4th IEEE Conference on Automation Science and Engineering
Key Bridge Marriott, Washington DC, USA
August 23-26, 2008

Modified ART1 Neural networks for Cell Formation using Production Data

S.G. Ponnambalam†, R.SudhakaraPandian‡, S.S.Mahapatra*, S.Saravanasankar‡

Abstract - **In the present work, an attempt has been made to form disjoint machine cells using modified ART1 (Adaptive Resonance Theory) to handle the real valued workload matrix. The methodology first allocates the machines to various machine cells and then parts are assigned to those cells with the aid of degree of belongingness through a membership index. The proposed algorithm uses a supplementary procedure to effectively take care of the problem of generating cells with single machine that may be encountered at times. A modified grouping efficiency (MGE) is proposed to measure the performance of the clustering algorithm. The results of modified ART1 algorithm are compared with the results obtained from K-means clustering and genetic algorithm. The modified ART1 results are also compared with the literature results in terms of number of exceptional elements. The performance of the proposed algorithm is tested with genetic algorithm and K-means clustering algorithm. The results distinctly indicate that the proposed algorithm is quite flexible, fast and efficient in computation for cell formation problems and can be applied in industries with convenience.**

Keywords: Cell formation; Adaptive Resonance Theory Networks; K-means clustering; Grouping efficiency.

I. INTRODUCTION

Cellular Manufacturing aims at formation of machine cells for achieving the benefits of mass production to batch production with higher values of variety, product-mix and total quantity. A cell is a group of closely located workstations where multiple and sequential operations are performed on one or more part families. The basic idea in cellular manufacturing is to group the machines into machine cells and the parts into part families with an aim to achieve reduced material handling, reduced manufacturing lead time, reduced work-in-process, reduced setup time, increased flexibility and maximum utilization of resources. Major approaches in cell formation problems can be categorized as visual inspection, parts classification and coding and production flow analysis.

However, the production flow analysis is quite popular method in industries as the existing production data can be utilized to form machine cells. The past research work reveals that the cell formation problems are addressed with zero-one incidence matrices in most cases. These approaches can hardly incorporate the real life production factors.

The production data such as lot size of the products, machine capacity, operational time and operation sequence need to be considered in order to generalize cell formation problem. In the present work, an attempt is made to address the generalized cell formation problem with operational time of the parts.

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. The most significant contributions by the researchers are towards similarity coefficient methods, graph theory, mathematical programming, meta-heuristics, fuzzy set theory and neural networks which are used to solve cell formation problems. 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] and GRAFICS [8] 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. The popular algorithms of this category include SA based algorithm proposed by Boctor [9], GA based algorithm proposed by Venugopal and Narendran [10], Jayakrishnan Nair and Narendran [11] proposed an algorithm called CASE which considers sequence of operations that a part undergoes through a number of machines. Fernando [12], TS based algorithm proposed by Wu et al. [13]. Meanwhile many researchers have proposed artificial neural network (ANN) based methodologies for solving the cell formation problems. There are many popular ANN models found in the literature [14-16] which are efficient in producing satisfactory solutions to this NP-hard problems.

Kao et al. [14] introduced back propagation neural network model for GT whereas Kaparthi and Suresh [16] made an attempt to introduce adaptive resonance theory (ART1). There are certain disadvantages of ART1 network – (i) it will recognize only the binary input data and (ii) the resulting solution is highly influenced by the order of presentation of input vectors. Chen and Cheng [17] have successfully overcome the second disadvantage using some supplementary procedures.

† S.G.Ponnambalam is with School of Engineering, Monash University, 46150 Petaling Jaya, Malaysia (Phone: +60-3-56360600, e-mail: sgponnmbalam@eng.monash.edu.my).
‡ R.SudhakaraPandian (sudhakarme@rediffmail.com) and S.Saravanasankar (ssaravanasankar@yahoo.co.in) are with Kalasalingam University, India
* S.S.Mahapatra is with National Institute of Technology, Rourkela. India

978-1-4244-2023-0/08/$25.00 ©2008 IEEE.

II THE MODIFIED ART1 ALGORITHM TO CELL FORMATION CONSIDERING PRODUCTION DATA

The proposed algorithm is a modified version of ART1, adapted from the method proposed by Yoh-Han Pao [18] that accommodates analogue patterns (real step valued matrix) instead of binary form of input vectors. Let M be the total number of machines and N be the total number of parts. Then workload matrix size becomes M x N. The elements of the matrix represent operation time T_{ij} which indicates that part j takes T_{ij} units of time to complete its operation in machine i.

NOTATIONS USED:

M number of machines.
N number of parts.
C total number of cells.
i machine index
j part index.
k cell index.
ρ vigilance threshold.
wt_{ji} top-down weights.
e_i Euclidean distance.
X_q input vector (rows) represented as (x_{q1} x_{q2} x_{q3}……..x_{qN}) for row q.
n number of machines in active node k.
m n+1.
P_{kj} membership value of part j belonging to cell k.
P_m maximum of P_{kj}.
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.
w_v weighting factor to the voids.
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.

The Procedure

Step 1: Initialize: Set nodes in the input layer equal to N (number of parts) and nodes in output layer equal to M (number of machines). Set vigilance threshold (ρ).
Step 2: Initialize top-down connection weights. Top-down weights wt_{ji} (0) = 0 for i = 1, 2,... M. and j = 1,2,... N.
Step 3: Let q =1. The first input vector X_1 (first row of the workload matrix) is presented to input layer and assigned to the first cluster. Then, first node in the output layer is activated.
Step 4: The top-down connection weights for the present active node are set equal to the input vector.
Step 5: Let q = q+1. Apply new input vector X_q. (input vectors are the rows of the workload matrix).
Step 6: Compute Euclidean distance between X_q and the exemplar stored in the top-down weights (wt_{ji}) for all active nodes i as given in the Eq. (1). This distance function is used to calculate similarity between the stored pattern and the present input pattern. If the similarity value is less than or equal to ρ (vigilance threshold), the present input is categorized under the same cluster as that of stored pattern.

$$e_i = \sqrt{\sum_{j=1}^{N} (x_{qj} - wt_{ji})^2} \qquad \textbf{(1)}$$

Step 7: Perform vigilance test: Find out minimum Euclidean distance.
Step 8: If min $e_i \leq \rho$ (threshold value), select output node for which Euclidean distance is minimum. If tie occurs, select the output node with lowest index number. Suppose output node k is selected. Then allocate the vector X_q to the node k (cell) and activate node k. Make increment to the number of machines in the active node k by one. If e_i's for all active nodes are greater than ρ, then go to step 9.
Step 9: Start a new cell by activating a new output node.
Step 10: Update top-down weights of active node k using Eq. (2).
When a vector is selected (to be allocated to an output node), its top-down weights are updated using more information of the previously stored exemplar and a relatively less information of the input vector (pattern).

$$wt_{jk} = \left(\frac{n}{m} \times wt_{jk}\right) + \left(\frac{1}{m} \times x_{qj}\right) \qquad \textbf{(2)}$$

Step 11: Go to step 5 and repeat till all the rows are assigned in the output nodes (cells).
Step 13: Assign parts to cells using the membership index given in Eq. (3).

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

The membership index P_{kj} represents the belongingness of the part j to the cell k. There are three components in the membership index as shown in Eq. (3). First component (f_{kj}/f_k) denotes the proportion of machines of cell k required by part j. The second component (f_{kj}/f_j) is a ratio between the number of machines in cell k required by part j and the total number of machines required by part j. The third component (T_{kj}/T_j) is the proportion of processing time of part type j that can be accommodated in cell k. The belongingness of the part j, $P_{kj,}$ is calculated for all the cells k = 1,2,3…C. Part j is assigned to a cell based on its maximum belongingness to the cells. The maximum belongingness can be calculated using Eq. (4). The value of P_m lies between 0 and 1 where P_m = 1 indicates that part j is perfectly eligible to belong the cell k.

$$P_m = \max \{P_{kj}\} \quad k=1, 2, 3 \ldots C. \qquad \textbf{(4)}$$

III. MEASURE OF PERFORMANCE

Grouping efficiency proposed by Chandrasekharan and Rajagopalan [6] and grouping efficacy was proposed by Kumar and Chandrasekaran [19]. These measures are suitable only for the zero-one incidence matrix. These measures cannot be adopted for generalized cell formation problem where information regarding operational times is of importance. In this work, a new measure for grouping efficiency termed as modified grouping efficiency (MGE) has been proposed to find out the performance of real valued cell formation problem with due emphasis on number of voids.

Modified Grouping Efficiency:

Weighting factor to the voids and modified grouping efficiency (MGE) are calculated using the following equations 5 and 6.

$$w_v = N_{vk} / N_{ek} \quad (5)$$

$$MGE = \frac{T_{pti}}{T_{pto} + \sum_{k=1}^{c} T_{ptk} + \sum_{k=1}^{c} T_{ptk} \cdot w_v} \quad (6)$$

Unlike grouping efficiency, modified grouping efficiency does not treat all the operations equally. Moreover a weighting factor for voids is considered to reflect the packing density of the cells. Modified grouping efficiency produces 100% efficiency when the cells are perfectly packed without any voids and exceptional elements.

IV. ILLUSTRATION OF THE PROPOSED ALGORITHM

The binary matrix is converted into real valued workload matrix by replacing the ones using uniform random numbers in the range of 0.5 to 1 and zeros remain unchanged in the same position. The resultant matrix is presented as input to the proposed algorithm. Initially the algorithm assigns the machines (rows) to the cells. After rows are sorted out, parts (columns) are assigned to the cells using the membership index given in the Eq. (3) and Eq. (4) to form the part families. For problem number 4 from Tables 1a and 1b, the first cell does not have any voids and hence weighting factor to the voids (w_v) is zero. The number of voids in second cell (N_{vk}) is four and the total number of elements (N_{ek}) is eighteen. Therefore, weighting factor for voids (w_v) for the second cell equals to 0.2222. The total processing time in second cell (T_{ptk}) is 11.234 and it is multiplied by its weighting factor for voids to produce T_{ptk} x w_v equal to 2.496. The total processing time inside both the cells (T_{pti}) is 15.379. As total number of exceptional elements is two, sum of their value (T_{pto}) is 1.385. The summation of (T_{pto}), (T_{pti}) and (T_{ptk} x w_v) is calculated as 19.2604. Now, the value of MGE is calculated by the ratio of 15.379 to 19.2604 and it is obtained using the proposed algorithm as 79.85%.

V. ELIMINATING CELLS WITH SINGLE MACHINE

A cell with a single machine is not desirable in cellular manufacturing because its expected advantages, in particular flexibility, will be lost. Therefore the following procedure can be adopted to deal with if single machine cells are found in the output.

- Calculate the average workload of each part in the cell.
- Calculate the Euclidean distance between the cells.
- Find out the minimum distance between cells.

Merge the cell (assigned with single machine) with the cell that has minimum Euclidean distance.

VI. RESULTS AND DISCUSSIONS

In this study, an efficient algorithm is proposed based on ART1 for generalized cell formation problem. The algorithm is coded in C^{++} and run on an IBM Pentium IV PC with 2.4 GHz Processor. Number of problems with varied sizes from open literature [12] are considered for testing the proposed algorithm. The workload (input) matrices are generated by replacing the ones in the incidence matrix with uniformly distributed random numbers in the range of 0.5 to 1 and zeros to remain in same positions. The problem sizes considered in this work ranges from 5 x 7 to 30 x 50. It is assumed that the lot size for all the products equal to one to characterize the behaviour of the sample problems considered in this study although it is not restrictive to one. In order to evaluate the performance of the proposed algorithm, the sample problems are tested with two more algorithms such as genetic algorithm (GA) and K-means clustering algorithm and the results are reported in Table 3

In this paper, the fitness function chosen is maximization of MGE. The population size is varied from one problem to another depending on size of the problem (population size ranges from 10 to 40). For small sized problem like 5 x 7, the population size is set at 10 whereas it is 40 for the problem size of 30 x 50. The chromosomes are selected using a well-known roulette wheel selection operator. The number of iterations (generations) is varied depending on size of the problem till solution converges. For problem size of 5 x 7, the number of generations is 100 and it is as high as 1100 generations for the problem size of 30 x 50. Position based crossover and mutation are adopted from Syswerda [21]. The probability of crossover and mutation is fixed to be 0.5 and 0.05 respectively for all the problems.

In K-means algorithm, the number of desired clusters is chosen first. The algorithm assigns machines to different clusters using Euclidean distance. The number of iterations is varied until no further improvement is possible in the solution. The number of clusters and number of iterations are varied depending on the size of the problem. It is observed that the number of iterations lies in the ranges of 20-35 for the sample problems considered in this work. The standard software SYSTAT is used to form clusters using K-means algorithm.

The computational time for few sample problems between modified ART1, GA and K-means algorithm are reported in Table 4. It certainly proves that the proposed ART1 is computationally faster than GA. The results obtained using ART1 are compared with the results produced by K-means clustering algorithm and GA and listed in Table 3. The modified ART1 results are also compared with the literature results in terms of number of exceptional elements reported by Goncalves and Resende [12] and by Venkumar and Haq [22]. It has been observed that number of exceptional elements reduces with proposed algorithm for most of the sample problems in comparison to solutions obtained by other methods. As far as MGE is concerned, solutions obtained by the proposed algorithm mostly outperforms other two methods in most of the tested problems. K-mean clustering seems to be inefficient in respect to performance measures like MGE and number of exceptional elements. However, all the methods are equally good when the problem size is small. Though GA is found to produce better results over the ART 1 for only medium size problems, the ART1 exhibits its superiority over GA and K-means for the required computational effort in terms of CPU time or number of iterations as depicted in Table 4 Modified ART 1 requires one iteration only whereas genetic algorithm and K-means clustering need more than one iterations for any size of the problem to obtain desired solution. Modified ART 1 is advantageous when the size of the problem increases.

In modified ART1, the threshold value greatly influences the number of cells obtained. The threshold value for each problem is varied from 1.5 to 2.5. It is observed that the number of cells equals to the total number of machines if the threshold value is set to zero. As the threshold value increases, the number of cells can be reduced as shown in Table 2. The threshold value for the problem 4 of size 6x8 is 2.00 and the modified grouping efficiency (MGE) obtained by the proposed algorithm is 79.85**%**. K-means algorithm and Genetic algorithm also produce same value of MGE for the problem number 4.

The supplementary procedure described in section 5 can be used to avoid cells with single machine that is encountered at times. The algorithm is flexible in the sense that maximum number of machines to be accommodated in a cell can be limited.

The modified grouping efficiency proposed in this study is evidently suitable to measure the performance of cell formation algorithm taking into account workloads on machines, weighting factor for voids, and exceptional elements.

VII. Conclusion

The proposed algorithm is tested with varied size problems from open literature and the solutions are compared with the solutions obtained from K-means clustering and GA. For smaller size problem solution i.e the number of exceptional elements obtained by ART 1 is matching with that of produced by other best algorithms found in the literature and for larger size problems the proposed ART1 outperforms other algorithms. Though GA is found to produce better results over the ART 1 for only medium size problems, the ART1 exhibits its superiority over GA and K-means for the required computational effort in terms of CPU time or number of iterations.

Since the algorithm uses simple network architecture the results are obtained in a single iteration whereas more number of iterations is required both in GA and K-means clustering depending on the size of the problem. Therefore, the modified ART1 is found to be computationally efficient for generating quick solutions for industrial applications. However, it has been observed that the proposed algorithm is sensitive to the order of presentation of the input vectors due to decaying of the stored template leading to unsystematic weight updating. Therefore, it may produce different solution if the order of presentation of the input vectors is changed. When workload matrix is used as input, no effective method exists to address this limitation. The work can be further extended in future incorporating production data like machine capacity, production volume and product sequence with varied product type, layout considerations and material handling systems enhancing it to more generalized manufacturing environment. The software used for the proposed algorithm can be obtained from the authors on request.

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] G.Srinivasan, T.T.Narendran, (1991) A non hierarchical clustering algorithm for group technology GRAFICS, International Journal of Production Research 29 (3) 463-478.

[9] F.F.Boctor, (1991)A linear formulation of the machine-partcell formation problem, International Journal of Production Research 29 343-346.

[10] 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.

[11] .Jayakrishnannair, T.T.Narendran, (1998) CASE: A clustering algorithm for cell formation with sequence data, International Journal of Production Research, 36, 157-179.

[12] 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.

[13] Wu, Low, Wu, (2004) A tabu search approach to the cell formation problem, International Journal of Advanced Manufacturing Technology 23, 916-924.

[14] Y. Kao, Y.B. Moon, (1991) A Unified Group Technology Implementation Using the backpropagation Learning Rule of Neural Networks, Computers and Industrial Engineering 20 (4)) 425-437.

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

[16] S.Kaparthi, N.C.Suresh, (1992) Machine Component cell formation in GT: A NN approach, International Journal of Production Research 30 1353-1367.

[17] Chen, C.S.Cheng, A neural network-based cell formation algorithm in cellular manufacturing, International Journal of Production Research 33 (2) (1995) 293-318.

[18] Yoh-Han Pao, Adaptive Pattern Recognition and Neural Networks, Addison –Wesley Publishing Company, New York Inc, 1989.

[19] Kumar.K.R 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.

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

[21] Syswerda, Scheduling optimization using genetic algorithm. Handbook of Genetic Algorithms, Van Nostrand Reinhold, Newyork. 1991.

[22] Venkumar.P and A.N.Haq, "Manufacturing Cell Formation Using Modified ART1 Networks," International Journal of Advanced Manufacturing Technology, 26 (2005), 906-916.

i / j	P1	P2	P3	P4	P5	P6	P7	P8
M1	0	0.53	0	0.99	0	0	0.83	0
M2	0.91	0.82	0.83	0	0.91	0.92	0.86	0.97
M3	0	0	0.79	0	0	0.56	0	0.88
M4	0	0	0	0.53	0	0	0.51	0
M5	0.98	0	0.83	0	0.71	0.58	0	0.54
M6	0	0	0	0.54	0	0	0.74	0

Table 1a. Input Matrix

i / j	**P4**	**P7**	**P1**	**P2**	**P3**	**P5**	**P6**	**P8**
M1	0.99	0.83						
M4	0.53	0.51						
M5	0.54	0.74						
M2			0.91	0.82	0.83	0.91	0.92	0.97
M3			0	0	0.79	0	0.56	0.88
M5			0.98	0.00	0.83	0.71	0.58	0.54

Table 1b. Output Matrix

S.N	Problem Size	CPU Time (Sec)		
		K-means	GA	Modified ART1
1	5 x 7	0.0913	0.1098	0.0602
2	5 x 7	0.0913	0.1098	0.0601
3	5 x 18	0.1013	0.1533	0.0693
4	6 x 8	0.1009	0.1302	0.0635
5	7 x 11	0.1421	0.2310	0.0698
6	7 x 11	0.1421	0.2315	0.0691
7	8 x 12	0.1652	0.2632	0.0723
8	8 x 20	0.2033	0.3213	0.0726
9	8 x 20	0.2365	0.3320	0.0732
10	10 x 10	0.2965	0.4891	0.0921
11	10 x 15	0.3354	0.5561	0.1032
12	14 x 24	0.4462	0.7884	0.2118
13	14 x 24	0.4501	0.7890	0.2118
14	24 x 16	0.6691	0.9320	0.2360
15	16 x 30	0.5264	1.0439	0.3963
16	20 x 20	0.6852	1.8925	0.4522
17	20 x 35	0.8653	2.0182	0.5023
18	20 x 35	0.8660	2.0231	0.5023
19	24 x 40	1.3566	3.9616	0.7286
20	24 x 40	1.3593	3.9835	0.7293
21	30 x 41	2.6850	5.3222	0.9120
22	30 x 50	3.9910	6.0354	1.8105
23	30 x 50	4.6330	6.0439	1.8549

Table 4. CPU Time for test problems

S.N	Threshold Value (ρ)	Number of cells	Machines allocated	Parts allocated
1	2	Cell –1	0 3 6 7 10 11	1 3 6 8 11 17 21 29
		Cell –2	1	0 9 15
		Cell –3	2 5 8 14	4 18 22 24 26 27 28
		Cell –4	4 9 13 15	5 7 10 13 14 16 20 23 25
		Cell –5	12	2 12 19
2	2.3	Cell –1	0 3 6 7 10 11	1 3 6 8 11 17 21 29
		Cell –2	1	0 9 15 19
		Cell –3	2 5 8 12 14	2 4 12 18 22 24 26 27 28
		Cell –4	4 9 13 15	5 7 10 13 14 16 20 23 25
3	2.5	Cell –1	0 3 6 7 10 11	1 3 6 8 11 17 21 29
		Cell –2	1 2 5 8 12 14	0 4 9 12 15 18 19 22 24 26 27 28
		Cell –3	4 9 13 15	2 5 7 10 13 14 16 20 23 25

Table 2. Effect of Threshold value on number of cells for the problem size 16 x 30.

S. No	Problem No as referred by Gancalves and Resende (2002)	Problem Size	Number of cells	Number of Exceptional Elements			Modified Grouping Efficiency		
				K-means	GA	ART 1	K-means	GA	ART 1
1	Problem 1	5 x 7	2	2	2	2	77.25	77.25	77.25
2	Problem 2	5 x 7	2	2	2	2	78.34	78.34	78.34
3	Problem 3	5 x 18	2	7	7	7	81.87	81.87	81.87
4	Problem 4	6 x 8	2	2	2	2	79.85	79.85	79.85
5	Problem 5	7 x 11	2	3	3	3	61.77	61.77	61.77
6	Problem 6	7 x 11	2	1	1	1	65.48	65.48	65.48
7	Problem 7	8 x 12	3	9	9	9	83.40	83.40	83.40
8	Problem 8	8 x 20	3	0	0	0	77.14	77.14	77.14
9	Problem 9	8 x 20	3	0	0	0	93.28	93.28	93.28
10	Problem 10	10 x 10	4	7	3	7	68.13	73.19	68.13
11	Problem 11	10 x 15	3	1	1	1	71.00	71.15	71.15
12	Problem 12	14 x 24	4	31	32	28	61.50	61.65	61.71
13	Problem 13	14 x 24	6	0	0	0	90.28	84.58	90.28
14	Problem 14	24 x 16	4	34	29	30	46.70	52.02	51.39
15	Problem 15	16 x 30	5	7	9	9	71.60	73.89	73.89
16	Problem 18	20 x 20	3	12	15	17	56.65	56.14	53.98
17	Problem 20	20 x 35	6	20	22	26	61.84	60.23	55.51
18	Problem 21	20 x 35	3	33	25	17	50.51	52.35	53.19
19	Problem 22	24 x 40	2	2	0	2	59.43	62.42	60.59
20	Problem 23	24 x 40	2	6	6	4	57.00	62.11	69.70
21	Problem 30	30 x 41	2	28	28	25	60.00	59.74	61.30
22	Problem 31	30 x 50	3	15	20	15	64.81	64.64	64.81
23	Problem 32	30 x 50	2	42	29	22	49.13	50.72	51.10

Table 3. Performance of the proposed modified ART1 algorithm with K-means and GA.