, 37, 1563–1567, 2010.

[57] Anvari M., Mehrabad M.S. and Barzinpour F., "Machine–part cell formation using a hybrid particle swarm optimization", International Journal of Advanced Manufacturing Technology, 47, 745–754, 2010.

[58] Arkat J., Saidi M. and Abbasi B., "Applying simulated annealing to cellular manufacturing system design", International Journal of Advanced Manufacturing Technology, 32, 531–536, 2007.

[59] Ateme-Nguema B.H. and Dao T.M., "Optimization of cellular manufacturing systems design using the hybrid approach based on the ant colony and tabu search techniques", Proceedings of the IEEE IEEM, pp. 668–673, 2007.

[60] Ateme-Nguema B.H. an Daod T.M., "Quantized Hopfield networks and tabu search for manufacturing cell formation problems", International Journal of Production Economics, 121, 88–98, 2009.