Hybrid Genetic Algorithm for Machine-Component Cell Formation

Murugaiyan Pachayappan, Ramasamy Panneerselvam

Department of Management Studies, School of Management, Pondicherry University, Pondicherry, India
Email: panneer_dms@yahoo.co.in

Abstract

This paper considers machine-component cell formation problem of cellular manufacturing system. Since this problem comes under combinatorial category, development of a meta-heuristic is a must. In this paper, a hybrid genetic algorithm is presented. Normally, in genetic algorithm, the initial population is generated by random assignment of genes in each of the chromosomes. In this paper, the initial population is created using ideal seed heuristic. The proposed algorithm is compared with four other algorithms using 28 problems from literature. Through a completed factorial experiment, it is observed that the proposed algorithm outperforms the other algorithms in terms of grouping efficiency as well as grouping efficacy.

Keywords

1. Introduction

The productivity of an organization is very much affected by the type of layout and the design of the layout used. In reality, there are four different layouts, viz. process layout, product layout, cellular layout and fixed position layout. The cellular layout combines the benefits of process layout and product layout. In this paper, the cellular layout is considered.

The objective of the cellular layout design is to group a given number of components which uses a set of machines into a distinct machine-component cells such that the processing requirements of the components assigned to each machine cell are fully met within that cell itself. This problem comes under combinatorial category. Hence, the usage of heuristics is inevitable.

Several methods have been developed to solve this cell formation problem, like mathematical programming

approach (Mahdavi *et al*. [1], Khaksar-Haghani *et al*. [2], Arkat *et al* [3]), meta-heuristic approach includes Genetic Algorithm (GA) (Saeedi *et al* [4], Banerjee and Das [5], Arkat *et al* [6], Yin and Khoo [7], Ozcelik and Sarac [8], Sarac and Ozcelik [9]), Simulated Annealing (SA) (Wu *et al*. [10], Lin *et al*. [11], Paydar *et al*. [12], Kia *et al*. [13]) and Hybrid heuristics (Rezaeian *et al*. [14], Ghezavati and Saidi-Mehrabad [15], Elbenani and Ferland [16], Rafiei and Ghodsi [17], Paydar and Saidi-Mehrabad [18], Dalfard [19]). Nevertheless, the complexity of the problem and the significant issues involved in obtaining the result create necessity for more effective algorithms.

In this paper, a hybrid genetic algorithm is developed to obtain machine-component cells to maximize each of the measures, viz. grouping efficiency and grouping efficacy.

2. Literature Review

As there are many methodologies to solve machine-component cell formation problem, generating a meaningful machine groups and component families using an algorithm gives a good start to obtain the best solution. Srinivasan and Narendran [20] proposed a non-hierarchical clustering algorithm in which initial seed is generated by maximum density rule. Srinivasn [21] used minimum spanning tree (MST) method to identify the initial seed for the machine-component cell formation problem from the given machine-component incidence matrix. A comparative study is carried out by Miltenburg and Zhang [22] and reported that the quality of the solution obtained by the seed generation algorithms is better than that of ROC (Rank Order Cluster) algorithm, SLINK or ALINK. Kao and Li [23] initially generated component seeds by applying the ant colony recognition algorithm and used the seed to obtain a complete block diagonal matrix. Nambirajan and Panneerselvam [24] developed a simulated annealing algorithm for the machine-component cell formation problem and compared its results with a set of existing algorithms and found that their algorithm surpassed the other algorithms in terms of grouping efficiency as well as grouping efficacy.

The genetic algorithm proposed by Holland [25] has applications in optimization of different problems of reality. Its application has been formalized by Goldberg [26]. This method is further implemented by Venugopal and Narendran [27] to handle multi-objective cell formation problem in which the volume of inter-cell moves and the total within cell moves are minimized. Balakrishna and Jog [28] initially formed the similarity index matrix and then proposed parallel genetic algorithm to solve the cell formation problem. Joines *et al*. [29], Su and Hsu [30] and Mahdavi *et al*. [31] proposed a mathematical model and then implemented a genetic algorithm to minimize the inter-cell moves. Many researchers (James *et al*. [32], Kelling *et al*. [33] and Tunnukij and Hicks [34]) developed hybrid-genetic algorithm and measured the quality of the machine-component cell formation by grouping measures, viz. grouping efficiency, grouping efficacy, grouping index, etc. Tariq *et al*. [35] and Pailla *et al*. [36] applied genetic algorithm (GA) with local search heuristic (LSH) to maximize the machine utilization and to minimize the inter-cell moves. Banerjee and Das [5] investigated the cell formation problem by Predator-Prey Genetic Algorithm by local selection strategy. Sarac and Ozcelik [9] introduced three different selection and crossover operations and tested the performance of proposed algorithm with an existing algorithm using 15 test problems. Vin and Delchambre [37] defined the generalized cell formation problem and developed a grouping genetic algorithm to solve it. Li *et al*. [38] developed a genetic algorithm for virtual cell reconfiguration problem. Deep and Sing [39] designed a cellular manufacturing system for a dynamic environment. They first developed a mathematical model for this problem and then developed a genetic algorithm to solve this problem.

Among different algorithms reviewed in this paper, the genetic algorithm is considered for further improvement, because the population used in this algorithm consists of chromosomes whose genes are generated randomly. This amounts to searching the entire solution space of the machine-component cell formation problem.

3. Problem Definition

Let $C\left(C_1, C_2, \cdots, C_j, \cdots, C_n\right)$ be the set of components to be manufactured and $M\left(M_1, M_2, \cdots, M_i, \cdots, M_m\right)$ be the set of machines which are required to manufacture the given set of components. Here, n is the total number of component types and m is the total number of machine types.

Let $a_{ij} = 1$, if the component j requires processing on machine i;
     = 0, otherwise.
         for $i = 1, 2, 3, 4, \cdots, m$ and $j = 1, 2, 3, 4, \cdots, n$.

A sample machine-component incidence matrix $\left(a_{ij}\right)$ is shown in **Figure 1**. The rearrangement of rows and columns of this matrix gives an alternate form of machine-component incident matrix as shown in **Figure 2**.

The machine-component incidence matrix shown in **Figure 1** consists of 5 machines and 7 components. This is converted to a form as shown in **Figure 2**, which shows a block diagonal form with two odd elements in the off-diagonal blocks. This solution has two machine-component cells. The machines 1 and 4 and the components 6, 2 and 4 form the machine-component cell 1. Similarly the machines 2, 3 and 5 and the components 5, 7, 1 and 3 form the machine-component cell 2. From **Figure 2**, it is clear that there are two exceptional elements, which are with respect to machine 3 and component 6 and machine 1 and component 5. So, the exceptional element of component 6 should be processed in machine-component cell 2, because the machine 3 is assigned to it. Similarly the exceptional elements of component 5 should be processed in machine-component cell 1. In this process, the component 6 as well as the component 5 moves to another cell to process missing operation. Such movements between machine-component cells are called inter-cell moves which should be minimized

The objective of the design of the cellular layout is to obtain a block diagonal form such that the desired measures of performance are optimized. The grouping efficiency and grouping efficacy are the well known measures of performance of the cellular layout, whose formulas are as given below.

The grouping efficiency is introduced by Chandrasekharan and Rajagopalan [40] [41] to define the quality of the solution, named as grouping efficiency $\left(\eta\right)$, which is the weighted sum of two functions as given below.

$$\eta = q\eta_1 + \left(1-q\right)\eta_2$$

where η_1 is the ratio of number of 1^s in the diagonal blocks to the total number of elements in the diagonal blocks.

η_2 is the ratio of number of 0^s in the off-diagonal blocks to the total number of elements in the off-diagonal blocks.

q is a weighting factor $\left(0 \leq q \leq 1\right)$ and it is usually assumed as 0.5.

Kumar and Chandrasekaran [42] proposed another measure namely grouping efficacy $\left(E\right)$ to overcome the weaker discriminating power of grouping efficiency measure by assigning equal weight for the number of voids and the number of exceptional elements. This measure is defined as follows.

$$E = \frac{e - e_0}{e - e_v}$$

where e: Total number of 1s in the matrix.
e_0: The number of exceptional elements.
e_v: The number of voids in the diagonal box.

	c_1	c_2	c_3	c_4	c_5	c_6	c_7
m_1	0	1	0	1	1	1	0
m_2	1	0	1	0	0	0	0
m_3	1	0	1	0	0	1	1
m_4	0	1	0	1	0	1	0
m_5	1	0	0	0	1	0	1

Figure 1. Sample machine-component incidence matrix.

	c_6	c_2	c_4	c_5	c_7	c_1	c_3
m_1	1	1	1	1	0	0	0
m_4	1	1	1	0	0	0	0
m_2	0	0	0	0	0	1	1
m_3	1	0	0	0	1	1	1
m_5	0	0	0	1	1	1	0

Figure 2. Rearranged machine-component matrix (block-diagonal matrix).

In this paper, these two measures are considered to measure the grouping accuracy of machine-component cell formation.

4. Fundamentals of Genetic Algorithm

Genetic algorithm (GA) applied to the machine-component cell formation problem involves the following.
- Representation of the genes in chromosomes;
- Selection of chromosomes for crossover;
- Crossover and mutation operations;
- Repair strategy.

In the GA algorithm proposed in this paper, an initial population is generated randomly and the fitness function value of each of the chromosomes is then evaluated. Then, the processes of crossover and mutation are applied over the chromosomes of a subpopulation to produce their offspring. Then their fitness function values are evaluated. Then these offsprings replace the corresponding chromosomes in the population. This process is repeated for a specified number of iterations (generations) and finally the best fitness function value among the machine-component cell formations with respect to the top most machine chromosome and component chromosome of all the generations is selected as the final solution and the corresponding machine-component cell formation result is suggested for implementation.

4.1. Representation and Selection of Chromosomes

In this paper, each chromosome is represented based on the representation followed by Venugopal and Narendran's [27]. The chromosomes are classified into two types, viz. machine chromosomes and component chromosomes.

Gene positions from left to right in a machine chromosome represent the machine numbers from low to high, respectively. The number of genes in a machine chromosome is equal to the number of machines. The gene value at a particular gene position of a machine chromosome represents the cell number to which the corresponding machine is assigned.

Similarly, gene positions from left to right in a component chromosome represent the component numbers from low to high, respectively. The number of genes in a component chromosome is equal to the number of components. The gene value at a particular gene position of a component chromosome represents the cell number to which the corresponding component is assigned.

4.2. Two-Point Crossover Method

The genetic algorithm uses two-point crossover method for machine chromosomes as well as component chromosomes. Consider two machine chromosomes as shown in **Figure 3**. The chromosomes consist of eight genes. The number of machine cells 4. So, the gene values in both the chromosomes are in between 1 and 4, both inclusive.

Now, randomly generate two crossover points and let them be 3 and 6. The offsprings based on the two-point crossover method are shown in **Figure 4**.

Gen position	1	2	3	4	5	6	7	8
Machine chromosome 1	1	1	2	4	1	3	2	1
Machine chromosome 2	2	2	3	1	4	2	1	4

Figure 3. Machine chromosomes.

Gen position	1	2	3	4	5	6	7	8
Machine offspring 1	1	1	3	1	4	2	2	1
Machine offspring 2	2	2	2	4	1	3	1	4

Figure 4. Machine chromosomes.

In **Figure 4**, the machine offspring 1 is obtained by copying the genes at gene positions 1 and 2 of the machine chromosome 1 and the genes at the gene positions 3, 4, 5 and 6 of the machine chromosome 2 and the genes at the gene positions 7 and 8 of the machine chromosome 1. Similarly, the machine offspring 2 is obtained by copying the genes at the gene positions 1 and 2 of the machine chromosome 2, genes at the gene positions 3, 4, 5 and 6 of the machine chromosome 1 and the genes at the gene positions 7 and 8 of the machine chromosome 2.

The consecutive pairs of machine chromosomes are taken at a time and then the two-point crossover method is applied on them to obtain their corresponding offspring. The method is also used for the component chromosomes.

4.3. Mutation

Mutation is the process of randomizing the genes by swapping the genes at two randomly selected positions of each offspring obtained after crossover of two chromosomes. This is done based on a given probability for mutation. If the generated probability is less than or equal, say 0.30, then mutation will be performed; otherwise, the mutation will be not performed on the offspring.

4.4. Repair Strategy for Offspring

The offspring obtained after the crossover and mutation operations may be infeasible or ill-structured. Under such situation, a new repair strategy is applied to obtain a feasible offspring.

Let a sample new machine offspring after the crossover and mutation operations be as shown below.

New machine offspring: 2   4   1   4   1   2

In the machine offspring, the machine cell numbers are 1, 2 and 4. In this distribution of machine cell numbers, the machine cell 3 is missing, which makes the machine offspring infeasible. Hence, the cell numbers must be modified such that they are in continuous order starting from 1.

Start from left of the offspring and make the first gene to 1 and then wherever the gene value is equal to the gene value in the first position of the offspring, make it to 1.

Then move to next non-updated gene location (L) and make it to 2 and then wherever the non-updated gene value is equal to the gene value at the gene location L, make it to 2. Continue this process, until all the gene values are updated.

The resultant repaired offspring is as shown below.

Repaired offspring: 1   2   3   2   3   1

Similarly logic can be applied to repair any component offspring, if there is discontinuity in component cell numbers.

4.5. Density Index of Matching Machine Groups with Component Families

While applying the genetic algorithm to the machine-component cell formation problem, after forming machine cells and component families, next these must be matched based on certain measure. In existing researches, there is no specific method to perform this step.

In this paper, the machine cells and component families are matched based on a measure called "density index". The computation of the density index by assigning the machine group i to the component family j is explained using some sample result shown in **Figure 5**.

Actually, in **Figure 5**, the machine cells are appropriately matched with the component families to give a sample block diagonal form. The fitness value is then calculated for this block diagonal matrix. The grouping efficiency and the grouping efficacy of this machine-component cell formation are 61.90% and 34.28%, respectively. The summary of machine-component cells is given in **Table 1**.

Each machine cell can be assigned to each component cell. As an example if the machine cell 1 (MC1), which consists of machines m_1 and m_6 is assigned to the component cell 1 (CC1), which consists of components c_1 and c_5, then the corresponding density index D_{11} is computed using the following formula. This value and other values of the density matrix are shown in **Figure 6**.

$$D_{11} = \frac{\text{Number of 1s in the machine-component cell w.r.t. MC1 and CC1}}{\text{Size of the machine-component cell w.r.t. MC1 and CC1}} = \frac{3}{4} = 0.75$$

After applying Hungarian method designed for the assignment problem, which is proposed in this paper to match the machine groups with component families, the grouping of machine cells and component families are changed. The new block diagonal form is obtained and shown in **Figure 7**. The grouping efficiency and grouping efficacy of this machine-component cell formation are 70.67% and 45.45%, respectively, which are improved from the respective previous values.

5. Hybrid Genetic Algorithm

In this section, a hybrid genetic algorithm (HGA) is presented to form machine-component cells for a given cellular manufacturing system.

	c_1	c_5	c_2	c_3	c_7	c_9	c_4	c_6	c_8
m_1	1	1	0	0	0	1	0	0	0
m_6	0	1	0	1	0	0	0	0	0
m_2	0	0	1	1	1	0	1	1	1
m_4	0	0	0	0	1	0	1	1	0
m_3	0	0	1	1	1	0	1	0	1
m_5	0	0	1	0	1	1	0	1	0
m_7	0	0	0	0	1	0	1	1	0

Figure 5. Block-diagonal matrix.

Table 1. Summary of machine component cells of **Figure 5**.

Machine Cell No.	Machines	Component Cell (Family) No.	Components
MC1	m1,m6	CC1	c_1,c_5
MC2	m2,m4	CC2	c_2,c_3,c_7,c_9
MC3	m3,m5,m7	CC3	c_4,c_6,c_8

	CC1	CC2	CC3
MC1	0.75	0.25	0
MC2	0	0.5	0.83
MC3	0	0.58	0.55

Figure 6. Density matrix.

	c_1	c_5	c_2	c_3	c_7	c_9	c_4	c_6	c_8
m_1	1	1	0	0	0	1	0	0	0
m_6	0	1	0	1	0	0	0	0	0
m_2	0	0	1	1	1	0	1	1	1
m_4	0	0	0	0	1	0	1	1	0
m_3	0	0	1	1	1	0	1	0	1
m_5	0	0	1	0	1	1	0	1	0
m_7	0	0	0	0	1	0	1	1	0

Figure 7. New block diagonal matrix.

The steps of the algorithm are presented below.

Step 1: Input the following.

- Number of machines (m);
- Number of components (n);
- Machine component incidence matrix $\left[a_{ij}\right]$;
- Size of the population of machine chromosomes (N1) as well as component chromosomes (N1) ;
- Set generation count (GC) = 1.
- Maximum number of generations to be carried out (MNG)

Step 2: Find the integer values of $m/2$ and $n/2$ rounded to next integer and find the minimum of them. Treat it as the maximum number of cells (MC).

Step 3: Generate an initial population of machine chromosomes by assigning the machines to different cells in the range from 1 to MC. A sample machine chromosome is shown in **Table 2** by assuming MC as 2.

Step 4: For each of the machine chromosomes, find ideal seed for machine chromosome as well as for component chromosome by performing the following.

4.1) Form machine groups. The machine groups of the machine chromosome shown in **Table 2** are given in **Table 3**.

4.2) Form component groups based on the machine groups by following the guidelines given below.

- Assign a component to a machine cell in which it has the maximum number of operations;
- In case of tie on the maximum number of operations, break the tie randomly.

Let the number of component groups be CG.

4.3) If the number of machine groups is equal to the number of component families, find the desired measure of performance: Grouping efficiency (or) Grouping efficacy and go to Step 5;

Otherwise, go to Step 4.4.

4.4) Form machine groups based on the component groups obtained in Step 4.2 by following the guidelines given below.

- Assign a machine to a component cell in which it processes maximum number of components;
- In case of tie on the maximum number of components processed by the machine, break the tie randomly.

Let the number of machine groups be MG.

4.5) If the number of machine groups is equal to the number of component groups, find the desired measure of performance: Grouping efficiency (or) Grouping efficacy and go to Step 5; Otherwise, go to Step 4.2.

Step 5: Sort the machine groups and component groups together in the decreasing order of their fitness function values, either grouping efficiency or grouping efficacy as the case may be.

Step 6: Select the top most 30% of the sorted population rounded to an even number and let the size of this subpopulation be N2.

Step 7: For each of the successive two machine chromosomes as well as for each of the successive two component chromosomes, perform the following.

7.1) Perform two-point crossover operation to obtain their corresponding two offspring.

7.2) Perform mutation with a mutation probability of 0.30 on each of the machine offsprings as well as component offsprings.

7.3) If some of the machine cell numbers are missing in a machine offspring, repair the respective machine offspring, which will modify the machine cell numbers so that they are continuous from the machine cell number 1 onwards.

Table 2. Sample machine chromosome.

Machine	1	2	3	4	5
Cell Number	2	1	1	2	2

Table 3. Machine groups of machine chromosome.

Machine Cell	Machines		
1	2	3	
2	1	4	5

7.4) If some of the component facility numbers are missing in a component offspring, repair the respective component offspring, which will modify the component family numbers so that they are continuous from the component family number 1 onwards.

7.5) Perform further repair on either machine offspring or component offspring to bring equality in the number of machine cells and the number of component families.

7.6) For each of the machine offspring and component offspring combination, perform the following.

7.6.1) Form density matrix $\left[D_{ij} \right]$, which represents density of 1^s in the sub matrix in the machine-component incidence matrix with respect to the ith machine cell and the jth component family, for $i = 1, 2, 3, \cdots, \text{MC}$ and $j = 1, 2, 3, \cdots, \text{MC}$.

7.6.2) Using Hungarian method which is designed for assignment problem, match the machine groups with the component families such that the resultant sum of the density values is maximized to obtain a machine-component cell formation.

7.6.3) Find the fitness function value (grouping efficiency or grouping efficacy) for the machine-component cell formation obtained in Step 7.6.2.

Step 8: Store the machine offspring and component offspring of the subpopulation in the respective chromosomes of the population.

Step 9: Sort the machine chromosomes and component chromosomes together in the decreasing order of their fitness function values, either grouping efficiency or grouping efficacy as the case may be.

Step 10: Increment the generation count by 1 (GC = GC + 1).

Step 11: If GC ≤ MNG then go to Step 6.

Step 12: Rework the results of the topmost machine chromosome and component chromosome by following the step 7.6 and print the corresponding machine-component cell formation and the grouping efficiency or grouping efficacy.

6. Comparison of HGA with Existing Algorithms

The performance of the hybrid genetic algorithm (HGA) presented in this research is compared with four other existing algorithms, viz. ZODIAC (Chandrasekharan and Rajagopalan [43]), GRAFICS (Srinivasan and Narendran [20]), Algorithm-1 (Nambirajan and Panneerselvam [44]) and Algortihm-2 (Nambirajan and Panneerselvam [24]). The HGA algorithm is coded in C++ and all experiments are executed on personal computer with i5 processor in Window-7 operating system.

A complete factorial experiment is designed to examine the effects of two factors, viz. "Problem Size" and "Algorithm" and their interaction on grouping efficiency as well as groping efficacy.

The number of levels of the factor "Problem Size" is 14 and they are in terms of "Number of machines" and "Number of components" are 5×7, 7×7, 8×20, 9×10, 10×12, 12×19, 14×24, 15×10, 20×20, 20×40, 24×40, 30×50, 40×20 and 40×40. The number of levels of the factor "Algorithm" is 5 (Proposed algorithm and four existing algorithms). The number of replications carried out under each of the experimental combinations is 2. So, the total number of observations of this experiment is 140 ($14 \times 5 \times 2$) for each of the grouping measures. The data for all the replications under each experimental combination are selected from literature, which are as cited in **Table 4** for grouping efficiency as well as in **Table 5** for grouping efficacy.

The model of ANOVA is as given below.

$$Y_{ijk} = \mu + A_i + B_j + \left(AB \right)_{ij} + e_{(ij)k}$$

where Y_{ijk} is the kth replication under the ith treatment of the Factor A and the jth treatment of the Factor B.

μ is the overall mean of the response.

A_i is the effect of the ith treatment of the Factor A on the response.

B_j is the effect of the jth treatment of the Factor B on the response.

$\left(AB \right)_{ij}$ is the interaction effect of the ith treatment of the Factor A and the jth treatment of the Factor B on the response.

$e_{(ij)k}$ is the random error associated with the kth replication under the ith treatment of the Factor A and the jth treatment of the Factor B.

Table 4. Results of grouping efficiency.

S. No	Problem Size	Reference	REP	ZODIAC	GRAFICS	Algorithm-1	Algorithm-2	HGA
1	5 × 7	King and Nakornchai [45]	1	85.62	85.62	85.62	91.30	91.30
		Waghodekar and Sahu [46]	2	72.20	74.51	73.85	77.10	82.61
2	7 × 7	Nambirajan [47]	1	68.20	70.90	73.00	81.35	87.61
		Vohra *et al* [48]	2	74.34	75.09	79.97	82.14	92.11
3	8 × 20	Chandrasekharan, and Rajagopalan [40] [41]	1	95.83	95.83	95.83	95.83	95.83
		Chandrasekharan, and Rajagopalan [40] [41]	2	71.87	76.30	71.88	71.88	79.24
4	9 × 10	Choobineh [49]	1	81.63	81.63	87.51	87.51	93.06
		Safer *et al* [50]	2	77.11	77.11	77.11	83.84	88.16
5	10 × 12	Nambirajan [47]	1	71.31	74.41	80.73	88.54	89.63
		Nambirajan [47]	2	83.33	83.33	83.33	88.32	92.04
6	12 × 19	De Witte [51]	1	78.58	74.43	81.26	81.26	87.46
		Tam [52]	2	76.18	76.18	77.74	77.74	87.46
7	14 × 24	Askin and Subramanian [53]	1	82.54	82.54	82.54	85.14	89.51
		Stanfel [54]	2	83.90	83.90	83.90	84.84	91.32
8	15 × 10	Balasubramanian and Panneerselvam [55]	1	88.31	88.31	88.31	90.69	93.23
		Chan and Milner [56]	2	96.00	96.00	96.00	96.00	96.00
9	20 × 20	Mosier and Taube [57]	1	53.05	71.15	74.63	70.06	85.70
		Nambirajan [47]	2	69.57	73.96	75.09	78.98	83.43
10	20 × 40	Nambirajan [47]	1	72.17	77.44	76.02	79.95	82.99
		Nambirajan [47]	2	61.28	65.08	79.15	79.17	81.89
11	24 × 40	Chandrasekharan and Rajagopalan [58]	1	100.00	100.00	100.00	100.00	100.00
		Chandrasekharan and Rajagopalan [58]	2	91.16	91.16	91.16	91.82	93.43
12	30 × 50	Stanfel [54]	1	75.35	85.24	85.28	86.52	81.62
		Stanfel [54]	2	62.92	85.56	86.00	87.53	81.70
13	40 × 20	Nambirajan [47]	1	71.51	73.19	70.66	78.99	78.53
		Nambirajan [47]	2	58.91	64.47	79.75	79.55	80.05
14	40 × 40	Nambirajan [47]	1	60.15	71.96	83.52	84.07	79.97
		Nambirajan [47]	2	50.51	61.10	75.33	75.49	78.53

Table 5. Results of grouping efficacy.

S. No	Problem Size	Reference	REP	ZODIAC	GRAFICS	Algorithm-1	Algorithm-2	HGA
1	5 × 7	King and Nakornchai [45]	1	73.68	73.68	73.68	75.00	75.00
		Waghodekar and Sahu [46]	2	56.62	60.87	62.50	68.00	69.57
2	7 × 7	Nambirajan [47]	1	42.14	44.45	48.00	61.11	68.75
		Vohra *et al* [48]	2	52.91	53.33	60.87	65.22	70.00
3	8 × 20	Chandrasekharan, and Rajagopalan [40] [41]	1	85.24	85.24	85.24	85.24	85.25
		Chandrasekharan, and Rajagopalan [40] [41]	2	58.33	58.13	58.33	58.33	58.72
4	9 × 10	Choobineh [49]	1	62.16	62.16	64.52	64.52	65.71
		Safer *et al* [50]	2	52.63	52.63	52.63	57.58	57.58
5	10 × 12	Nambirajan [47]	1	46.61	48.65	55.17	73.08	76.92
		Nambirajan [47]	2	61.11	61.11	61.11	65.63	73.33
6	12 × 19	De Witte [51]	1	56.56	52.80	54.55	54.55	57.96
		Tam [52]	2	54.63	54.63	56.31	56.31	57.67
7	14 × 24	Askin and Subramanian [53]	1	64.36	64.36	64.36	68.75	73.24
		Stanfel [54]	2	65.55	65.55	67.05	67.86	71.62
8	15 × 10	Balasubramanian and Panneerselvam [55]	1	75.00	75.00	75.00	77.08	77.08
		Chan and Milner [56]	2	92.00	92.00	92.00	92.00	92.00
9	20 × 20	Mosier and Taube [57]	1	21.63	38.26	40.29	40.85	42.76
		Nambirajan [47]	2	39.27	42.55	43.33	46.91	57.93
10	20 × 40	Nambirajan [47]	1	42.38	47.11	45.24	48.65	59.05
		Nambirajan [47]	2	27.43	29.95	45.45	45.45	55.96
11	24 × 40	Chandrasekharan and Rajagopalan [58]	1	100.00	100.00	100.00	100.00	100.00
		Chandrasekharan and Rajagopalan [58]	2	37.85	73.51	73.51	72.97	72.87
12	30 × 50	Stanfel [54]	1	46.06	56.32	56.99	56.45	54.81
		Stanfel [54]	2	21.11	47.96	48.72	49.47	43.46
13	40 × 20	Nambirajan [47]	1	34.28	35.98	35.9	39.90	46.86
		Nambirajan [47]	2	29.13	31.79	39.15	38.62	48.23
14	40 × 40	Nambirajan [47]	1	27.30	31.31	35.69	36.57	35.48
		Nambirajan [47]	2	24.98	28.03	31.96	31.43	39.90

In this model, the Factor A_i (Problem size) is a random factor and the Factor B_j (Algorithm) is a fixed factor. Since, the Factor A is a random factor, the interaction AB_{ij} is also a random factor. The replications are always random and the number of replication under each experimental combination is k. The derivation of the expected mean square (EMS) formulas for this combination of factors is given in Panneerselvam [59]. To test the effect of A_i as well as AB_{ij}, the respective F ratio is formed by dividing the mean sum of squares of the respective component (A_i or AB_{ij}) by the mean sum of squares of error. The F ratio for the component B_j is formed by dividing its mean sum of squares by the mean sum of squares of AB_{ij}.

The alternative hypotheses of this model are as stated below.

H_1: There is a significant difference between the different pairs of treatments of the Factor A (Problem Size) in terms of grouping efficiency/grouping efficacy.

H_2: There is a significant difference between the different pairs of treatments of the Factor B (Algorithm) in terms of grouping efficiency/grouping efficacy.

H_3: There is a significant difference between the different pairs of interaction between Factor A and Factor B in terms of grouping efficiency/grouping efficacy.

6.1. Comparison of Algorithms Based on Grouping Efficiency

This section presents the comparison of algorithms based on grouping efficiency. The ANOVA results of the grouping efficiency measure shown in **Table 4** are shown in **Table 6**.

From the results shown in the **Table 6**, it is clear that the factor "Problem Size" and the factor "Algorithm" are having significant effect on the grouping efficiency.

Since there is significant difference between the algorithms in terms of grouping efficiency, next the best algorithm is obtained using Duncan's multiple range test.

The treatment means in terms of grouping efficiency with respect to the algorithm are shown in **Figure 8** in ascending order from left to right. The standard error is obtained using the following formula. One can notice the fact that the mean sum of squares of the interaction term AB is used in estimating the standard error, because the F ratio for the factor 'Algorithm" is obtained by dividing its mean sum of squares by the mean sum of squares of the interaction $\left(AB_{ij}\right)$ (Panneerselvam [59]).

$$\text{SE} = \left(\frac{\text{MSS}_{AB}}{n}\right)^{0.5} = \left(\frac{30.082^{0.5}}{28}\right) = 1.0365$$

By referring to Duncan's table (Panneerselvam [59]), at a significant level of 0.05, the Least Significant Range (LSR) for each of the pairs of treatments of the Factor B is computed and shown in **Figure 8** along with the actual difference between the means of that pair of treatments. From **Figure 8**, it is clear that the proposed HGA is significantly different from all other algorithms and it is superior to all of them in terms of grouping efficiency.

6.2. Comparison of Algorithms Based on Grouping Efficacy

This section presents the comparison of algorithms based on grouping efficacy. The ANOVA results of the grouping efficiency measure shown in **Table 5** are shown in **Table 7**.

From the results shown in the **Table 7**, it is clear that the factor "Problem Size" and the factor "Algorithm" are having significant effect on the grouping efficacy.

Table 6. ANOVA results of grouping efficiency measure.

| Source of Variation	Sum of Square	Degree of Freedom	Mean Sum of Squares	Formula for F ratio	$F_{\text{Calculated}}$	F_{Table} at $\alpha = 0.05$	Remarks
Problem (A)	6319.438	13	486.111	$\dfrac{\text{MSS}_A}{\sigma^2}$	11.889	1.863	Accept alternative hypothesis, H_1
Algorithm (B)	2307.438	4	576.859	$\dfrac{\text{MSS}_B}{\text{MSS}_{AB}}$	19.176	2.503	Accept alternative hypothesis, H_1
$A \times B$	1564.250	52	30.082	$\dfrac{\text{MSS}_{AB}}{\sigma^2}$	0.736	1.524	Reject alternative hypothesis, H_1
Error	2862.063	70	40.887				
Total	13053.190	139					

Table 7. ANOVA results of grouping efficacy.

Source of Variation	Sum of Square	Degree of Freedom	Mean Sum of Squares	Formula for F ratio	$F_{Calculated}$	F_{Table} at $\alpha = 0.05$	Remarks
Problem (A)	33359.350	13	2566.103	$\dfrac{MSS_A}{\sigma^2}$	22.676	1.863	Accept alternative hypothesis, H_1
Algorithm (B)	2337.375	4	584.344	$\dfrac{MSS_B}{MSS_{AB}}$	16.704	2.503	Accept alternative hypothesis, H_1
$A \times B$	1819.031	52	34.981	$\dfrac{MSS_{AB}}{\sigma^2}$	0.309	1.524	Reject alternative hypothesis, H_1
Error	7921.407	70	113.163				
Total	45437.160	139					
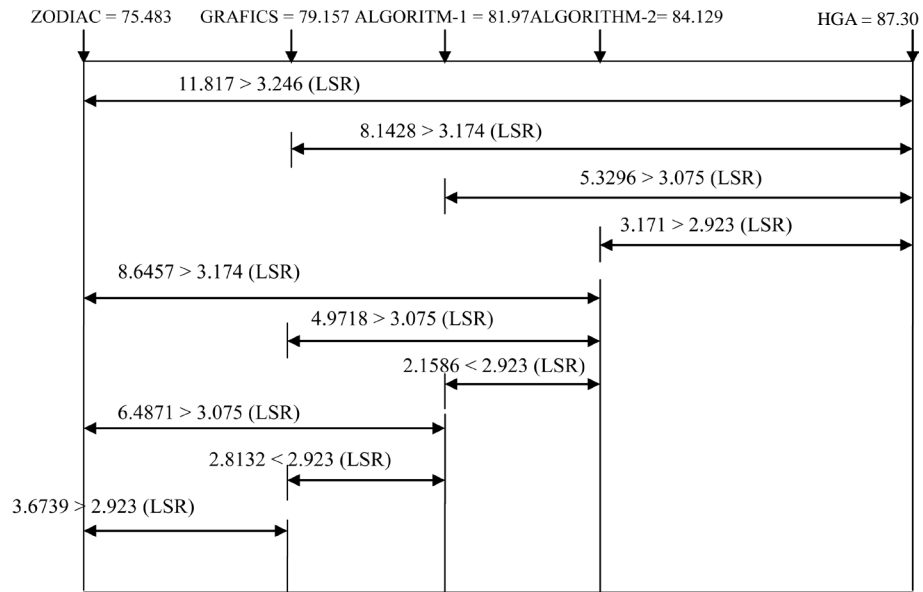
Figure 8. Duncan's multiple range tests for grouping efficiency.
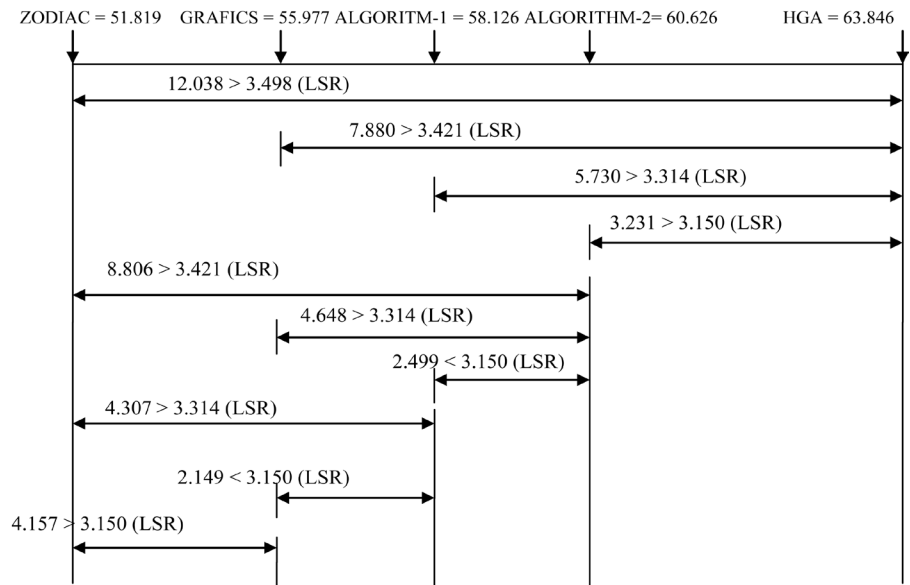
Figure 9. Duncan's multiple range tests for grouping efficacy.

Since there is significant difference between the algorithms in terms of grouping efficacy, next the best algorithm is obtained using Duncan's multiple range test.

The treatment means in terms of grouping efficacy with respect to the algorithm are shown in **Figure 9** in ascending order from left to right. The standard error for this performance measure is 1.1177.

By referring to Duncan's table (Panneerselvam [59]), at a significant level of 0.05, the Least Significant Range (LSR) for each of the pairs of treatments of the Factor *B* is computed and shown in **Figure 9** along with the actual difference between the means of the grouping efficacy of that pair of treatments. From **Figure 9**, it is clear that the proposed HGA is significantly different from all other algorithms and it is superior to all of them in terms of grouping efficacy.

7. Conclusion

The cellular manufacturing system helps companies to improve productivity by way of combining the benefits of process layout and product layout. In this paper, the design of machine-component cells for a given machine-component incidence matrix is attempted. Since this problem is a combinatorial problem, a hybrid genetic algorithm is developed to maximize each of the performance measures, viz. grouping efficiency and grouping efficacy. Then, it is compared with four existing algorithms for each of the performance measures using a complete factorial experiment, in which the problem is treated as one factor and the algorithm is treated as another factor. It is found that the proposed hybrid genetic algorithm surpasses the performances of all other four algorithms. The construction of machine chromosomes and component chromosomes, and also the use of Hungarian method to match the machine groups with component families are unique contribution of this research.

Acknowledgements

References

[1] Mahdavi, I., Paydar, M.M., Solimanpur, M. and Saidi-Mehrabad, M. (2010) A Mathematical Model for Integrating Cell Formation Problem with Machine Layout. *International Journal of Industrial Engineering & Production Research*, **21**, 61-70.

[2] Khaksar-Haghani, F., Kia, R., Javadian, N., Tavakkoli-Moghaddam, R. and Baboli, A. (2011) A Comprehensive Mathematical Model for the Design of a Dynamic Cellular Manufacturing System Integrated with Production Planning and Several Manufacturing Attributes. *International Journal of Industrial Engineering & Production Research*, **22**, 199-212.

[3] Arkat, J., Naseri, F. and Ahmadizar, F. (2011) A Stochastic Model for the Generalized Cell Formation Problem Considering Machine Reliability. *International Journal of Computer Integrated Manufacturing*, **24**, 1095-1102. http://dx.doi.org/10.1080/0951192X.2011.627944

[4] Saeedi, S., Solimanpur, M., Mahdavi, I. and Javadian, N. (2010) Heuristic Approaches for Cell Formation in Cellular Manufacturing. *Journal of Software Engineering & Applications*, **3**, 674-682. http://dx.doi.org/10.4236/jsea.2010.37077

[5] Banerjee, I. and Das, P. (2012) Group Technology Based Adaptive Cell Formation Using Predator-Prey Genetic Algorithm. *Applied Soft Computing*, **12**, 559-572. http://dx.doi.org/10.1016/j.asoc.2011.07.021

[6] Arkat. J., Hosseini, L. and Farahani, M.H. (2011) Minimization of Exceptional Elements and Voids in the Cell Formation Problem Using a Multi-Objective Genetic Algorithm. *Expert System with Applications*, **38**, 9597-9602. http://dx.doi.org/10.1016/j.eswa.2011.01.161

[7] Yin, X.F. and Khoo, L.P. (2011) An Exact Schema Theorem for Adaptive Genetic Algorithm and Its Application to Machine Cell Formation. *Expert Systems with Applications*, **38**, 8538-8552. http://dx.doi.org/10.1016/j.eswa.2011.01.055

[8] Ozcelik, F. and Sarac, T. (2012) A Genetic Algorithm Extended Modified Sub-Gradient Algorithm for Cell Formation Problem with Alternative Routings. *International Journal of Production Research*, **50**, 4025-4037. http://dx.doi.org/10.1080/00207543.2011.588264

[9] Saraç, T. and Ozcelik, F. (2012) A Genetic Algorithm with Proper Parameters for Manufacturing Cell Formation Problems. *Journal of Intelligent Manufacturing*, **23**, 1047-1061. http://dx.doi.org/10.1007/s10845-010-0446-8

[10] Wu, T.-H., Chang, C.-C. and Yeh, J.-Y. (2009) A Hybrid Heuristic Algorithm Adopting both Boltzmann Function and Mutation Operator for Manufacturing Cell Formation Problems. *International Journal of Production Economics*, **120**, 669-688. http://dx.doi.org/10.1016/j.ijpe.2009.04.015

[11] Lin, S.W., Ying, K.C. and Lee, Z.J. (2010) Part-Machine Cell Formation in Group Technology Using a Simulated Annealing-Based Meta-Heuristic. *International Journal of Production Research*, **48**, 3579-3591. http://dx.doi.org/10.1080/00207540902896212

[12] Paydar, M.M., Mahdavi, I., Sharafuddin, I. and Solimanpur, M. (2010) Applying Simulated Annealing for Designing Cellular Manufacturing Systems Using MDmTSP. *Computers & Industrial Engineering*, **59**, 929-936. http://dx.doi.org/10.1016/j.cie.2010.09.003

[13] Kia, R., Baboli, A., Javadian, N., Tavakkoli-Moghaddam, R., Kazemi, M. and Khorrami, J. (2012) Solving a Group Layout Design Model of a Dynamic Cellular Manufacturing System with Alternative Process Routings, Lot Splitting and Flexible Reconfiguration by Simulated Annealing. *Computers & Operations Research*, **39**, 2642-2658. http://dx.doi.org/10.1016/j.cor.2012.01.012

[14] Rezaeian, J., Javadian, N., Tavakkoli-Moghaddam, R. and Jolai, F. (2011) A Hybrid Approach Based on the Genetic Algorithm and Neural Network to Design an Incremental Cellular Manufacturing System. *Applied Soft Computing*, **11**, 4195-4202. http://dx.doi.org/10.1016/j.asoc.2011.03.013

[15] Ghezavati, V.R. and Saidi-Mehrabad, M. (2011) An Efficient Hybrid Self-Learning Method for Stochastic Cellular Manufacturing Problem: A Queuing-Based Analysis. *Expert Systems with Applications*, **38**, 1326-1335. http://dx.doi.org/10.1016/j.eswa.2010.07.012

[16] Elbenani, B. and Ferland, J.A. (2012) An Exact Method for Solving the Manufacturing Cell Formation Problem. *International Journal of Production Research*, **50**, 4038-4045. http://dx.doi.org/10.1080/00207543.2011.588622

[17] Rafiei, H. and Ghodsi, R. (2013) A Bi-Objective Mathematical Model toward Dynamic Cell Formation Considering Labor Utilization. *Applied Mathematical Modelling*, **37**, 2308-2316. http://dx.doi.org/10.1016/j.apm.2012.05.015

[18] Paydar, M.M. and Saidi-Mehrabad, M. (2013) A Hybrid Genetic-Variable Neighborhood Search for the Cell Formation Problem Based on Group Efficiency. *Computers & Operations Research*, **40**, 980-990. http://dx.doi.org/10.1016/j.cor.2012.10.016

[19] Dalfard, V.M. (2013) New Mathematical Model for Problem of Dynamic Cell Formation Based on Number and Average Length of Intra and Intercellular Movements. *Applied Mathematical Modelling*, **37**, 1884-1896. http://dx.doi.org/10.1016/j.apm.2012.04.034

[20] Srinivasan, G. and Narendran, T.T. (1991) GRAFICS—A Nonhierarchical Clustering-Algorithm for Group Technology. *International Journal of Production Research*, **29**, 463-478. http://dx.doi.org/10.1080/00207549108930083

[21] Srinivasan, G. (1994) A Clustering Algorithm for Machine Cell Formation in Group Technology Using Minimum Spanning Trees. *International Journal of Production Research*, **32**, 2149-2158. http://dx.doi.org/10.1080/00207549408957064

[22] Miltenburg, J. and Zhang, W. (1991) A Comparative Evaluation of Nine Well-Known Algorithms for Solving the Cell Formation Problem in Group Technology. *Journal of Operations Management*, **10**, 44-72. http://dx.doi.org/10.1016/0272-6963(91)90035-V

[23] Kao, Y. and Li, Y.L. (2008) Ant Colony Recognition Systems for Part Clustering Problems. *International Journal of Production Research*, **46**, 4237-4258. http://dx.doi.org/10.1080/00207540601078054

[24] Nambirajan, T. and Panneerselvam, R. (1999) Machine-Component Cell Design Using Simulated Annealing. *International Journal of Management and Systems*, **15**, 185-208.

[25] Holland, J.H. (1975) Adaptation in Natural and Artificial Systems. The University of Michigan Press, Ann Arbor.

[26] Goldberg, D.E. (1989) Genetic Algorithms in Search Optimization & Machine Learning. Addison Wesley, Boston.

[27] Venugopal, V. and Narendran, T.T. (1992) A Genetic Algorithm Approach to the Machine-Component Grouping Problem with Multiple Objectives. *Computers & Industrial Engineering*, **22**, 469-480. http://dx.doi.org/10.1016/0360-8352(92)90022-C

[28] Balakrishnan, J. and Jog, P.D. (1995) Manufacturing Cell Formation Using Similarity Coefficients and a Parallel Genetic TSP Algorithm: Formulation and Comparison. *Mathematical and Computer Modelling*, **21**, 61-73. http://dx.doi.org/10.1016/0895-7177(95)00092-G

[29] Joines, A.J., Culbreth, C.T. and King, E.R. (1996) Manufacturing Cell Design: An Integer Programming Model Employing Genetic Algorithms. *IIE Transactions*, **28**, 69-85. http://dx.doi.org/10.1080/07408179608966253

[30] Su, C.T. and Hsu, C.M. (1998) Manufacturing Cell Formation Using Genetic Algorithm vs. Neural Networks. *Journal of the Chinese Institute of Industrial Engineers*, **15**, 127-139. http://dx.doi.org/10.1080/10170669.1998.10432953

[31] Mahdavi, I., Paydar, M.M., Solimanour, M. and Heidarzade, A. (2009) Genetic Algorithm Approach for Solving a Cell

Formation Problem in Cellular Manufacturing. *Expert Systems with Application*s, **36**, 6598-6604. http://dx.doi.org/10.1016/j.eswa.2008.07.054

[32] James, T.L., Brown, E.C. and Keeling, K.B. (2007) A Hybrid Grouping Algorithm for the Cell Formation Problem. *Computers & Operations Research*, **34**, 2059-2079. http://dx.doi.org/10.1016/j.cor.2005.08.010

[33] Keeling, K.B., Brown, E.C. and James, T.L. (2007) Grouping Efficiency Measures and Their Impact on Factory Measures for the Machine-Part Cell Formation Problem: A Simulation Study. *Engineering Applications of Artificial Intelligence*, **20**, 63-78. http://dx.doi.org/10.1016/j.engappai.2006.04.001

[34] Tunnukij, T. and Hicks, C. (2009) An Enhanced Grouping Genetic Algorithm for Solving the Cell Formation Problem. *International Journal of Production Research*, **47**, 1989-2007. http://dx.doi.org/10.1080/00207540701673457

[35] Tariq, A., Hussain, I. and Ghafoor, A. (2009) A Hybrid Genetic Algorithm for Machine-Part Grouping. *Computers & Industrial Engineering*, **56**, 347-356. http://dx.doi.org/10.1016/j.cie.2008.06.007

[36] Pailla, A., Trinbade, A.R., Parada, V. and Ochi, L.S. (2010) A Numerical Comparison between Simulated Annealing and Evolutionary Approaches to the Cell Formation Problem. *Expert Systems with Applications*, **37**, 5476-5483. http://dx.doi.org/10.1016/j.eswa.2010.02.064

[37] Vin, E. and Delchambre, A. (2014) Generalized Cell Formation: Iterative versus Simultaneous Resolution with Grouping Genetic Algorithm. *Journal of Intelligent Manufacturing*, **25**, 1113-1124. http://dx.doi.org/10.1007/s10845-013-0749-7

[38] Li, J., Wang, A. and Tang, C. (2014) Production Planning in Virtual Cell of Reconfiguration Manufacturing System Using Genetic Algorithm. *International Journal of Advanced Manufacturing Technology*, **74**, 47-64. http://dx.doi.org/10.1007/s00170-014-5987-0

[39] Deep, K. and Singh, P.K. (2015) Design of Robust Cellular Manufacturing System for Dynamic Part Population Considering Multiple Processing Routes Using Genetic Algorithm. *Journal of Manufacturing Systems*, **35**, 155-163. http://dx.doi.org/10.1016/j.jmsy.2014.09.008

[40] Chandrasekharan, M.P. and Rajagopalan, R. (1986) An Ideal Seed Non-Hierarchical Clustering Algorithm for Cellular Manufacturing. *International Journal of Production Research*, **24**, 451-464. http://dx.doi.org/10.1080/00207548608919741

[41] Chandrasekharan, M.P. and Rajagopalan, R. (1986) MODROC: An Extension of Rank Order Clustering for Group Technology. *International Journal of Production Research*, **24**, 1221-1233. http://dx.doi.org/10.1080/00207548608919798

[42] Kumar, C.S. and Chandrasekharan, M.P. (1990) Grouping Efficacy: A Quantitative Criterion for Goodness of Block Diagonal Forms of Binary Matrices in Group Technology. *International Journal of Production Research*, **28**, 233-243. http://dx.doi.org/10.1080/00207549008942706

[43] Chandrasekharan, M.P. and Rajagopalan, R. (1987) ZODIAC—An Algorithm for Concurrent Formation of Part Families and Machine Cells. *International Journal of Production Research*, **25**, 835-850. http://dx.doi.org/10.1080/00207548708919880

[44] Nambirajan, T. and Panneerselvam, R. (1995) Nonhierarchical Clustering of Machine-Component Cells Using Seeds from an Efficient Seed Generation Algorithm. *Industrial Engineering Journal*, **24**(11), 10-16 & **24**(12), 1-17.

[45] King, J.R. and Nakornchai, V. (1982) Machine-Component Group Formation in Group Technology: Review and Extension. *International Journal of Production Research*, **20**, 117-133. http://dx.doi.org/10.1080/00207548208947754

[46] Waghodekar, P.H. and Sahu, S. (1984) Machine-Component Cell Formation in Group Technology: MACE. *International Journal of Production Research*, **22**, 937-948. http://dx.doi.org/10.1080/00207548408942513

[47] Nambirajan, T. (1998) Machine-Component Cell Design Using Simulated Annealing with an Efficient Seed Generation Algorithm. Ph.D. Dissertation, Pondicherry University, Puducherry.

[48] Vohra, T., Chen, D.S., Chang, J.C. and Chen, H.C. (1990) A Network Approach to Cell Formation in Cellular Manufacturing. *International Journal of Production Research*, **28**, 2075-2084. http://dx.doi.org/10.1080/00207549008942854

[49] Choobineh, F. (1988) A Framework for the Design of Cellular Manufacturing Systems. *International Journal of Production Research*, **26**, 1161-1172. http://dx.doi.org/10.1080/00207548808947932

[50] Safer, S.M., Kern, G.M. and Wei, J.C. (1992) A Mathematical Programming Approach for Dealing with Exceptional Elements in Cellular Manufacturing. *International Journal of Production Research*, **30**, 1029-1036. http://dx.doi.org/10.1080/00207549208942940

[51] De Witte, J. (1980) The Use of Similarity Coefficient in Production Flow Analysis. *International Journal of Production Research*, **18**, 503-514. http://dx.doi.org/10.1080/00207548008919686

[52] Tam, K.Y. (1990) An Operation Sequence Based Similarity Coefficient for Part Families Formations. *Journal of Manufacturing Systems*, **9**, 55-68. http://dx.doi.org/10.1016/0278-6125(90)90069-T

[53] Askin, R.G. and Subramanian, S.P. (1987) A Cost-Based Heuristic for Group Technology Configuration. *International Journal of Production Research*, **25**, 101-113. http://dx.doi.org/10.1080/00207548708919825

[54] Stanfel, L.E. (1985) Machine Clustering for Economic Production. *Engineering Costs and Production Economics*, **9**, 73-81. http://dx.doi.org/10.1016/0167-188X(85)90012-6

[55] Balasubramanian, K.N. and Panneerselvam, R. (1993) Covering Technique-Based Algorithm for Machine Grouping to form Manufacturing Cells. *International Journal of Production Research*, **31**, 1479-1504. http://dx.doi.org/10.1080/00207549308956803

[56] Chan, H.M. and Milner, D.A. (1982) Direct Clustering Algorithm for Group Formation in Cellular Manufacturing. *Journal of Manufacturing Systems*, **1**, 65-75. http://dx.doi.org/10.1016/S0278-6125(82)80068-X

[57] Mosier, C. and Taube, L. (1985) Weighted Similarity Measure Heuristics for the Group Technology Machine Clustering Problem. *Omega*, **13**, 577-583. http://dx.doi.org/10.1016/0305-0483(85)90046-5

[58] Chandrasekharan, M.P. and Rajagopalan, R. (1989) Groupability: An Analysis of the Properties of Binary Data Matrices for Group Technology. *International Journal of Production Research*, **27**, 1035-1052. http://dx.doi.org/10.1080/00207548908942606

[59] Panneerselvam, R. (2012) Design and Analysis of Experiments. PHI Learning Pvt. Limited, New Delhi.