S.G.Ponnambalam, R.Sudhakar Pandian, S.S.Mahapatra, and S.Saravansankar, “Cell formation with workload data in cellular manufacturing system using genetic algorithm”, paper no. IEEM07-P-0080, IEEE International Conference on Industrial Engineering and Engineering Management, Hotel Furama River Front, 405 Havelock Road, Singapore 169633, 2-5 December 2007.

CELL FORMATION WITH WORKLOAD DATA IN CELLULAR MANUFACTURING SYSTEM USING GENETIC ALGORITHM

S.G. Ponnambalam[†], R.SudhakaraPandian[‡], S.S.Mohapatra[*], S.Saravanasankar[‡]
[†]School of Engineering, Monash University, 46150 Petaling Jaya, Malaysia
[‡],Kalasalingam University, Virudhunagar, India.
[*]National Institute of Technology, Rourkela. India
[†]sgponnmbalam@eng.monash.edu.my

Abstract - **Cellular Manufacturing System (CMS) is regarded as an efficient production strategy for batch type of production. CMS rests on the principle of grouping the machines into machine cells and parts into part families based on suitable similarity criteria. Usually zero-one machine-part incidence matrix (MPIM) obtained from the route sheet information is used to form machine cells. In this paper, an attempt has been made to solve the cell formation problem considering work load data and a genetic algorithm (GA) is suggested to form machine cells and part families. The performance of the proposed algorithm is compared with existing algorithms such as K-means algorithm and modified ART1 algorithm found in the literature using a newly defined performance measure known as modified grouping efficiency (MGE). The proposed algorithm is tested with problems from open literature and the results are compared with the existing algorithms found in the literature. The results support the better performance of the proposed algorithm.**

Keywords – **Cell Formation, Grouping efficiency, Genetic algorithm.**

I.INTRODUCTION

Cellular manufacturing is a manufacturing philosophy where the similar parts are grouped together based on design and/or manufacturing attributes. The basic problem in cellular manufacturing is to group the machines into machine cells and the parts into part families. In order to develop a generalized and a more realistic model for cell formation problem besides considering the route cards of the parts, it is essential to consider also the other useful production data like lot size of the products, machine capacity, operation time and sequence. In this paper, an attempt has been made to solve the problem with operation time of the parts**.** Meta-heuristics like Genetic Algorithm, Simulated Annealing, Tabu search and Ants Colony Systems seem to be prominent algorithms for solving cell formation problems using MPIM. Therefore, it is highly desirable to test the performance of meta-heuristics using real valued matrices. To this end, an algorithm based on most commonly used meta-heuristic known as Genetic Algorithm is proposed. The model has been tested using wide variety of problems from literature and found to be consistent in producing good results. The basic purpose of this study is to develop a simple and efficient methodology to produce quick solutions for shop floor managers with least computational efforts.

II. LITERATURE REVIEW

Burbidge [1] viewed group technology GT as a change from an organization of people mainly on process, to an organization based on completed products, components and major completed tasks. From 1960 onwards there are many approaches presented in the literature. Initially the methods like similarity coefficient methods (SCM) [2], rank order clustering (ROC) [3] and graph theory [4] methods were developed only to group the similar machines into machine cells and the grouping of parts into part families was done only in the supplementary step of the procedure. Later clustering methods such as the MODROC [5], ZODIAC [6], MACE [7] are reported for solving the cell formation problems. Subsequently many algorithms which are based on metaheuristics like simulated annealing (SA) algorithm, genetic algorithm (GA), tabu search (TS) were also developed to solve the cell formation problems [8-10].
Invariably most of the above said procedures are suitable to work with only the binary form of MPIM. In this work an attempt is made to propose a simple and efficient genetic algorithm for cell formation problem with real-valued non-binary workload data

III. GENETIC ALGORITHM – AN OVERVIEW

Genetic Algorithms (GAs) are adaptive search and optimization algorithms, based on the mechanics of natural selection and natural genetics [11]. They mimic the survival-of-the-fittest principle of nature to make the search process. GA begins with a population of strings (individuals) created randomly, representing the design/decision variables. Thereafter, each string in the population is evaluated to find its fitness value (i.e. the objective function value of the given optimization problem). The population is then operated by three main operators (known as genetic operators) - *Reproduction, Crossover* and *Mutation* - to create a new and better population. The new population is further evaluated for the fitness values and tested for termination. If the termination criteria is not met, the population is iteratively

1-4244-1529-2/07/$25.00 ©2007 IEEE

operated by the above three genetic operators and evaluated.

Genetic Operators

Reproduction is the first operator applied on a population. In reproduction, individual strings from the population are selected, (this is why, this operator is also known as *Selection* operator) and copied according to their fitness values to form an intermediate mating pool containing more of *highly fit* individuals and less of *less fit* individuals.

IV. THE PROPOSED GA BASED ALGORITHM

The flow chart shown in the Fig. 1.depicts the GA based algorithm whose design layout is as under:

A. GA coding scheme

Representation forms a key role in the development of GA. . In the proposed pheno style coding method, each solution is coded as a set of numbers. The total number of numbers will be equal to the total number of machines to be grouped. The position of a number will represent the machine number and the value of the number will represent the machine cell to which the particular machine will belong to.

Example: For a five machines two cells solution

Coded solution 2 1 2 1 2

Decoded solution

1 2 3 4 5 ⟶ machine number

↓ ↓ ↓ ↓ ↓

2 1 2 1 2 ⟶ cell number

.

In the example solution, the machines 1, 3 and 5 belong to cell number two, the machines 2 and 4 belong to cell number one.

After representation, the population of chromosomes in the range of 10 to 40 depending on the size of problem is chosen. The initial solutions are generated randomly and the generated solution is subjected to iterations or generations.

B. Objective Function

The objective of this work is to minimize the total cell load variation. [8]. The visit of the components to the machines has been denoted in terms of their work load as shown in Table II for the computation of cell load variation.

Minimize

$$Z = \sum_{i=1}^{m} \sum_{k=1}^{c} x_{ik} \sum_{j=1}^{p} (w_{ij} - m_{kj})^2 \quad (1)$$

$$m_{kj} = \frac{\sum_{i=1}^{m} X_{ik} \cdot w_{ij}}{\sum_{i=1}^{m} X_{ik}} \quad (2)$$

where $\sum_{i=1}^{m} X_{ik} \cdot w_{ij}$ is the total cell load of cell k that induced by part j and $\sum_{i=1}^{m} X_{ik}$ is the total number of machines in cell k. For a predefined number of cell k, the Z value is calculated using (1).

m	-	number of machines. (i = 1,2,3…,m)
p	-	number of parts. (j = 1,2,3….,p)
c	-	number of cells. (k = 1,2,3…,c)
$[x_{ik}]$	-	machine cell (m x k) membership matrix where x_{ik} = 1 if i^{th} machine is in cell k, = 0 otherwise.
$[w_{ij}]$	-	machine part (m x p) matrix in terms of workload on machine i induced by part j.
$[m_{kj}]$	-	cell part (k x p) matrix of average cell load as in (2).

C. Reproduction

A fitness function value is computed for each string in the population and the objective is to find a string with the maximum fitness function value. Since objective is minimization, it is required to map it inversely and then maximize the resultant. Goldberg [11] suggested a mapping function given as, F(t) - fitness function of t^{th} string [F (t) = Z_{max} – Z (t)]

Z_{max} - max [Z (t)] of all strings (t).

The advantage is that the worst strings get a fitness function value of zero and there is no chance of the worst strings getting reproduced into the next generation.

D. Crossover

With the phenotype coding scheme used in the proposed algorithm, single point crossover operation is performed. In the crossover operation a pair of strings are selected at random with a crossover probability. For a selected pair of strings a crossover site is selected at random. The genes (numbers) after the crossover site are swapped to produce the pair of offspring strings.

The strings before crossover

$Parent_1$ 1 2 | 2 1 2
$Parent_2$ 1 2 | 1 1 2
S

The strings after crossover

Off $spring_1$ 1 2 | 1 1 2
Off $spring_2$ 1 2 | 2 1 2
S

E. Mutation

Bitwise mutation is performed with the intermediate population with a mutation probability. Two sites are selected at random and the corresponding numbers are exchanged to get the new solution.

The strings before mutation
1 **2** 1 **1** 2

The strings after mutation
1 **1** 1 **2** 2

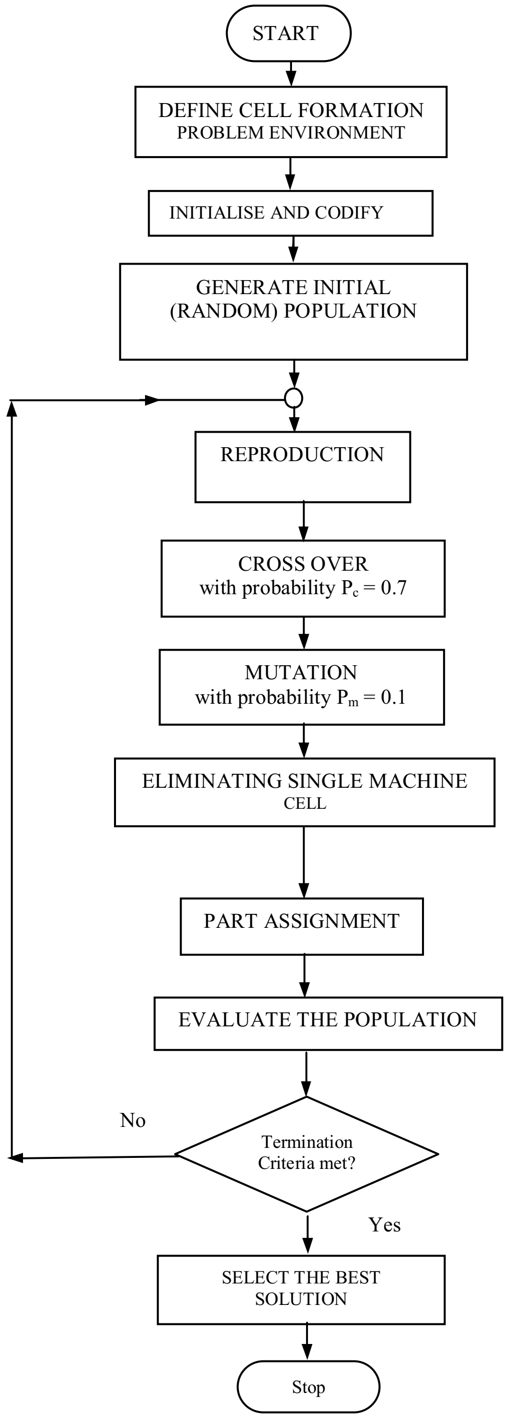

Fig.1 Flow chart of the proposed algorithm

F. Eliminating machine cell with single machine

If single machine found in any cell, the following operations are carried out to merge the single machine cells with other cells. The average workload of each part in the cell and the Euclidean distance between the cells are calculated. The minimum Euclidean distance between cells is found out. Cells with single machine are merged to the cells which has minimum Euclidean distance.

G. Part Assignment

The following procedure given by Zolfaghari and Liang [12] is used to assign parts into the machine cells. A machine cell which processes the part for a larger number of operations than any other machine cell is found out and the corresponding part is assigned into that cell. Ties are broken by choosing the machine cell which has the largest percentage of machines visited by the part. In the case of tie again the machine cell with the smallest identification number is selected. Thus all the parts are assigned to all the cells which form part families using membership index given in (3).

$$P_{kj} = \frac{f_{kj}}{f_k} \cdot \frac{f_{kj}}{f_j} \cdot \frac{T_{kj}}{T_j} \qquad (3)$$

P_{kj} -Membership index of part j belongs to cell k.
f_{kj} - Number of machines in cell k required by part j.
f_k - Total number of machines in cell k.
f_j - Total number of machines required by part j.
T_{kj} - Processing time of part j in cell k.
T_j - Total processing time required by part j.

V. MEASURE OF PERFORMANCE

To evaluate the grouping in the cell formation problem the performance measures usually considered in the literature are grouping efficiency and grouping efficacy [13].

As this measures are found not suitable for the problem considered in this work which is of real valued data a new performance measure known as Modified Grouping Efficiency (MGE) formulated to assess the grouping efficiency of the proposed algorithm and to compare the efficiency with other algorithms found in the literature. The proposed MGE is calculated using (4).

$$MGE = \frac{T_{pti}}{T_{pto} + \sum_{k=1}^{c} T_{ptk} + \sum_{k=1}^{c} T_{ptk} \cdot \frac{N_{vk}}{N_{ek}}} \qquad (4)$$

T_{pti} - Total processing time inside the cells.
T_{pto} - Total processing time outside the cells.
T_{ptk} - Total processing time of cell k.
N_{vk} - Number of voids in cell k.
N_{ek} - Total number of elements in cell k.

TABLE I
Machine – Part Incidence Matrix of Size 6x8

	1	2	3	4	5	6	7	8
1	0	1	0	1	0	0	1	0
2	1	1	1	0	1	1	1	1
3	0	0	1	0	0	1	0	1
4	0	0	0	1	0	0	1	0
5	1	0	1	0	1	1	0	1
6	0	0	0	1	0	0	1	0

TABLE II
Real valued workload matrix

	1	2	3	4	5	6	7	8
1	0	0.53	0	0.99	0	0	0.83	0
2	0.91	0.82	0.83	0	0.91	0.92	0.86	0.97
3	0	0	0.79	0	0	0.56	0	0.88
4	0	0	0	0.53	0	0	0.51	0
5	0.98	0	0.83	0	0.71	0.58	0	0.54
6	0	0	0	0.54	0	0	0.74	0

TABLE III
Output matrix

	4	7	1	2	3	5	6	8
1	0.989	0.830	0.000	0.526	0.000	0.000	0.000	0.000
4	0.528	0.514	0.000	0.000	0.000	0.000	0.000	0.000
6	0.540	0.744	0.000	0.000	0.000	0.000	0.000	0.000
2	0.000	0.859	0.913	0.823	0.832	0.908	0.916	0.974
3	0.000	0.000	0.000	0.000	0.787	0.000	0.561	0.884
5	0.000	0.000	0.975	0.000	0.830	0.708	0.583	0.540

VI. RESULTS AND DISCUSSION

In this study, an efficient algorithm based on genetic algorithm has been proposed for cell formation problem considering operational time of the parts instead of conventional zero-one incidence matrix with the objective of minimizing total cell load variation. The algorithm is coded in C^{++} and tested on the Pentium IV machine. The problem of size 6×8, as shown in example Table I is considered for illustration. The real valued workload matrix is presented to the algorithm. The algorithm generates a solution after 250 generations with population size of 10. The output is in the block diagonal form of 2 cells as shown in Table III.

The appropriate values of the GA parameters are arrived at, based on the satisfactory performance of trials conducted for this application with different ranges of values. The crossover probability was varied from 0.4 to 0.9 and it was found that the solution was improving faster for a cross over probability of 0.70. Similarly in the range from 0.01 to 0.20, the mutation probability of 0.1 was found to retain better solutions than worse solutions. Similarly the Population size for the different problem sizes were arrived by conducting trials over a range and the one which gives satisfactory results was fixed for the concerned problem size. The population sizes for the whole set of example problems varied between 10 and 40.

Number of problems with varied sizes form literature, as given in Table IV, is considered for testing the proposed algorithm. The real valued matrix is produced by assigning random numbers in the range of 0.5 to 5 as uniformly distributed values by replacing the ones in the incidence matrix and zeros to remain in its same positions. The crossover probability and mutation probability have been fixed to 0.7 and 0.1 respectively. To tune the algorithm these values can be varied depending on size of problem. The number of generations is varied from problem to problem in the range of 250 to 900. Similarly, the population size is varied in the range of 10 to 40 depending on the size of problem. A new method, Modified Grouping Efficiency, has been proposed to measure the performance of the grouping with real values. The results are compared with the results obtained from K-means clustering algorithm [14] and the modified ART1 algorithm [15] as presented in Table 1V. The size of machine-part incidence matrices considered in this paper ranges from 5×7 to 30×50. The number of cells is varied from 2 to 6. The exceptional elements are comparatively reduced in the proposed algorithm.

VII. CONCLUSION

In this work a GA based algorithm to solve the cell formation problem using the non binary real valued work load data as an input matrix. The proposed algorithm is tested with benchmark problems found in the literature and the results are compared with the existing algorithms mainly K-means clustering algorithm and the modified ART1 algorithm. In addition to the commonly used measure of performance that is the number of exceptional elements, a newly developed performance measure namely modified grouping efficiency is also applied to evaluate the efficiency of the proposed algorithm. The proposed GA based algorithm outperforms the existing methods both in terms of exceptional elements and modified grouping efficiency..

The GA based algorithm may be suitably modified and employ to solve the cell formation problem with other non binary real value data like machine capacity, production volume and product sequence.

REFERENCES

[1] Burbidge, J. L. (1977). A Manual Method of Production Flow Analysis, *The Production Engineer*, 56, 34-38.

[2] Seifoddini, H and Wolfe, P.M., (1986), Application of similarity coefficient method in GT. IIE Transactions, 21, 382-388.

[3] King, (1980)Machine Component grouping in production flow analysis: An approach using a rank order clustering algorithm, International Journal of Production Research 18 (2) 213-232.

[4] Rajagopalan, and Batra. (1975). Design of cellular manufacturing systems: a graph theoretic approach, International Journal of Production Research, 13 (6), 567-579.

[5] M.P.Chandrasekaran, R.Rajagopalan, (1986) MODROC: an extension of rank order clustering for group technology, International Journal of Production Research 24 (5) 1221-1233.

[6] M.P.Chandrasekaran,R.Rajagopalan, (1987) ZODIAC: An algorithm for concurrent formation of part families and machine cells, International Journalof Production Research 25 (6) 835-850.

[7] Waghodekar, P.H. and Sahu, S. (1984). Machine-component cell formation in group technology: MACE. International Journal of Production Research 22, 937–948.

[8] V.Venugopal, T.T.Narendran, (1992) A genetic algorithm approach to the machine component grouping problem with multiple objectives, Computers and Industrial Engineering22 469-480.

[9] Jose Fernando, Maurico, (2002) A Hybrid Genetic Algorithm in cell formation problems, AT&T Labs Research Technical Report TD-5FE6RN, AT&T Labs Research, Florham Park, NJ 07932 USA.

[10] G.A.Carpenter, S.Grossberg, (2002).Adaptive ResonanceTheory, A Handbook of Brain Theory and Neural Networks, Second Edition MIT Press,Massachusetts, USA.

[11] Goldberg, D.E. (1989). Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley, Reading, MA.

[12] Zolfaghari, M.Liang. (2003)A new genetic algorithm for the machine/part grouping problem involving processing times and lot sizes, Computers and Industrial engineering 45, 713-731.

[13] K.R.Kumar and Chandrasekaran.M.P, (1990) Grouping efficacy: A quantitative criterion forgoodness of block diagonal forms of binary matrices in group technology., International Journal of Production Research, 28, 2233-243.

[14] Yiu-Ming Cheung, (2003) K-means: A new generalized K- means clustering algorithm. Pattern Recognition Letters, 24, 2883-2893.

[15] S.S.Mahapatra,R.Sudhakarapandian,(2006)Adaptive Resonance Theory Applied To Generalized Machine Cell Formation, *International Journal for Manufacturing Science & Technology*, Vol.8, no.1, p. 13-21.

TABLE IV
PERFORMANCE OF THE PROPOSED ALGORITHM

S.No	Problem from Literature [9]	Problem size	No. of cells	K-means		modified ART1		Proposed Algorithm	
				EE	MGE	EE	MGE	EE	MGE
1	King and Nakornchai (1982)	5 x 7	2	2	77.25	2	77.25	2	77.25
2	Waghodekar and sahu (1984)	5 x 7	2	2	78.34	2	78.34	2	78.34
3	Seiffodini (1989)	5 x 18	2	7	81.87	7	81.87	7	81.87
4	Kusiak (1992)	6 x 8	2	2	79.85	2	79.85	2	79.85
5	Kusiak (1987)	7 x 11	2	3	61.77	3	61.77	3	61.77
6	Boctor (1991)	7 x 11	2	1	65.48	1	65.48	1	65.48
7	Seiffodini and wolfe (1986)	8 x 12	2	6	57.00	4	69.70	6	69.70
8	Chandrasekaran et al. (1986)a	8 x 20	2	28	60.00	25	61.30	28	61.30
9	Chandrasekaran et al. (1986)b	8 x 20	3	9	83.40	9	83.40	9	83.40
10	Mosier et al. (1985)	10 x 10	3	0	77.14	0	77.14	0	77.14
11	Chan et al. (1982)	10 x 15	3	0	93.28	0	93.28	0	93.28
12	Askin et al. (1987)	14 x 23	2	2	59.43	2	60.59	0	62.42
13	Stanfel (1985)	14 x 24	4	7	68.13	7	68.13	3	73.19
14	Mccornick et al. (1972)	24 x 16	4	34	46.70	30	51.39	29	52.02
15	Boctor (1991)-2	16 x 30	3	6	68.55	6	68.55	6	68.55
16	Boctor (1991)-3	16 x 30	3	4	67.89	4	67.89	4	67.89
17	Boctor (1991)-5	16 x 30	3	8	70.05	9	68.99	8	70.05
18	Boctor (1991)-6	16 x 30	3	4	69.73	4	69.73	3	70.91
19	Boctor (1991)-7	16 x 30	3	5	71.50	5	71.50	5	71.50
20	Boctor (1991)-8	16 x 30	3	11	65.18	13	59.71	11	65.18
21	Boctor (1991)-9	16 x 30	3	8	71.78	9	69.49	8	71.78
22	Srinivasan et al. (1990)	16 x 30	3	15	64.81	15	64.81	20	64.81
23	Mosier et al. (1985)	20 x 20	2	42	49.13	22	51.10	29	51.10
24	Carrie (1973)	20 x 35	3	1	71.00	1	71.15	1	71.15
25	Boe et al. (1991)	20 x 35	4	31	61.50	28	61.70	32	61.70
26	Kumar et al. (1986)	23 x 20	3	38	51.70	39	48.14	42	51.92
27	Chandrasekaran et al. (1989)a	24 x 40	6	0	90.28	0	90.28	0	94.58
28	Chandrasekaran et al. (1989)b	24 x 40	5	7	71.60	9	73.89	9	73.89
29	Kumar et al. (1987)	30 x 41	3	12	56.65	17	53.98	15	56.14
30	Stanfel (1985)a	30 x 50	6	20	61.84	26	55.51	22	62.23
31	Stanfel (1985)b	30 x 50	3	33	50.51	17	53.19	25	55.35