J Intell Manuf (2012) 23:1047–1061
DOI 10.1007/s10845-010-0446-8

A genetic algorithm with proper parameters for manufacturing cell formation problems

Tugba Saraç · Feristah Ozcelik

Received: 13 August 2009 / Accepted: 2 August 2010 / Published online: 15 August 2010

Abstract One fundamental problem in cellular manufacturing is the formation of product families and machine cells. Many solution methods have been developed for the cell formation problem. Since efficient grouping is the prerequisite of a successful Cellular Manufacturing installation the research in this area will likely be continued. In this paper, we consider the problem of cell formation in cellular manufacturing systems with the objective of maximizing the grouping efficacy. We propose a Genetic Algorithm (GA) to obtain machine-cells and part-families. Developed GA has three different selection and crossover operators. The proper operators and parameters of the GA were determined by design of experiments. A set of 15 test problems with various sizes drawn from the literature is used to test the performance of the proposed algorithm. The corresponding results are compared to several well-known algorithms published. The comparative study shows that the proposed GA improves the grouping efficacy for 40% of the test problems.

Keywords Cell formation problem · Genetic algorithms · Grouping efficacy · Design of experiments

Introduction

Group technology (GT) groups parts that have similar design characteristics or manufacturing characteristics into part families in order to make manufacturing systems more efficient and productive. Cellular manufacturing is the implementation of group technology in the manufacturing process. Cellular manufacturing decomposes the entire production system into several mutually separable production cells, then assigns machines to these cells to process one or more part families (Wu et al. 2009). One fundamental problem in cellular manufacturing is the formation of product families and machine cells. The objective of this cell formation problem is to form perfect (i.e. disjoint) groups in which products do not have to move from one cell to the other for processing (Gonçalves and Resende 2004).

Many solution methods have been developed for the cell formation problem. These include cluster analysis approaches, graph partitioning approaches, and mathematical programming approaches, among others. McCormick et al. (1972) developed a matrix clustering approach called the bond energy algorithm. Carrie (1973) developed an approach based on cluster analysis and similarity coefficients. King (1980) used a rank order clustering technique to diagonalize the machine-part incidence matrix and later improved this approach (King and Nakornchai 1982). Chandrasekharan and Rajagopalan (1986a) presented a modified version of the rank order clustering approach and a non-hierarchical clustering method (1986b) which used the ideal-seed method. ZODIAC, zero-one data: ideal seed algorithm for clustering was an improved version of their earlier work (Chandrasekharan and Rajagopalan 1987). Srinivasan (1994) developed a clustering algorithm that utilized minimum spanning trees. GRAPHICS, another non-hierarchical clustering approach, was presented by Srinivasan and Narendran (1991). Graph-based techniques include those by Rajagopalan and Batra (1975) and Vannelli and Kumar (1986). Mathematical programming approaches that model the clustering problem as a p-median or assignment problem include those by Kusiak (1987) and Srinivasan et al. (1990).

T. Saraç (✉) · F. Ozcelik
Industrial Engineering Department, Eskisehir Osmangazi University, Eskisehir, Turkey
e-mail: tsarac@ogu.edu.tr

F. Ozcelik
e-mail: fdurmaz@ogu.edu.tr

Other approaches that have been applied to the cell formation problem include simulated annealing (Boctor 1991; Sofianopoulou 1997; Baykasoglu 2004; Islier 2005), tabu search (Lozano et al. 1999; Wu et al. 2004; Islier 2005), GAs (Joines et al. 1996; Cheng et al. 1998; Mak et al. 2000; Gonçalves and Resende 2004; Islier 2005; Doulabi et al. 2009), GGA (Brown and Sumichrast 2001), neural networks (Guerrero et al. 2002; Ozturk et al. 2006), ant systems (Islier 2005) and fuzzy logic (Xu and Wang 1989; Arıkan and Gungor 2005).

Since efficient grouping is the prerequisite of a successful CM installation the research in this area will likely be continued. The purpose of this study is to develop a procedure that is efficient and effective for obtaining machine-part groupings. To achieve this, a genetic algorithm working with different crossover operators (single-point, double-point and uniform) and reproduction types (roulette wheel, stochastic sampling and tournament) is developed and the proper GA parameters and operators are selected by using design of experiments. A set of 15 test problems with various sizes drawn from the literature is used to test the performance of the proposed algorithm. The corresponding results are compared to several well-known algorithms published.

The remainder of this article is organized as follows. In section "Cell formation problem", we describe the cell formation problem. The proposed genetic algorithm is presented in section "The Proposed genetic algorithm". Section "Computational results" shows the computational results on problems with various sizes, and section "Conclusion" concludes the paper.

Cell formation problem

Grouping is the backbone of a manufacturing cell formation. The input to the cell formation problem is the binary machine-part incidence matrix derived from production flow analysis (PFA) chart. Columns of an incidence matrix represent machines and rows represent parts. A matrix element a_{ij} is '1' if machine i is used to process part j, and '0' if otherwise. Figure 1a shows an example of PFA chart for six parts and four machines. The aim is to convert the incidence matrix into a block-diagonalized form by certain column and row operations. Here specific subsets of parts (part families) are linked to certain machine subsets to form cells. After rearrangement of rows and columns, three blocks can be obtained along the diagonal of the solution matrix in Fig. 1b.

Several measures of goodness of machine-part groups in cellular manufacturing have been proposed. Grouping efficacy is one of the most widely used measures (Keeling et al. 2007). The grouping efficacy can be defined as

$$\text{Grouping efficacy} = \mu = \frac{N_1 - N_1^{\text{Out}}}{N_1 + N_0^{\text{In}}}$$

	M1	M2	M3	M4
P1	0	0	1	0
P2	1	0	0	0
P3	0	1	0	1
P4	0	0	1	0
P5	0	1	0	1
P6	1	0	0	0

(a)

	M4	M2	M3	M1
P3	1	1	0	0
P5	1	1	0	0
P1	0	0	1	0
P4	0	0	1	0
P2	0	0	0	1
P6	0	0	0	1

(b)

Fig. 1 Block diagonalization of an incidence matrix: **a** incidence matrix and **b** block-diagonalized matrix

where

N_1 total number of 1's in machine-part incidence matrix;
N_1^{Out} total number of 1's outside the diagonal blocks;
N_0^{In} total number of 0's inside the diagonal blocks.

The closer the grouping efficacy is to 1, the better will be the grouping. Grouping efficacy is used as the fitness function of the GA described in the following section.

The proposed genetic algorithm

Genetic algorithms are powerful and broadly applicable in stochastic search and optimization techniques based on principles from evolution theory (Gen and Cheng 1997). GAs, which are different from normal optimization and search procedures: (a) Work with a coding of the parameter set, not the parameters themselves. (b) Search from population of points, not a single point. (c) Use payoff (objective function) information, not derivatives or other auxiliary knowledge. (d) Use probabilistic transition rules, not deterministic rules (Goldberg 1989).

Cell formation problems are NP-hard (King and Nakornchai 1982) so an efficient search heuristic will be useful for tackling such a problem. In this study, we developed a GA working with different crossover operators (single-point, double-point and uniform) and reproduction types (roulette wheel, stochastic sampling and tournament) to solve the cell formation problem. The developed GA is discussed in detail below.

Representation

The genetic algorithm presented in this paper uses a random key alphabet U(0,1). An important feature of random keys is that all offspring formed by crossover are feasible solutions. This is accomplished by moving much of the feasibility issue into the fitness evaluation procedure. If any random key vector can be interpreted as a feasible solution, then any crossover is feasible. Through the dynamics of the genetic algorithm, the system learns the relationship between random key vectors and solutions with good objective values.

	machines						parts					
	1	2	3	4	5	6	1	2	3	4	5	6
0.3	**0.2**	**0.8**	**0.3**	**0.5**	**0.7**	**0.1**	**0.4**	**0.5**	**0.8**	**0.2**	**0.1**	**0.3**

Fig. 2 A sample chromosome for 6-machines, 6-parts problem

The proposed chromosome consists of three sections: one representing the number of cells to be formed, one representing the parts, one representing the machines. The first section contains only one gene. The second and the third sections contain m (number of the machines) and n (the number of the parts) genes respectively. All genes in the chromosome have values between 0 and 0.99 and the genes in second and third sections are decoded by using the *number of cells* (c) which is determined by the first gene. For example, a solution for the 6-machines, 6-parts problem can be represented as the chromosome is given in Fig. 2. Since this example contains 6 machines, the maximum c can be 6 so the range [0, 1) is divided into 6 equal ranges as given below.

[0, 0.1667)	1 cell
[0.1667, 0.3334)	2 cells
[0.3334, 0.5001)	3 cells
[0.5001, 0.6667)	4 cells
[0.6667, 0.8335)	5 cells
[0.8335, 1)	6 cells

The number of the cells is determined by using the range that the value of the first gene is in. In this example, the first gene's value is 0.3. It is in the second range [0.1667, 0.3334), so the number of cells is 2. For the rest of the genes of the chromosome, the range [0, 1) is divided into equal c ranges. In this example, this number is 2 so range [0, 1) is divided into two equal ranges. If the value of genes is in the first range [0, 0.5) it means that related machine or part is assigned to the first cell otherwise it is in the second cell. The solution, chromosome given in Fig. 2 represents, has two cells with the cell 1 containing parts {1, 4, 5, 6} and machines {1, 3, 6} and cell 2 containing parts {2, 3} and machines {2, 4, 5}.

Initial population is constructed by generating random numbers in range [0,1) for each genes in the chromosomes of the initial population.

Handling constraints

The central question in applying genetic algorithms to the constrained optimization is how to handle constraints because the genetic operators used to manipulate the chromosomes often yields infeasible offspring. In this study, the chromosome representation guaranties the feasibility. It is a big advantage because there is no need any constraint handling techniques.

Genetic operators

A classical GA is composed of three operators: reproduction, crossover and mutation. Operators taken into consideration for solving cell formation problems with GA are discussed as follows.

The reproduction operator allows individual strings to be copied for possible inclusion in the next generation. The chance that a string will be copied is based on the string's fitness value, calculated from a fitness function. We use three types of reproduction operators; roulette wheel, stochastic sampling, and tournament.

Crossover enables the algorithm to extract the best genes from different individuals and recombine them into potentially superior children. The proposed GA has three different crossover operators; single point, double point and uniform.

Reproduction and crossover alone can obviously generate a staggering amount of differing strings. However, depending on the initial population chosen, there may not be enough variety of strings to ensure the GA searches the entire problem space, or the GA may find itself converging on strings that are not quite close to the optimum it seeks due to a bad initial population. Some of these problems may be prevented by introducing a mutation operator into the GA. In proposed algorithm, a random number is generated for all genes. If the random number is smaller than mutation rate, the value of the gene is changed with a new random number in range [0,1). The value of gene is protected, if random number is bigger than the mutation rate.

When creating a new generation, there is always a risk of losing the fittest individuals. Using elitism, the fittest individuals are copied to the next generation. The other ones undergo the crossover and mutation. The elitism selection improves the efficiency of a GA considerably, as it prevents losing the best results. In our GA, the chromosome with the best fitness value is copied to the next generation directly.

We use two types of termination conditions with together. One of them checks whether the algorithm has run a fixed number (*nf*) of generations. And the other one stops the algorithm if the solution is same during an identical number (*ni*) of generation even if *ni* is smaller than the *nf*.

Computational results

The choice of parameters in a genetic algorithm is a major difficulty when it comes to their utilization. Design of experiment has been performed to identify appropriate values for GA parameters on three problems (small, medium and large sizes). The results are used to solve 15 cell formation instances collected from the literature. The proposed algorithm was coded in Visual Basic Application and implemented on a Pentium IV 1.60 GHz personal computer with 256 MB RAM.

Identifying efficient GA operators and parameters

In general, since the values of the GAs' operators and parameters significantly dependent upon the problem considered, the investigation of significant operators and parameters that affect the performance of the GAs and the determination of the optimum values of these become an important issue. Therefore, the GA operators and parameters that affect the performance of the GA, developed to solve cell formation problem, are examined. The proper operators and the optimum values of the parameters are explored using design of experiments. Five design factors were identified as potentially important for performance of the proposed GA: (A) population size, (B) crossover rate, (C) mutation rate, (D) crossover operator, and (E) reproduction type. Based on the related research, two possible levels for factor A and three possible levels for the other factors were considered as shown in Table 1.

To improve the accuracy of the comparisons among GA parameters by eliminating the variability among the problems, block design strategy is used.

Three typical problems were chosen, small (5 machines and 7 parts—problem N in Table 6), medium (24 machines and 40 parts—problem B in Table 6) and large (40 machines and 100 parts—problem L in Table 6). 2×3^4 full factorial experiment is designed for these three problems. A single replicate of a complete factorial experiment is run within each block (problem). As a result, $162 \times 3 = 486$ experiments are done. The analysis of variance (ANOVA) is then performed to determine the significant factors for the selected criterion. The ANOVA table is given in Table 2.

When a factor has an effect on the fitness value (V), it gets a low P value and high F (Fisher's test) value. It is investigated from Table 2 that, since $P_{Blocks}, P_C, P_E < 0.05$ we conclude that the main effects of Blocks, E (Reproduction type) and C (Mutation rate) effects solution success (V). Since $P_{A*E}, P_{C*E} < 0.05$, there are interactions between A and E, C and E. While determining the proper levels of the factors, we should start the factor with the highest F values and go on with decreasing order. In Table 2, after blocks, E has the highest F values. According to Fig. 3, level 3 of E should be selected for the best performance. Since interaction effect of

Table 1 Design factors and their levels

Factor	Level 1	Level 2	Level 3
A Population size	30	50	
B Crossover rate	0.60	0.75	0.90
C Mutation rate	0.001	0.005	0.01
D Crossover operator	Single-point	Double-point	Uniform
E Reproduction type	Roulette wheel	Stochastic sampling	Tournament

Table 2 ANOVA table

Source	DF	Seq SS	Adj SS	Adj MS	F	P
Blocks	2	6.16270	6.16270	3.08135	343.37	0.000
A	1	0.00001	0.00001	0.00001	0.00	0.978
B	2	0.01856	0.01856	0.00928	1.03	0.356
C	2	0.09770	0.09770	0.04885	5.44	0.005
D	2	0.01660	0.01660	0.00830	0.93	0.397
E	2	3.02999	3.02999	1.51499	168.82	0.000
A*B	2	0.00077	0.00077	0.00038	0.04	0.958
A*C	2	0.01630	0.01630	0.00815	0.91	0.404
A*D	2	0.00042	0.00042	0.00021	0.02	0.977
A*E	2	0.05431	0.05431	0.02716	3.03	0.050
B*C	4	0.01414	0.01414	0.00353	0.39	0.813
B*D	4	0.00946	0.00946	0.00237	0.26	0.901
B*E	4	0.00533	0.00533	0.00133	0.15	0.964
C*D	4	0.01969	0.01969	0.00492	0.55	0.700
C*E	4	0.41785	0.41785	0.10446	11.64	0.000
D*E	4	0.00577	0.00577	0.00144	0.16	0.958
Error	442	3.96649	3.96649	0.00897		
Total	485	13.83609				

DF degree of freedom, *seq SS* sequential sum of squares, *adj SS* adjusted sum of squares, *adj MS* adjusted mean squares, *F* test statistic, *P* probability

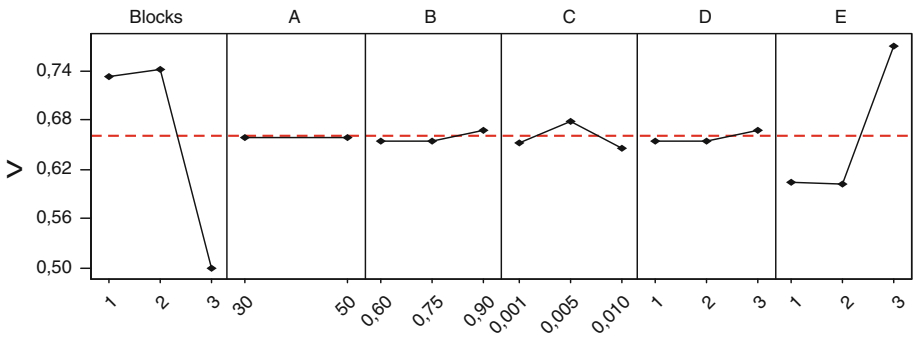


Fig. 3 Main effects plot—means for V

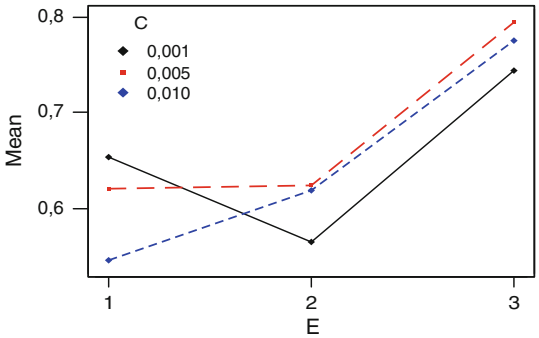


Fig. 4 Interaction plot C*E—means for V

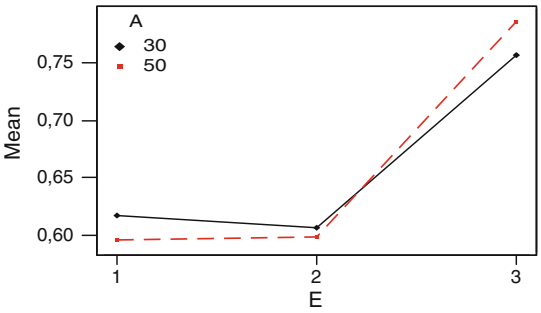


Fig. 5 Interaction plot A*E—means for V

C and E is more critical than main effect of C, the level of C should be selected according to interaction plot in Fig. 4 as 0.005.

Similarly, the level of A is determined as 50 based on the Fig. 5. B and D don't have an effect on the fitness value (V).

As a result, we selected the following levels of the factors A, B, C, D and E as 50, 0.9, 0.005, uniform and tournament, respectively to use our tests and these are called as parameter set 1.

Since blocks have the highest F value, the additional analysis is performed separately for each problem size.

Analysis for small size problems

As seen in Table 3, none of the factors has an effect on the fitness value (V). Proper GA parameters are determined by using the main effects plot given in Fig. 6. We selected the following levels of the factors A, B, C, D and E as 30, 0.9, 0.01, two point and tournament respectively for small size problems.

Analysis for medium size problems

It is investigated from Table 4 that, since $P_E, P_C < 0.05$ we conclude that the main effects of E (Reproduction type) and C (Mutation rate) effects solution success (V). Since $P_{A*C}, P_{C*D}, P_{C*E} < 0.05$, there are interactions between A and C, C and D, C and E. While determining the proper levels of the factors, we should start the factor with the highest F values and go on with decreasing order. In Table 4, E has the highest F values. According to Fig. 7, level 3 of E should be selected for the best performance. Since interaction effect of C and E is more critical than main effect of C, the level of C should be selected according to interaction plot in Fig. 8 as 0.01.

Similarly, the level of A is determined as 30 based on the Fig. 9 and the level of D is determined as double point crossover based on the Fig. 10. B doesn't have an effect on the fitness value (V).

As a result, we selected the following levels of the factors A, B, C, D and E as 30, 0.9, 0.01, double point crossover and tournament respectively for medium size problems.

Analysis for large size problems

As seen in Table 5, since $P_E, P_C, P_B, P_D < 0.05$ we conclude that the main effects of E, C, B and D effects solution success (V). Since $P_{A*E}, P_{C*E} < 0.05$, there are interactions between A and E, C and E. While determining the proper levels of the factors, we should start the factor with the highest F values and go on with decreasing order. In Table 5, E has the highest F values. According to Fig. 11, level 3 of E should be selected for the best performance. Since interaction effect of C and E is more critical than main effect of C, the level of C should be selected according to interaction plot in Fig. 12 as 0.005.

Table 3 ANOVA table for small size problems

Source	DF	Seq SS	Adj SS	Adj MS	F	P
A	1	0.0002261	0.0002261	0.0002261	1.00	0.319
B	2	0.0004522	0.0004522	0.0002261	1.00	0.371
C	2	0.0004522	0.0004522	0.0002261	1.00	0.371
D	2	0.0004522	0.0004522	0.0002261	1.00	0.371
E	2	0.0004522	0.0004522	0.0002261	1.00	0.371
A*B	2	0.0004522	0.0004522	0.0002261	1.00	0.371
A*C	2	0.0004522	0.0004522	0.0002261	1.00	0.371
A*D	2	0.0004522	0.0004522	0.0002261	1.00	0.371
A*E	2	0.0004522	0.0004522	0.0002261	1.00	0.371
B*C	4	0.0009044	0.0009044	0.0002261	1.00	0.410
B*D	4	0.0009044	0.0009044	0.0002261	1.00	0.410
B*E	4	0.0009044	0.0009044	0.0002261	1.00	0.410
C*D	4	0.0009044	0.0009044	0.0002261	1.00	0.410
C*E	4	0.0009044	0.0009044	0.0002261	1.00	0.410
D*E	4	0.0009044	0.0009044	0.0002261	1.00	0.410
Error	120	0.0271326	0.0271326	0.0002261		
Total	161	0.0364029				

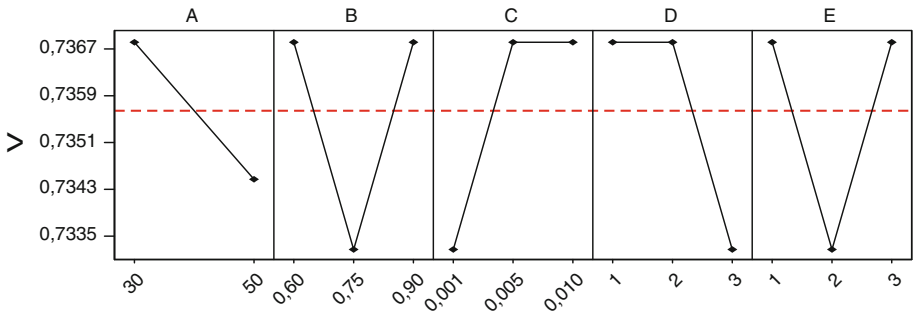


Fig. 6 Main effects plot—means for V

Table 4 ANOVA table for medium size problems

Source	DF	Seq SS	Adj SS	Adj MS	F	P
A	1	0.000000	0.000000	0.000000	0.00	0.994
B	2	0.012819	0.012819	0.006410	1.39	0.252
C	2	0.080438	0.080438	0.040219	8.74	0.000
D	2	0.021880	0.021880	0.010940	2.38	0.097
E	2	0.606920	0.606920	0.303460	65.96	0.000
A*B	2	0.004401	0.004401	0.002201	0.48	0.621
A*C	2	0.036743	0.036743	0.018371	3.99	0.021
A*D	2	0.010605	0.010605	0.005302	1.15	0.319
A*E	2	0.017364	0.017364	0.008682	1.89	0.156
B*C	4	0.017501	0.017501	0.004375	0.95	0.437
B*D	4	0.015506	0.015506	0.003877	0.84	0.501
B*E	4	0.000729	0.000729	0.000182	0.04	0.997
C*D	4	0.055052	0.055052	0.013763	2.99	0.021
C*E	4	0.384444	0.384444	0.096111	20.89	0.000
D*E	4	0.007808	0.007808	0.001952	0.42	0.791
Error	120	0.552057	0.552057	0.004600		
Total	161	1.824266				

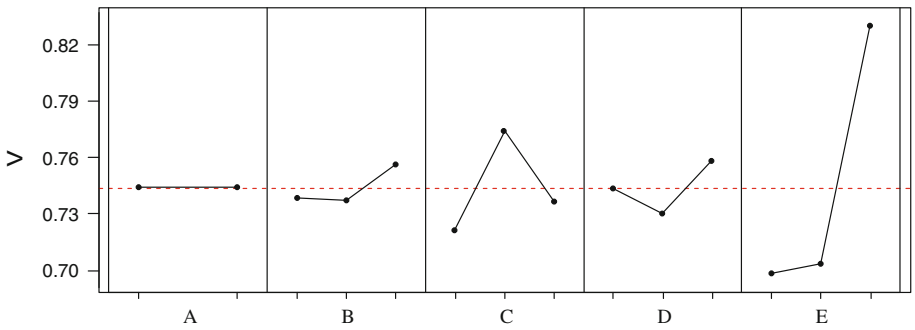


Fig. 7 Main effects Plot—means for V

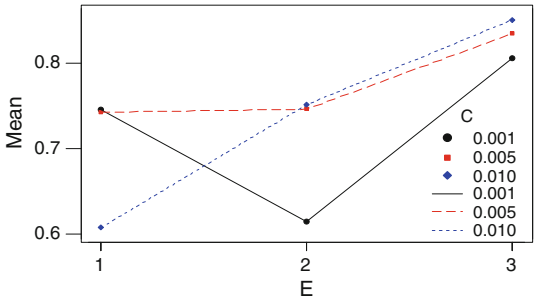


Fig. 8 Interaction plot C*E—means for V

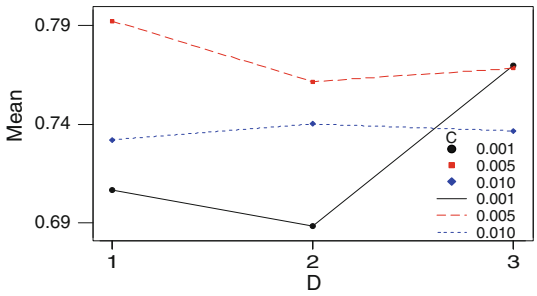


Fig. 10 Interaction plot C*D—means for V

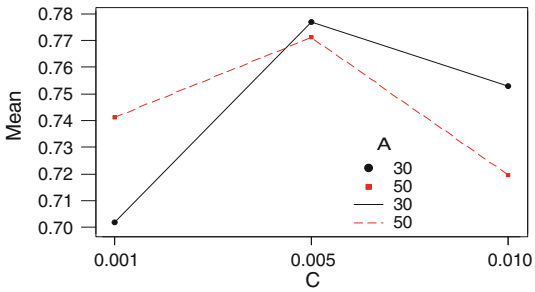


Fig. 9 Interaction plot A*C—means for V

Similarly, the level of A is determined as 50 based on the Fig. 13. The level of B and D are determined as 0.9 and uniform crossover respectively, based on the Fig. 11.

As a result, we selected the following levels of the factors A, B, C, D and E as 50, 0.9, 0.005, uniform and tournament respectively for large size problems.

The selected parameters and operators by analyzing all of the problem types separately are called as parameter set 2. In addition to the values determined by design of experiments, the values of the termination parameters were set as $ni = 3{,}000$ and $nf = 500$. Problem B is solved by the proposed GA with the selected parameters. Figure 14 shows the convergence of the algorithm. After 495 generations passed, the algorithm reached the best fitness value.

Comparing GA with other artificial intelligence techniques

The functionality and performance of the proposed algorithm is examined by using standard data sets from the literature. A data set consists of a machine-part incidence matrix and the number of cells (c) in the corresponding solution. The characteristics of the used data sets are given in Table 6.

Test problems are solved with the proposed GA by using the parameter set 1 and parameter set 2 that are determined by design of experiments. While the problems were being solved with parameter set 2, the sizes that are given in the second column of Table 7 were used to select the GA parameters. All of the test problems are replicated five times and the best grouping efficacy (z_{best}), mean of grouping efficacy (z_{mean}), standart deviation of the grouping efficacy (std of z) and mean of the generation number where best grouping efficacy was obtained are given in Table 7. In the table the best results are marked as bold. According to the table, parameter set 2 improves the values of the best grouping efficacy that is obtained by parameter set 1 for 4 (27%) problems.

The best results of the proposed GA are also compared with those of algorithms reported in the literature, i.e., simulated annealing (SA), tabu search (TS), genetic algorithms (GA), and ant system (AS) developed by Islier (2005) and competitive neural network (CNN) of Ozturk et al. (2006). As shown in Table 8, results obtained by the proposed GA are better than or equal to those reported results (except problem I). More specifically, the algorithm obtains values of the grouping efficacy that are equal to the best ones found in the literature for eight problems and improves the values of the grouping efficacy for six problems. Improvements are made by same c values with sources for one problem (E) and by one more c values from sources for the other five (H, F, G, K and N).

Please note that the proposed GA is better than Islier's GA. Islier (2005) developed a classic GA with single point crossover operator and roulette wheel selection mechanism.

Table 5 ANOVA table for large size problems

Source	DF	Seq SS	Adj SS	Adj MS	F	P
A	1	0.00036	0.00036	0.00036	0.19	0.662
B	2	0.01552	0.01552	0.00776	4.13	0.019
C	2	0.10611	0.10611	0.05306	28.21	0.000
D	2	0.01550	0.01550	0.00775	4.12	0.019
E	2	4.95651	4.95651	2.47826	1317.75	0.000
A*B	2	0.00196	0.00196	0.00098	0.52	0.595
A*C	2	0.00576	0.00576	0.00288	1.53	0.220
A*D	2	0.00760	0.00760	0.00380	2.02	0.137
A*E	2	0.07121	0.07121	0.03560	18.93	0.000
B*C	4	0.00920	0.00920	0.00230	1.22	0.305
B*D	4	0.01189	0.01189	0.00297	1.58	0.184
B*E	4	0.01418	0.01418	0.00355	1.89	0.117
C*D	4	0.01335	0.01335	0.00334	1.77	0.138
C*E	4	0.34821	0.34821	0.08705	46.29	0.000
D*E	4	0.00968	0.00968	0.00242	1.29	0.279
Error	120	0.22568	0.22568	0.00188		
Total	161	5.81272				

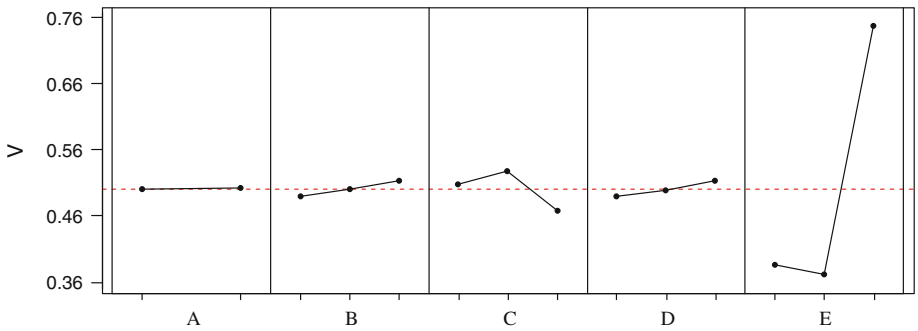


Fig. 11 Main effects plot—means for V

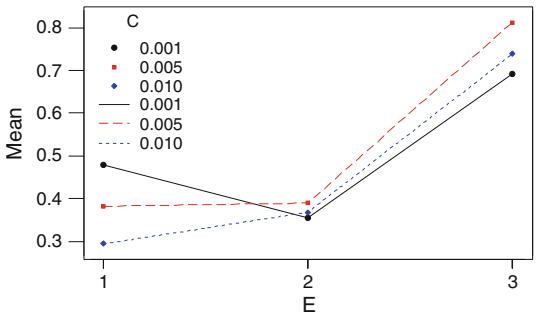


Fig. 12 Interaction plot C*E—means for V

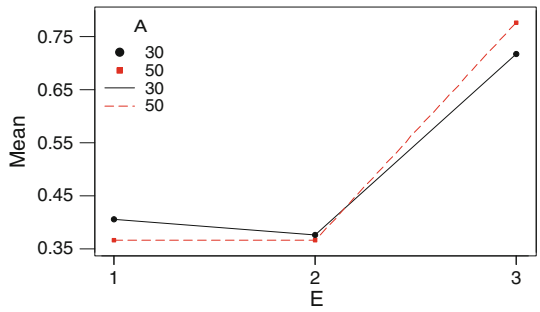


Fig. 13 Interaction plot A*E—means for V

He determined the GA parameters by making use of the previously gained experience on GAs, a population of 30, a reproduction rate of 0.8, a crossover rate of 0.9 and a mutation rate of 0.1. But in this study, three different crossover operators and selection mechanisms are coded. The best ones and the other important GA parameters such as population size, crossover rate and mutation rate are determined by design of experiments.

To further evaluate the performance of the selected GA operators and parameters, another test was run. In this test, the proposed algorithm is modified as to solve the cell formation problem with the chromosome design and the operators of Islier's GA. To show the performance of the proposed parameter set, test problems are solved by using the chromosome design of Islier's GA with both the parameter set of him (population size is 30, crossover rate is 0.90, mutation rate is 0.1, single point crossover and roulette wheel) and us (set 2). The results

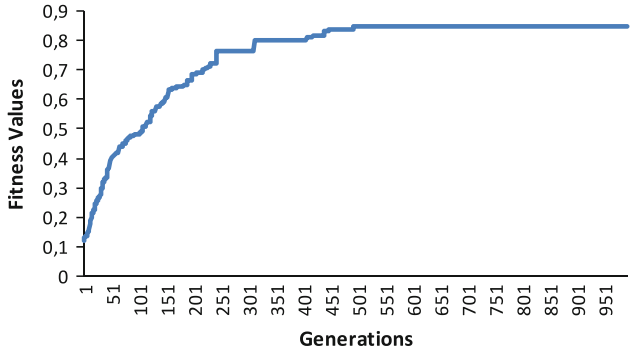


Fig. 14 The best values of generations of problem B

are given in Table 9. The first columns of the tables refer to the test problems. GA is replicated five times with both of the Islier's parameters and the parameter set 2. The best obtained values of the grouping efficacy (z_{best}), the mean (z_{mean}) and standard deviation (std of z) of the grouping efficacy and the mean of the generation number where best grouping efficacy was obtained (mean of # gen.) of the solutions are given separately for each parameter set in the tables. The results show that the proposed parameter set improves the performance of Islier's GA averagely 88% and even if the chromose design changes, the proposed parameter set is still quite successful.

The block diagonal matrices for the best solutions found by proposed GA for all problems are given in Appendix A.

Conclusions

In this paper, we propose a GA for the cell formation problem with the objective of maximizing the grouping efficacy. The control parameters that affect the performance of the GA, developed to solve cell formation problem, are examined and the proper parameters are explored using design of experiments. Computational experience with the algorithm, on a set of 15 GT problems from the literature, has shown that it performs remarkably well. The algorithm obtained solutions that are at least as good as the ones found the literature except one problem. For 40% of the problems, the algorithm improved the previous solutions.

Computational results have shown that the proposed parameter set improves the performance of classic GA taken

Table 6 Data sets used and their characteristics

Source	Set	Number of machines	Number of parts	Number of cells
Data sets 1 to 7 of	A	24	40	7
Chandrasekharan and	B	24	40	7
Rajagopalan (1989)	C	24	40	7
	D	24	40	7
	E	24	40	7
	F	24	40	7
	G	24	40	7
Carrie (1973)	H	20	35	4
Zolfaghari and Liang (2002)	I	7	8	3
Askin and Standrige (1993)	J	6	8	3
Burbidge (1977)	K	16	43	5
Chandrasekharan and Rajagopalan (1987)	L	40	100	10
Chan and Milner (1982)	M	10	15	3
Waghodekar and Sahu (1984)	N	5	7	2
deWitte (1980)	O	12	19	3

Table 7 Comparison of parameter set 1 and parameter set 2

Set	Size	GA with parameter set 1					GA with parameter set 2				
		z_{best}	z_{mean}	Std of z	# Cells	Mean of # gen.	z_{Best}	z_{mean}	Std of z	# Cells	Mean of # gen.
A	M	**1.0000**	1.0000	0.0000	7	294	**1.0000**	1.0000	0.0000	7	255
B	M	**0.8511**	0.8511	0.0000	7	344	**0.8511**	0.8511	0.0000	7	313
C	M	**0.7351**	0.7351	0.0000	7	290	**0.7351**	0.7351	0.0000	7	381
D	M	**0.7551**	0.7137	0.0926	7	289	**0.7551**	0.7070	0.0671	7	531
E	M	0.4971	0.4676	0.0275	7	517	**0.5172**	0.4849	0.0376	7	682
F	M	0.4255	0.4069	0.0155	7	571	**0.4444**	0.4314	0.0132	8	749
G	M	0.4149	0.3860	0.0200	7	577	**0.4229**	0.4010	0.0164	8	620
H	M	**0.7831**	0.7823	0.0018	5	165	**0.7831**	0.7823	0.0018	5	282
I	S	**0.6389**	0.6197	0.0113	3	35	**0.6389**	0.6285	0.0142	3	28
J	S	**0.8889**	0.8889	0.0000	3	24	**0.8889**	0.8889	0.0000	3	41
K	M	0.5346	0.4997	0.0305	6	473	**0.5380**	0.5216	0.0156	6	577
L	L	**0.8403**	0.8403	0.0000	10	962	**0.8403**	0.8403	0.0000	10	962
M	S	**0.8182**	0.7390	0.1084	3	37	**0.8182**	0.8182	0.0000	3	61
N	S	**0.7500**	0.7297	0.0224	3	10	**0.7500**	0.7395	0.0059	3	10
O	S	**0.5657**	0.5641	0.0036	3	86	**0.5657**	0.5625	0.0029	3	82

Table 8 Test results

Set	SA	TS	GA	AS	CNN	Proposed GA	
						Grouping efficacy	Number of cells
A	0.1485	0.1826	0.2252	0.7047	**1.000**	**1.000**	7
B	0.1367	0.1826	0.2524	0.6149	**0.851**	**0.851**	7
C	0.1423	0.1903	0.2336	0.4667	**0.735**	**0.735**	7
D	0.1682	0.2000	0.2217	0.4971	**0.755**	**0.755**	7
E	0.1446	0.1739	0.2284	0.3575	0.419	**0.517**	7
F	0.1383	0.1940	0.2342	0.3208	0.397	**0.444**	8
G	0.1494	0.1937	0.2182	0.3100	0.385	**0.423**	8
H	0.1992	0.2678	0.3839	0.6940	0.757	**0.783**	5
I	0.6000	0.4444	**0.7391**	**0.7391**	0.621	0.639	3
J	0.6190	0.5217	**0.8889**	**0.8889**	**0.889**	**0.889**	3
K	0.1757	0.2110	0.2925	0.3925	0.511	**0.538**	6
L	0.0853	0.1177	0.1320	0.3956	**0.840**	**0.840**	10
M	0.3784	0.3846	0.7857	**0.8182**	**0.818**	**0.818**	3
N	0.7368	0.5455	0.7368	0.7368	0.737	**0.750**	3
O	0.3571	0.3455	0.5288	0.5500	**0.566**	**0.566**	3

Table 9 Comparison of parameter set of Islier and parameter set 2

Set	GA with classical chromosome structure							
	Parameter set of Islier				Parameter set 2			
	z_{best}	z_{mean}	Std of z	Mean of # gen.	z_{best}	z_{mean}	Std of z	Mean of # gen.
A	0.3750	0.3603	0.0142	2228	**1.0000**	1.0000	0.0000	318
B	0.3711	0.3383	0.0344	2453	**0.8511**	0.8511	0.0000	398
C	0.3495	0.2910	0.0465	1911	**0.7351**	0.7351	0.0000	398
D	0.3495	0.3045	0.0350	2382	**0.7551**	0.7070	0.0671	479
E	0.2682	0.2496	0.0162	1767	**0.5172**	0.4788	0.0327	1002
F	0.2694	0.2361	0.0221	1449	**0.4317**	0.4106	0.0150	628
G	0.2524	0.2172	0.0213	886	**0.3969**	0.3828	0.0111	616
H	0.4734	0.4154	0.0510	1982	**0.7571**	0.7571	0.0000	157
I	**0.6389**	0.6352	0.0081	286	**0.6389**	0.6389	0.0000	32
J	**0.8889**	0.8889	0.0000	145	**0.8889**	0.8889	0.0000	44
K	0.3394	0.3145	0.0280	1194	**0.5115**	0.5030	0.0085	900
L	0.1471	0.1424	0.0039	1496	**0.8403**	0.8403	0.0000	745
M	**0.8182**	0.7479	0.0810	549	**0.8182**	0.7786	0.0885	67
N	**0.7368**	0.7368	0.0000	13	**0.7368**	0.7368	0.0000	15
O	0.5631	0.5438	0.0124	977	**0.5657**	0.563	0.0030	93

from literature (Islier 2005) averagely 88% and even if the chromose design changes, the proposed parameter set is still quite successful.

For further research, operation sequences can be used as data instead of binary incidence matrix. Thus, it will be possible to solve cell formation and cell layout problems simultaneously.

Acknowledgments The authors would like to thank A. Attila Islier, A. Sermet Anagun and also three anonymous referees for their valuable comments.

Appendix

See Fig. A1.

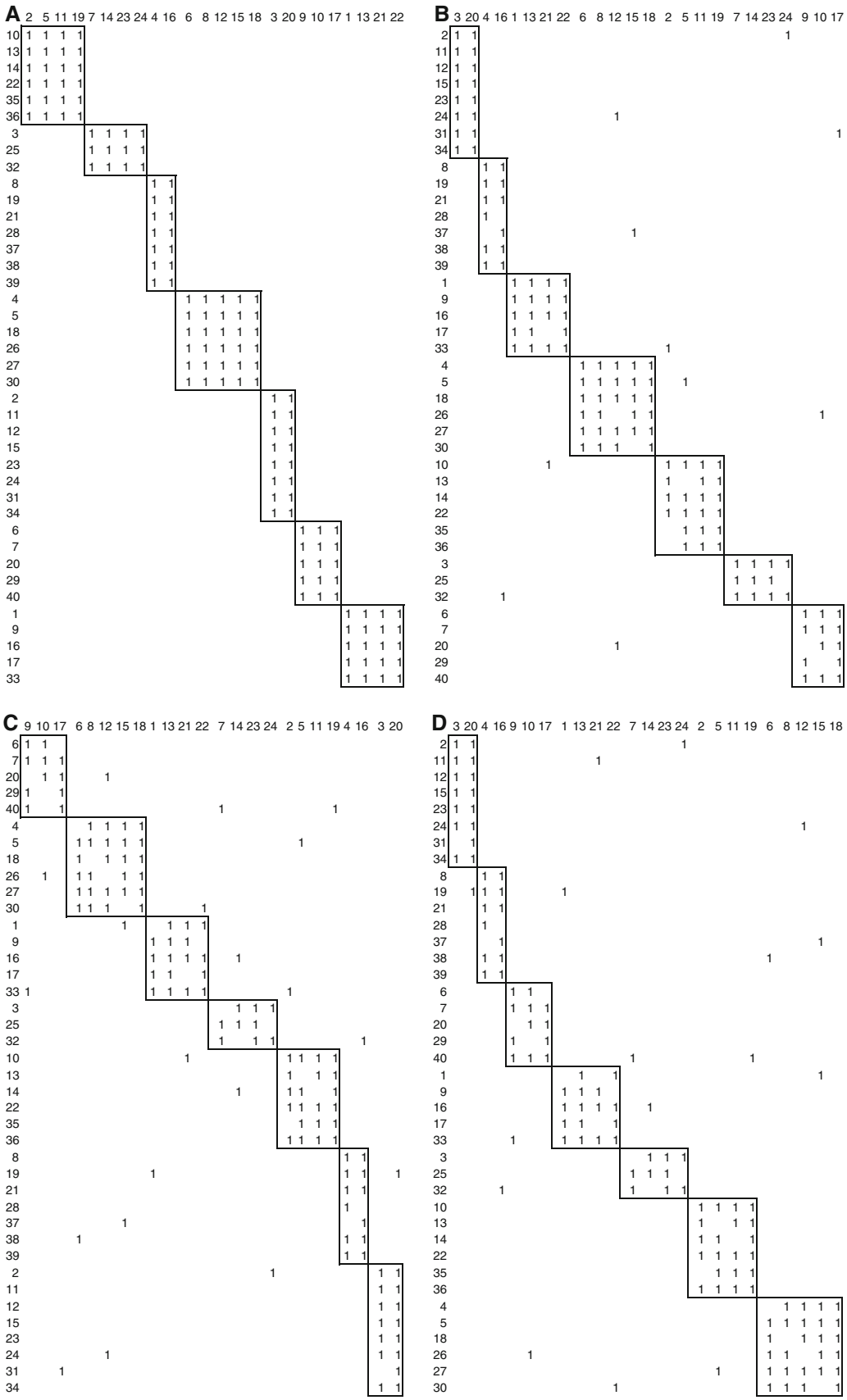


Fig. A1

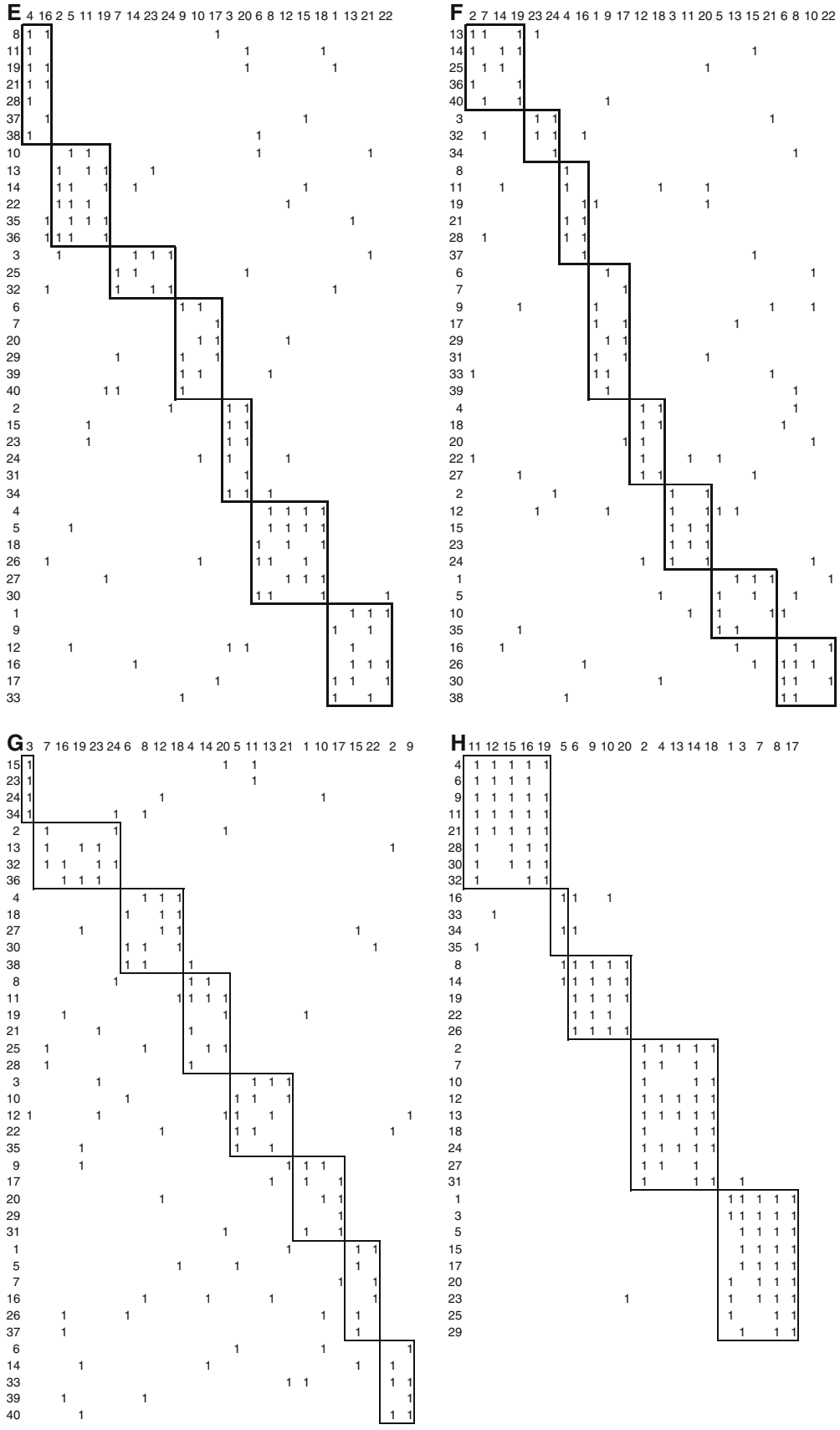


Fig. A1 continued

Fig. A1 continued

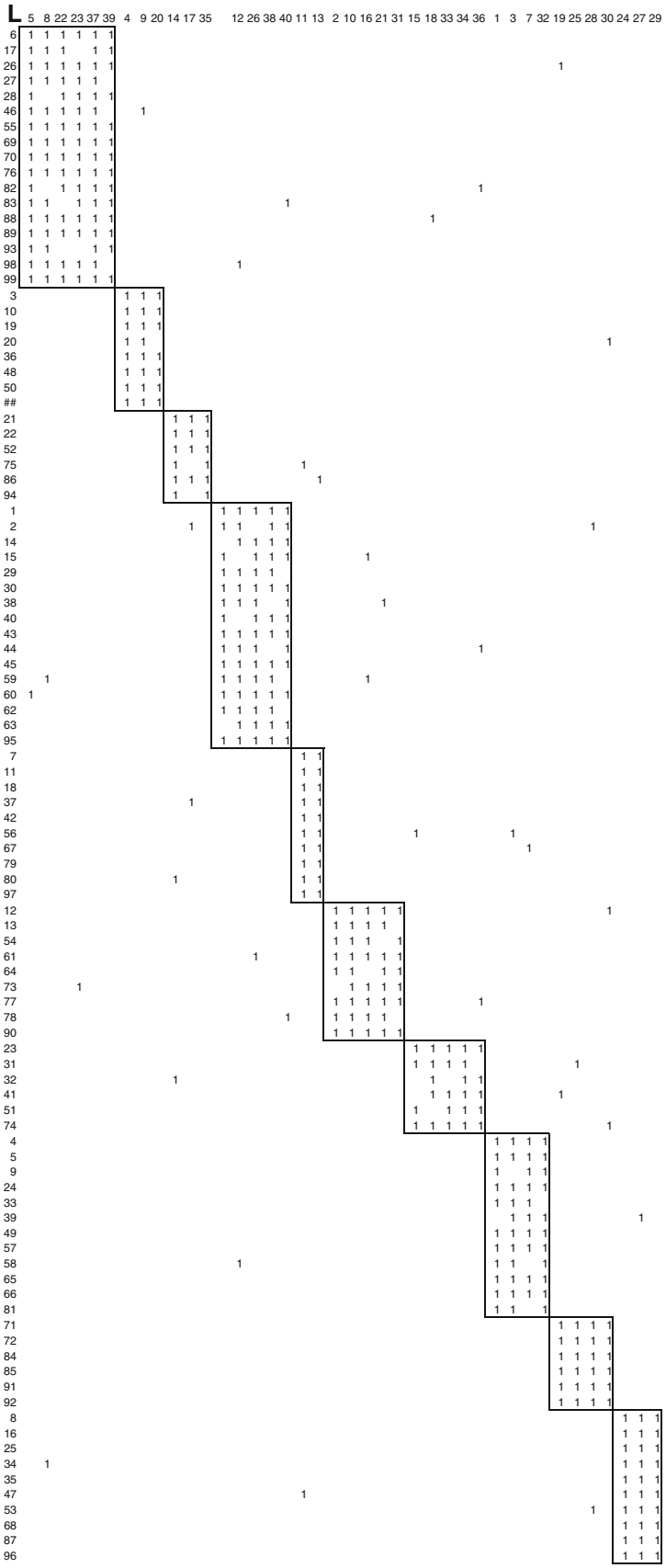
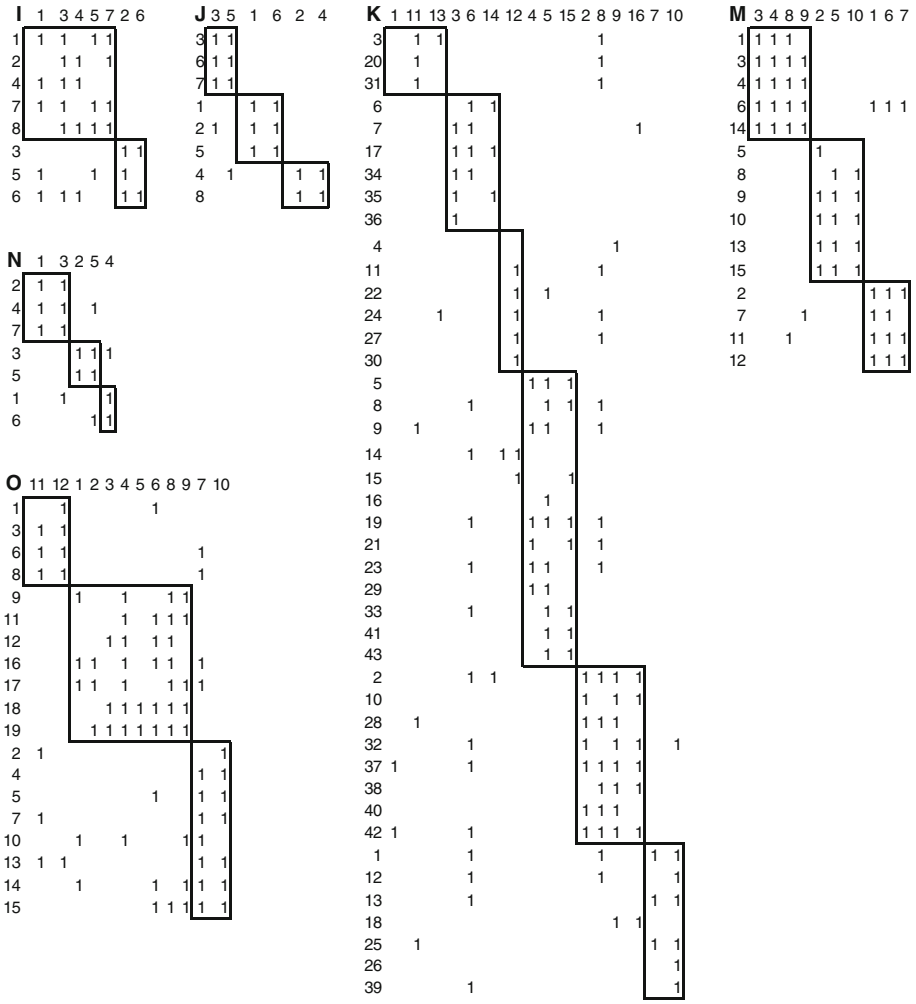


Fig. A1 continued



References

Arıkan, F., & Gungor, Z. (2005). A parametric model for cell formation and exceptional elements' problems with fuzzy parameters. *Journal of Intelligent Manufacturing, 16*, 103–114.

Askin, R. G., & Standrige, C. R. (1993). *Modeling and analysis of manufacturing systems*. New York: Wiley.

Baykasoglu, A. (2004). A meta-heuristic algorithm to solve quadratic assignment formulations of cell formation problems without presetting number of cells. *Journal of Intelligent Manufacturing, 15*(6), 753–759.

Boctor, F. (1991). A linear formulation of the machine-part cell formation problem. *International Journal of Production Research, 29*, 343–356.

Brown, E., & Sumichrast, R. (2001). CF-GGA: A grouping genetic algorithm for the cell formation problem. *International Journal of Production Research, 36*, 3651–3669.

Burbidge, J. L. (1977). A manual method for production flow analysis. *The Production Engineering, 57*, 34–38.

Carrie, A. S. (1973). Numerical taxonomy applied to group technology and plant layout. *International Journal of Production Research, 11*(4), 399–416.

Chan, H. W., & Milner, D. A. (1982). Direct clustering algorithm for group formation in cellular manufacture. *Journal of Manufacturing Systems, 1*, 65–74.

Chandrasekharan, M. P., & Rajagopalan, R. (1986). MODROC: An extension of rank order clustering for group technology. *International Journal of Production Research, 24*(5), 1221–1264.

Chandrasekharan, M. P., & Rajagopalan, R. (1986). An ideal seed non-hierarchical clustering algorithm for cellular manufacturing. *International Journal of Production Research, 24*(2), 451–464.

Chandrasekharan, M. P., & Rajagopalan, R. (1987). ZODIAC: An algorithm for concurrent formation of part-families and machine-cells. *International Journal of Production Research, 25*(6), 835–850.

Chandrasekharan, M. P., & Rajagopalan, R. (1989). Groupability: An analysis of the properties of binary data for group technology. *International Journal of Production Research, 27*(6), 1035–1052.

Cheng, C., Gupta, Y., Lee, W., & Wong, K. (1998). A TSP-based heuristic for forming machine groups and part families. *International Journal of Production Research, 36*, 1325–1337.

De Witte, I. (1980). The use of similarity coefficients in production flow analysis. *International Journal of Production Research, 18*, 503–514.

Doulabi, S. H. H., Hojabri, H., Davoudpour, H., Seyed-Alagheband, S. A., & Jaafari, A. A. (2009). Two-phase approach for solving cell-formation problem in cell manufacturing. In *Proceedings of*

the world congress on engineering and computer science, Vol II, October 20–22, 2009, San Francisco.

Gen, M., & Cheng, R. (1997). *Genetic algorithms and engineering design*. New York: Wiley.

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning*. Reading: Addison-Wesley.

Gonçalves, J. F., & Resende, M. G. C. (2004). An evolutionary algorithm for manufacturing cell formation. *Computers & Industrial Engineering, 47*, 247–273.

Guerrero, F., Lozano, S., Smith, K., Canca, D., & Kwok, T. (2002). Manufacturing cell formation using a new self-organizing neural network. *Computers, 42*, 377–382.

Islier, A. A. (2005). Group technology by an ant system algorithm. *International Journal of Production Research, 43*(5), 913–932.

Joines, J., Culbreth, C., & King, R. (1996). Manufacturing cell design: An integer programming model employing genetic algorithms. *IEE Transactions, 28*, 69–85.

Keeling, K. B., Brown, E. C., & 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.

King, J. R. (1980). Machine-component grouping in production flow analysis: An approach using a rank order clustering algorithm. *International Journal of Production Research, 18*, 213–232.

King, J. R., & Nakornchai, V. (1982). Machine component group formation in group technology: Review and extensions. *International Journal of Production Research, 20*(2), 117–133.

Kusiak, A. (1987). The generalized group technology concept. *International Journal of Production Research, 25*, 561–569.

Lozano, S., Adenso-Diaz, B., Eguia, I., & Onieva, L. (1999). A one-step tabu search algorithm for manufacturing cell design. *Journal of the Operational Research Society, 50*, 509–516.

Mak, K., Wong, Y., & Wang, X. (2000). An adaptive genetic algorithm for manufacturing cell formation. *International Journal of Advanced Manufacturing Technology, 16*, 491–497.

McCormick, W. T., Schweitzer, P. J., & White, T. W. (1972). Problem decomposition and data reorganization by a clustering technique. *Operations Research, 20*(5), 993–1009.

Ozturk, G., Ozturk, Z. K., & Islier A. A. (2006). A comparison of competitive neural network with other AI techniques in manufacturing cell formation, *Lecture Notes in Computer Science*, 575–583.

Rajagopalan, R., & Batra, J. (1975). Design of cellular production systems: A graph-theoretic approach. *International Journal of Production Research, 13*, 567–579.

Sofianopoulou, S. (1997). Application of simulated annealing to a linear model for the formation of machine cells in group technology. *International Journal of Production Research, 35*, 501–511.

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.

Srinivasan, G., & Narendran, T. (1991). GRAPHICS—A non-hierarchical clustering algorithm for group technology. *International Journal of Production Research, 29*, 463–478.

Srinivasan, G., Narendran, T. T., & Mahadevan, B. (1990). An assignment model for the part-families problem in group technology. *International Journal of Production Research, 28*(1), 145–152.

Vannelli, A., & Kumar, K. (1986). A method for finding minimal bottleneck cells for grouping part-machine families. *International Journal of Production Research, 24*, 387–400.

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

Wu, T., Low, C., & Wu, W. (2004). A tabu search approach to the cell formation problem. *International Journal OfAdvanced Manufacturing Technology, 23*, 916–924.

Wu, T. H., Chung, S. H., & Chang, C. C. (2009). Hybrid simulated annealing algorithm with mutation operator to the cell formation problem with alternative process routings. *Expert Systems with Applications, 36*, 3652–3661.

Xu, H., & Wang, H. (1989). Part family formation for GT applications based on fuzzy mathematics. *International Journal of Production Research, 27*, 1637–1651.

Zolfaghari, S., & Liang, M. (2002). Comparative study of simulated annealing, genetic algorithms and tabu search for binary and comprehensive machine-grouping problems. *International Journal of Production Research, 40*, 2141–2158.