



Solving a Cubic Cell Formation Problem with Quality Index Using a Hybrid Meta-Heuristic Approach

Burak URAZEL^{1,*} , Yeliz BURUK SAHIN² 

¹ Eskisehir Osmangazi University, Department of Electrical and Electronics Engineering, 26480, Eskisehir, TURKEY

² Eskisehir Osmangazi University, Department of Industrial Engineering, 26480, Eskisehir, TURKEY

Highlights

- The proposed mathematical model considers quality index besides void and EE.
- The performance of the methods is compared by using RPD and average RPD values.
- Superiority of the GA-SA hybrid method is proven due to especially for large-scale problems.

Article Info

Received: 01 Oct 2021
Accepted: 12 Apr 2022

Keywords

Cubic cell formation
Quality index
Hybrid GA-SA
Cellular manufacturing systems

Abstract

Although most previous studies on cell formation have involved the assignment of parts and machines to cells, in recent years the assignment of workers has also been considered and the studies have taken into account the human factor. An important step for the successful implementation of a Cellular Manufacturing System is to decide appropriate groups of parts, machines, and workers and then assign them to cells. Consideration of the skills of workers and machines in processing parts has enhanced cell performance. In this study, the problem of a cubic cell formation that takes into account the three-dimensional part-machine-worker matrix is addressed, and the minimization of the exceptional element and void as well as the maximization of the part quality index is aimed. The mathematical model used in this study was coded in the GAMS 24.2.1 software. A hybrid GA-SA approach was also proposed for the solution of large instances. The relative Percentage Deviation performance index was utilized to evaluate the performance of the algorithm. According to the results, the hybrid technique developed, considering technical cell performance criteria together with worker skills, shows promising results from the standpoint of the considered objective and the computational time.

1. INTRODUCTION

Group Technology (GT) aims to take advantage of the similarity in design and manufacturing by grouping similar parts together [1, 2]. Cellular Manufacturing System (CMS), an application of group technology in the manufacturing industry, is a widely used and successful technique in terms of shortening set-up times, in-process inventory, and requirements of the factory space as well as material routes [3]. With this technique, which tries to provide the aforementioned advantages by creating machine cells and part families, it is primarily aimed to minimize the part movement between cells [4, 5]. Essential parameters of the CMS problem are listed by Rafiee and Mohamaditalab [6]. One of the most important steps in the design of CMS is the solution of the cell formation problem. This problem has been discussed frequently in the literature [7-10]. The two basic elements that emerge in the solution of this problem are Exceptional Element (EE) and void. EE is the displacement that occurs when parts are processed on machines in separate cells. The voids inside the cell mean the absence of a link between the machine and the part [11, 12].

In cell formation literature, besides exact methods, many meta-heuristic techniques have been used. In some studies, Particle Swarm Optimization (PSO) [13, 14], Bacteria Foraging Algorithm (BFA) [15], Genetic Algorithm (GA) [11, 16], Simulated Annealing (SA) [17, 18] techniques were used. In addition, some hybrid [19] and stochastic approaches [20, 21] are also presented to solve the cell formation problem.

*Corresponding author, e-mail: yelizburuk@ogu.edu.tr

The problem of cell formation, based on the similarities of parts and machines, is critical for successful CMS design. The success of the system depends on the representation of the assumptions of real-life problems. Since the human and organizational factors are directly related to the success of the system, it is very important to identify and eliminate possible obstacles to the successful implementation of the CMS [22]. Although parts and machines have been taken into account in cell formation in most of the studies conducted, it has recently emerged that it is important to consider the human factor in these studies. The expertise and skills of the workers in machining and using machines seriously affect system quality and performance. The cell formation problem, which takes operator assignment into consideration, was first described by [23]. Important human issues in literature mainly include; the utilization of workers and machines [24-25], schedule plans [26, 27], promotion of workers from one skill to another [28-29], multi-period planning [30], workload balancing [13, 31], total non-interest workers in cells [32] and hiring, training, salary and firing costs [33].

In the literature, the three-dimensional cell formation problem, in which the workers are assigned to the cells as well as the part and the machine, is named as the Cubic Cell Formation (CCF) problem. In the CCF problem, the worker dimension is also included in the problem, taking into account technical skills related to the use of the equipment and other personal skills. Mahdavi et al. [34] included the operators in cell assignment as the third dimension, depending on their performance on the machines. They argued that the problem of assigning the operator to parts and machines is very important in order to increase the overall performance of the system. Bootaki et al. [25] aimed at maximizing the total part quality, depending on the part processing skills of the operators, as well as minimizing the intercellular movement. Mahdavi et al. [35] considered the interactional interest of workers, and the epsilon constraint method is presented to solve the developed bi-objective model. Sahin and Alpay [11] obtained successful solutions in large sizes with GA they developed for the problem of CCF, where they consider the void and EE as objectives. In the CCF problem, Bouaziz et al. [36] predicted that assigning more skilled workers from the alternative worker cluster to perform the part operation will increase the quality index of the parts to be produced. They presented a generalized CCF problem and solved it with a discrete flower pollination algorithm. Here, the factors affecting the part quality index are the machine size, machine life, age, and worker dimension, and the factors such as experience, skill, and expertise.

The structure of the CCF problem addressed by Bootaki et al. [25] was used in this study. In addition, the study of Bouaziz et al. [36] was considered to add the quality dimension to the objective function. The prominent contributions of this study can be listed as follows: (1) In addition to the main objectives of cell formation problems such as void and EE, a scalarized objective function term that aims to maximize the quality index is proposed. (2) A hybrid Genetic Algorithm-Simulated Annealing (GA-SA) is developed. (3) It is very important to consider operator capabilities for flexible and effective use of resources. In this regard, unlike previous studies, the CCF problem, which takes the quality of the parts into account, the performance of the study was tested on the data sets with different meta-heuristics, and the performance of the proposed methods was compared with the Relative Percentage Deviation (RPD).

In the following section, the problem of CCF is defined and a linearized mathematical model is explained and the proposed solution approaches are presented. The third section includes findings and discussions, and the last section gives the results and recommendations.

2. MATERIAL METHOD

In this work, while it is aimed to process similar parts and part operations in the same cell by considering minimization of voids, it is also aimed to maximize the part quality index with the minimum transfer of skilled workers between cells which are conflicting aims. The elements that make up the quality index are considered in part-machine and worker-machine dimensions. Factors related to machine life, precision, and other capabilities of the machine affect the part-machine quality. Worker-machine dimension is affected by the worker's ability and success in using the machine.

To facilitate the understanding of the problem, the movement between cells is defined for the following situations [25]:

Case 1: Part, machine, and worker are assigned to the same cell, in which case there is no movement between cells.

Case 2: While the machine and the worker are assigned to the same cell, the part has been assigned to another cell. Movement between cells is "1".

Case 3: While the part and the machine are assigned to the same cell, the worker has been assigned to a different cell and the movement between cells is "1".

Case 4: The worker and the part are assigned to a cell or cells different from the cell to which the machine is assigned, and in this case, the movement between cells is "2".

Processing of a part on a machine by a worker is shown with the three-dimensional matrix shown in Figure 1 similar to Bootaki et al. [25]. Ranking in the matrix is done over integers [1-5]. The value "0" indicates that the part cannot be machined by the relevant machine and worker. Past records and experiences are used to estimate these values. Workers' experience and skills are also important in determining these values [36].

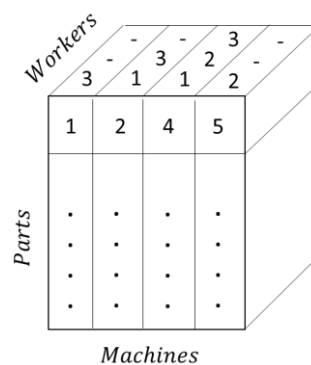


Figure 1. An example of the Part Quality matrix

2.1. Mathematical Model

The mathematical model discussed in the study of Bootaki et al. [25] is used in this study in terms of constraints. In addition, the structure similar to the quality index presented in Bouaziz et al. [36] has been taken into account in constructing the third part of the objective function. The following assumptions have been made in this model:

- The quality at which a part can be processed by a worker on a convenient machine is expressed by integers between "1" and "5".
- These above quality values are obtained through observation and based on old records.
- Workers can work on different machines (worker flexibility).
- Each job is assigned to one worker.

Indices

- i: index for parts ($i = 1, 2, \dots, P$);
- w: index for workers ($w = 1, 2, \dots, W$);
- m: index for machines ($m = 1, 2, \dots, M$);
- k: index for cells ($k = 1, 2, \dots, C$).

Parameters

- $r_{imw} = 1$, if i^{th} part can be processed on m^{th} machine by w^{th} worker; 0, otherwise;
- $a_{im} = 1$, if i^{th} part needs processing on m^{th} machine; 0, otherwise;
- q_{imw} = quality index of i^{th} part when it is processed in m^{th} machine by w^{th} worker
- N_m = number of identical machines
- LM_k = lower bound of machine assignment to cell k
- LP_k = lower bound of part assignment to cell k
- LW_k = lower bound of worker assignment to cell k
- $UQIB$ = Upper quality index bound

Decision variables

$x_{mk} = 1$, if m^{th} machine is assigned to cell k ; 0, otherwise

$y_{ik} = 1$, if i^{th} part is assigned to cell k ; 0, otherwise

$z_{wk} = 1$, if w^{th} worker is assigned to cell k ; 0, otherwise

$d_{imwk} = 1$, if i^{th} part is processed by m^{th} machine with w^{th} worker in cell k ; 0, otherwise

The objective function and constraints of 0-1 mixed integer mathematical model for the considered problem is given as follows:

$$\text{Min } Z = \alpha_1 Z_1 + \alpha_2 Z_2 + \alpha_3 Z_3 \quad (1)$$

where

$$Z_1 = \sum_{k=1}^C \left[\sum_{i=1}^P \sum_{m=1}^M \sum_{w=1}^W y_{ik} x_{mk} z_{wk} - \sum_{i=1}^P \sum_{m=1}^M \sum_{w=1}^W y_{ik} x_{mk} z_{wk} d_{imwk} \right]$$

$$Z_2 = \sum_{i=1}^P \sum_{k=1}^C \sum_{m=1}^M \sum_{w=1}^W [(2d_{imwk} x_{mk} - y_{ik} d_{imwk} x_{mk} - z_{wk} d_{imwk} x_{mk})]$$

$$Z_3 = UQIB - \sum_{i=1}^P \sum_{k=1}^C \sum_{m=1}^M \sum_{w=1}^W q_{imw} d_{i,m,w,k}$$

subject to

$$\sum_{k=1}^C z_{wk} = 1 \quad \forall w \quad (2)$$

$$\sum_{k=1}^C y_{ik} = 1 \quad \forall i \quad (3)$$

$$\sum_{k=1}^C x_{mk} \leq N_m \quad \forall m \quad (4)$$

$$\sum_{m=1}^M x_{mk} \geq LM_k \quad \forall k \quad (5)$$

$$d_{imwk} \leq r_{imw} x_{mk} \quad \forall i, m, w, k \quad (6)$$

$$\sum_{k=1}^C \sum_{w=1}^W d_{imwk} = a_{im} \quad \forall i, m \quad (7)$$

$$\sum_{i=1}^P y_{ik} \geq LP_k \quad \forall k \quad (8)$$

$$\sum_{w=1}^W z_{wk} \geq LW_k \quad \forall k \quad (9)$$

$$x_{mk}, y_{ik}, z_{wk}, d_{imwk} \in \{0,1\} \quad \forall (i, m, w, k). \quad (10)$$

In this model, Z is the objective function to be minimized and it is basically the weighted sum of three objectives: the number of total voids (Z_1), the total number of EEs (Z_2), and part quality index (Z_3). Since objective function direction of the study is minimization, a transformation in the form of “ $Z_3 =$

"*UQIB - Quality Score*" was applied to combine Z_3 with other objectives in the direction of minimization. *UQIB* here represents the greatest possible value for the selected part quality that can be assigned.

According to Equations (2) and (3) respectively, it is guaranteed that each worker and each part will be appointed to only a single cell. Equation (4) ensures that the number of identical machines to be placed in a cell does not exceed the number of machines available in that type. Equation (5) makes sure that the number of machines assigned to cell k is equal to or greater than LM_k . Equation (6) indicates whether the type m machines in cell k are used or not. Equation (7) ensures that each type of part will be processed on a proper machine by only one worker. Equation (8) identifies the minimum number of parts that should be assigned to each cell. Equation (9) specifies the minimum number of workers to be assigned to cell k . Finally, decision variables are given in Equation (10).

Here the non-linear objective function terms are linearized by using auxiliary binary variables. The weights were assumed to be equal in this study. The mathematical model was linearized similar to the studies of Mahdavi et al. [34] and Sahin and Alpay [11] as follows: Auxiliary binary variables (e.g., E_{imwk} , F_{imwk} , S_{imwk} , H_{imwk} , G_{imwk}) were used to linearize the non-linear objective-function terms that are the multiplication of the binary variables. The non-linear terms in the objective function are linearized with $E_{imwk} = x_{mk}d_{imwk}$, $F_{imwk} = x_{mk}y_{ik}z_{wk}$, $S_{imwk} = x_{mk}y_{ik}z_{wk}d_{imwk}$, $H_{imwk} = x_{mk}y_{ik}d_{imwk}$, and $G_{imwk} = x_{mk}z_{wk}d_{imwk}$. The model was linearized using the following constraints:

$$x_{mk} + y_{ik} + z_{wk} \geq 3 F_{imwk} \quad \forall i, m, w, k \quad (11)$$

$$x_{mk} + y_{ik} + z_{wk} \leq 2 + F_{imwk} \quad \forall i, m, w, k \quad (12)$$

$$x_{mk} + d_{imwk} \geq 2 E_{imwk} \quad \forall i, m, w, k \quad (13)$$

$$x_{mk} + d_{imwk} \leq 1 + E_{imwk} \quad \forall i, m, w, k \quad (14)$$

$$x_{mk} + y_{ik} + z_{wk} + d_{imwk} \geq 4 S_{imwk} \quad \forall i, m, w, k \quad (15)$$

$$x_{mk} + y_{ik} + z_{wk} + d_{imwk} \leq 3 + S_{imwk} \quad \forall i, m, w, k \quad (16)$$

$$x_{mk} + y_{ik} + d_{imwk} \geq 3 H_{imwk} \quad \forall i, m, w, k \quad (17)$$

$$x_{mk} + y_{ik} + d_{imwk} \leq 2 + H_{imwk} \quad \forall i, m, w, k \quad (18)$$

$$x_{mk} + z_{wk} + d_{imwk} \geq 3 G_{imwk} \quad \forall i, m, w, k \quad (19)$$

$$x_{mk} + z_{wk} + d_{imwk} \leq 2 + G_{imwk} \quad \forall i, m, w, k \quad (20)$$

$$E_{imwk}, F_{imwk}, S_{imwk}, H_{imwk}, G_{imwk} \in \{0,1\} \quad \forall i, m, w, k \quad (21)$$

The linearized model is described as follows:

$$\begin{aligned} \text{Min } Z = & \alpha_1 \sum_{k=1}^C \sum_{i=1}^P \sum_{m=1}^M \sum_{w=1}^W [F_{imwk} - S_{imwk}] + \alpha_2 \sum_{i=1}^P \sum_{k=1}^C \sum_{m=1}^M \sum_{w=1}^W [(2E_{imwk} - H_{imwk} - G_{imwk})] \\ & + \alpha_3 (UQIB - \sum_{i=1}^P \sum_{k=1}^C \sum_{m=1}^M \sum_{w=1}^W q_{imw} d_{i,m,w,k}) \end{aligned}$$

subject to constraints (2-21).

When the studies in the same field in the literature are examined, it is seen that the problem is too difficult to solve with the existing solvers, even with this new version. [37]. As a result, considering that the multi-objective nature of the study also increased the difficulty of the problem, it is necessary to develop metaheuristics to solve large instances.

2.2. Proposed Hybrid Method

Although GA easily provides the best or near best solutions for the global optimization problems with shorter solution times, it starts to perform poorly in avoiding local minimums when the size of the problem or the number of variables are increased [38]. To overcome this handicap, some hybrid approaches are developed that combine the superior features of different meta-heuristics. The success of these hybrid methods is proven in several problems and the results show that they can improve the solutions that have been obtained by the meta-heuristics [39].

In this study, a hybrid method that combines GA and SA is proposed. The philosophy behind this approach is first to apply single GA with the considered problem to obtain just a feasible solution, then initialize SA with this feasible solution and finally try to achieve the optimal solution by using a single SA. Since GA is applied to obtain a feasible solution instead of the optimal one and SA is initialized with a feasible solution instead of a random one, it is expected to reach the optimal solution or better solutions in shorter solution times. The pseudo-code for the proposed hybrid GA-SA method is depicted by the following algorithm.

Proposed GA-SA Algorithm

begin // Procedure GA

$g \leftarrow 0$

initialization $P_1(G)$

evaluation $P_1(G)$

while not termination criteria **do**

begin

recombination $P_1(G) \rightarrow P_2(G)$

evaluation $P_2(G)$

selection $P_1(G)$ and $P_2(G) \rightarrow P_1(G + 1)$

$g \leftarrow g + 1$

end

end

return P_{GA} : the fittest individual from $P_1(G)$

begin // Procedure SA

$s_k \leftarrow P_{GA}$ //initialize SA with the feasible solution achieved by GA

$T \leftarrow T_0$

$k \leftarrow 1$

while $T > T_{min}$ **do**

begin

generate $s_{k+1} \leftarrow s_k$ // neighborhood solution

$\Delta \leftarrow E(s_{k+1}) - E(s_k)$

if $\Delta < 0$ **then**

accept s_{k+1}

else

accept s_{k+1} considering probability $e^{\frac{-\Delta}{T}}$

decrease T

$k \leftarrow k + 1$

end

end

return P_{SA}

$P_{best} \leftarrow P_{SA}$ //The Best solution with SA is the optimal one for
//considered problem.

2.3. Genetic Algorithm

As it has been first introduced by Holland [40], GAs are population-based stochastic search techniques that are inspired by natural selection and evolution theory proposed by Charles Darwin. GAs can be applied to solve both unconstrained and constrained optimization problems in a wide range of engineering areas [41]. In GAs, potential solutions to the considered problem are regarded as individuals and encoded by simple data structures. These structures are called as chromosomes and each chromosome consists of a number of genes which has their own meaning and carry genetic information. Since these algorithms are iterative procedures performed over a population of chromosomes, first an initial population is generated. A score is appointed to each individual to show the quality of the solution. The fitness value of an individual is formed by calculating the objective function value of the corresponding individual according to the presented mathematical model. In each generation, random individuals are selected from the current population as parents and genetic operators such as crossover and mutation are applied to each parent. Afterwards, new generations are formed by selecting the individuals according to their fitness values based on survival of the fittest. In successive generations, worse solutions tend to disappear and good solutions survive. Thus these survived individuals become parents in the following generations and they result in better solutions. The evolutionary process of the GAs continues until the predetermined stopping criterion is met. Finally, the individual with the best fitness value is accepted as the best solution [42]. However, this best solution is not guaranteed to be the global optimal solution to the considered problem.

2.4. Chromosome Structure

In GAs, potential solutions to the considered problem are encoded by proper data structures or chromosomes. The chromosome structure used to express the considered problem in this manuscript, all genes have a uniformly distributed random value between 0 and 1. Each chromosome consists of two segments. In the first part of the chromosome, the machines, workers, and parts are assigned to the cells, while in the second part, the workers are assigned to the machines.

An example representation of the first segment is shown in Figure 2. This segment contains three sub-segments including 4 machines, 5 workers, and 5 parts. First, the values of each gene in the first segment should be decoded to identify which part, machine, and worker to be assigned to each cell. In a 2-cell problem, for each value that belongs to the interval $[0; 0.5)$, they are assigned to the first cell as given in Figure 3. On the other hand, for values that belong to interval $[0.5; 1)$, they are assigned to cell 2. Consequently, Figure 4 represents the decoded version of the chromosome segment. According to Figure 4, the first and fourth machines are assigned to cell 2, second and third machines to cell 1. Similarly, cell 1 contains second and third machines, third and fourth workers with first, second, and fourth parts while cell 2 contains first and fourth machines, first, second and fifth workers with third and fifth parts.

machine				worker					part				
M1	M2	M3	M4	W1	W2	W3	W4	W5	P1	P2	P3	P4	P5

Figure 2. The first segment of the chromosome

machine				worker					part				
0.6	0.4	0.2	0.8	0.8	0.5	0.4	0.2	0.5	0.2	0.4	0.6	0.3	0.7

Figure 3. An example representation of the first segment (random values)

machine				worker					part				
2	1	1	2	2	2	1	1	2	1	1	2	1	2

Figure 4. An example representation of the first segment (decoded version)

The second part of the chromosome refers to taking into account the ability of workers to use machines in the part-worker assignment. To illustrate the chromosome structure, example problem data seen in Tables 1, 2, and 3 are used. Table 1 shows the requirement of parts on different machines while Table 2 depicts the ability of workers to use machines. For example, the set of workers who can work with the parts on the

relevant machines can be seen in Table 2. The candidate workers' matrix shown in Table 3 is obtained by considering the capability of workers to process parts. According to Table 3, UQIB values are defined as follows: the maximum quality that can be achieved for the first part with the first machine is “4” with the fifth worker. As similar, this value is “5” for the first part on the third machine by the first worker. Finally, for this instance, since UQIB represents the greatest possible value for the selected part quality, it is calculated as “4+5+2+4+2+3+5=25”.

As seen in Figure 5, the number of sub-segments depends on the number of parts. Each sub-segment has a number of genes equal to the number of machines required for the corresponding part. According to Table 1, the second chromosome segment given in Figure 5 should be divided into five sub-segments. Since part 1 must be processed in machines {1, 3}; the sub-segment related to part 1 in Figure 5 has a length of 2 genes. The length of the remaining sub-segments can be easily seen in Figures 5 and decoded version of this chromosome is given in Figure 6.

Table 1. Part machine incidence matrix

Parts	Machines			
	1	2	3	4
1	1	0	1	0
2	1	0	0	0
3	0	0	0	1
4	0	1	0	0
5	0	0	1	1

Table 2. Machine-worker and part-worker incidence matrices

Machines	Workers					Workers	Parts				
	1	2	3	4	5		1	2	3	4	5
1	1	0	1	0	1	1	1	1	0	0	0
2	0	1	0	1	0	2	0	0	1	0	0
3	1	0	1	0	0	3	1	0	1	1	1
4	0	0	1	0	1	4	0	0	1	1	0
						5	1	1	0	1	1

Table 3. Candidate workers matrix with part quality indexes

Part	Machine	Worker				
		1	2	3	4	5
P1	1	2	0	2	0	4
	3	5	0	3	0	0
P2	1	1	0	0	0	2
P3	4	0	0	4	0	0
P4	2	0	0	0	2	0
P5	3	0	0	3	0	0
	4	0	0	3	0	5

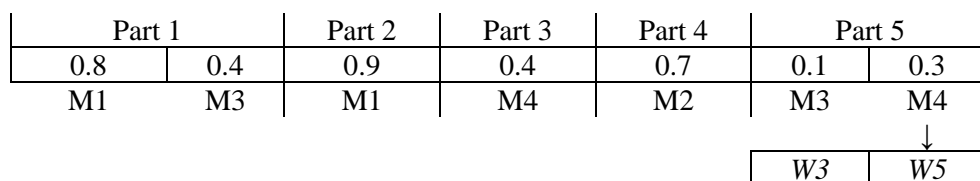


Figure 5. A representation of the second segment (random values)

According to Table 3, part 5 must be processed in machine 3 and machine 4. Besides, part 5 can be processed in the fourth machine by the workers {3, 5}. If the value of the second gene of the fifth sub-segment in Figure 5 (the gene related to a worker who processes part five in machine four) belongs to [0; 0.5), this means that the third worker will be responsible for machining the fifth part on the fourth machine. Similarly, if a random value corresponding to the interval [0.5; 1) is generated, then the fifth worker will be responsible for this process, respectively.

Part 1		Part 2	Part 3	Part 4	Part 5	
3	1	5	3	4	3	3
M1	M3	M1	M4	M2	M3	M4

W3	W5
----	----

↓

Figure 6. An example representation of the second segment (decoded version)

2.5. Fitness Evaluation and Constraint Handling

Two main methods can be used to include constraints to the model while applying GA to constrained optimization problems. Constraints can be handled by using penalty functions or by the chromosome structure itself. In this manuscript, most of the constraints in the proposed model are directly included in the solution by using the proposed chromosome structure. However, the constraints (5), (8), and (9) in the mathematical model cannot be handled by the proposed chromosome structure. These constraints are violated when there isn't any machine assigned in any cell, there isn't any worker assigned in any cell and there isn't any part assigned in any cell. Penalty functions are defined for each of them as given below:

$$CP_M = \begin{cases} pen_value_machine, & \text{if there isn't any machine assigned in any cell} \\ 0, & \text{otherwise} \end{cases}$$

$$CP_W = \begin{cases} pen_value_worker, & \text{if there isn't any worker assigned in any cell} \\ 0, & \text{otherwise} \end{cases}$$

$$CP_P = \begin{cases} pen_value_part, & \text{if there isn't any part assigned in any cell} \\ 0, & \text{otherwise} \end{cases}$$

In the above functions, the penalty values; *pen_value_machine*, *pen_value_worker*, and *pen_value_part* are determined as "50000" for all unsatisfied constraints mentioned above for the considered problem. Finally, these penalties are added to the objective function to form the fitness function. Therefore, the fitness function of the considered problem similar becomes as follows:

$$F_{fitness} = Z + CP_M + CP_W + CP_P. \quad (22)$$

2.6. Simulated Annealing

SA is another stochastic search technique which is first introduced by Kirkpatrick [43] and is too often used to solve problems with a discrete search space such as routing problems, travel salesman problems, task scheduling, etc. It has been developed based on the analogy between the way in which the crystalline structure of metal achieves near the lowest possible energy level during the annealing procedure of solids and the way in which a function may reach a minimum during a stochastic search of the solution space [44]. In this context, the states of the solid represent possible solutions to the optimization problem while the energy levels of each state correspond to the objective function values of the related solution. Therefore, the lowest energy level represents the optimal solution for the problem [45]. SA is a kind of iterative process and at each iteration new random solutions are generated for the current solutions. These new solutions are also called as neighborhood solutions. At the end of each iteration, all neighborhood solutions that lower the objective function value of the current solution are accepted. On the other hand, the neighborhood solutions that raise the objective function can be also accepted as a new solution according to the predefined

acceptance probability function. Since some worse new solutions are also accepted with a certain probability, SA performs successfully in avoiding to stuck at local minimums. These iterations are repeated until predetermined stopping criteria are met.

2.7. Deriving Neighborhood Solutions

In SA, it is tried to achieve the optimal solution by means of the neighborhood solutions derived from the current solution in each iteration. Therefore, the mechanism behind generating neighborhood solutions and the quality of these have an important role on the performance of the algorithm.

In this manuscript, SA is applied by expressing the problem with the chromosome structure mentioned given in Figure 6 which can be also considered as the decoded version of the main chromosome that consists of real numbers between [0;1]. By considering this decoded chromosome structure, a neighborhood solution is generated by modifying the value of a randomly selected gene.

If the selected gene takes place in the first segment, then it means that the assignments of machines, workers, or parts to cells will be modified. In this case, the value of the selected gene is replaced with remaining cells one by one in random order. For a 4-cell problem, consider the chromosome structure whose first segment is given in Figure 7 (a). Assume that the 6th gene is randomly selected to generate a neighborhood solution. Therefore, the value of the 6th gene will be replaced by the elements of the list which includes the remaining cells in a random order as given in Figure 7 (b). After each replacement, problem constraints are checked whether this newly generated solution is feasible or not. If the generated solution is feasible, then it is accepted as the neighborhood solution. Otherwise, the replacement procedure continues until a feasible solution is obtained. If there isn't any feasible solution is achieved after all cell values are tested, then the current solution is considered as the neighborhood solution. An example of a generated neighborhood solution is given in Figure 7 (c).

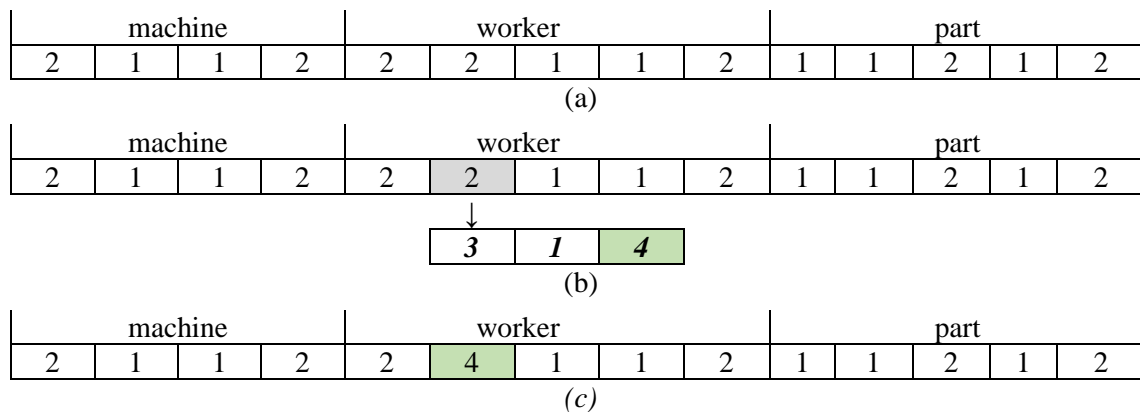


Figure 7. Example representation of chromosomes a) first segment of the chromosome b) modifying the randomly selected gene c) generated neighborhood solution

Since the gene to be modified is selected randomly, it is also possible that it can be located in the second segment and differences may occur in the worker-part assignment, depending on the ability to use the machines. For a 4-cell problem, consider the chromosome structure whose second segment is given in Figure 8 (a). This section shows the relationship between the assignment of workers and related machines. Assume that 1st gene of this segment is randomly selected to generate a neighborhood solution. Therefore, the selected gene is replaced by the elements of the list which includes the remaining workers in a random order who are able to use this machine as given in Figure 8 (b). According to Table 3, the first part in the first machine can be processed by workers {1, 3, 5}. So, the list includes workers 1 and 5.

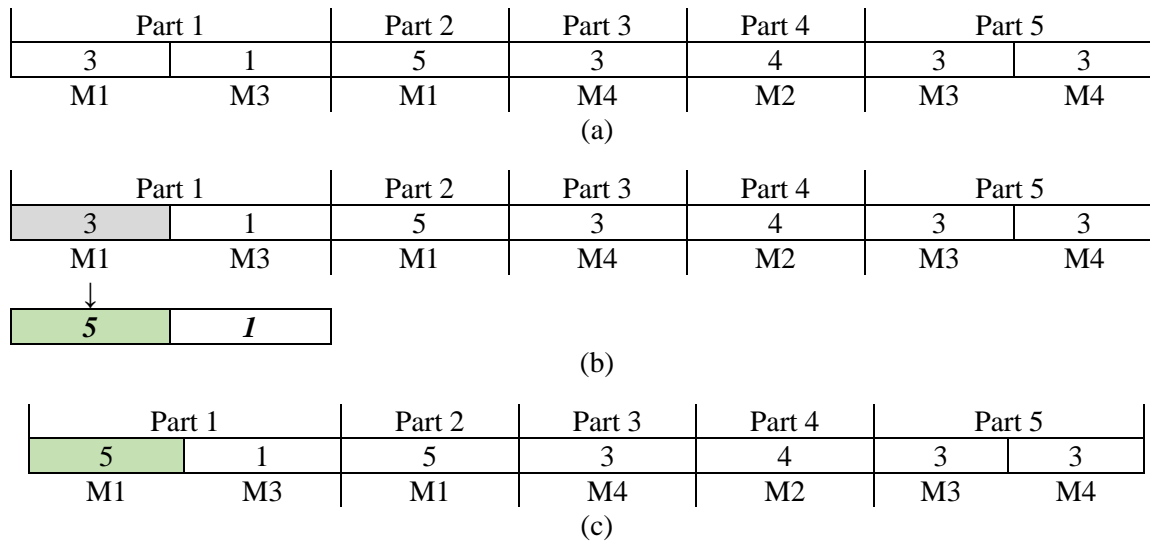


Figure 8. An example representation of chromosomes a) The Second segment of the chromosome b) Modifying the randomly selected gene c) Generated neighborhood solution

After each replacement, obtained interim solution is checked whether it is feasible or not. If this interim solution is feasible, then it is accepted as the neighborhood solution. Otherwise, the replacement procedure continues until a feasible solution is obtained. If there isn't any feasible solution is reached after all available workers are tested, then the current solution is considered as the neighborhood solution. An example of a generated neighborhood solution is given in Figure 8 (c).

3. RESEARCH FINDINGS AND DISCUSSIONS

The data set used in the study was first proposed by Mahdavi et al. [34]. In addition, due to the nature of the problem, the quality matrix has been added in this work and the data set is presented in Appendix A. In the analysis, the GAMS 24.2.1 software was run in a limited time (10800 sec) and the results were recorded. Although the best solutions for the small size problems are achieved with GAMS 24.2.1 by using the proposed model, it fails to obtain proven optimal solutions as the model size has increased. Therefore, a hybrid GA-SA method is coded in MATLAB R2013a software and applied to solve large-scale problems on a PC with an Intel (R) Core (TM) 2 duo CPU 2.53 GHz processor and 4 GB of RAM.

3.1. Parameter Setting and Stopping Criterion

It is obvious that the solution quality and the performance of metaheuristics largely depend on the choice of parameters. In this study, the parameters given in Sahin and Alpay [11] are used by considering the levels obtained by the Taguchi method. Since in this study GA is applied to achieve an initial solution point for SA instead of the optimal solution, population size and number of generations are taken lower than the ones given in Sahin and Alpay [11]. Roulette selection technique was used and "adaptive feasible mutation" in MATLAB R2013a [46] was selected as the mutation function. Besides, the initial temperature value used in SA was specified according to the problem sizes and taken as 500 for small size problems and 750 for large size problems. At the end of each outer iteration, the temperature value is updated by using standard geometric cooling scheduling as $T_{new} = 0.9 \times T_{old}$. For each temperature value, the number of generated neighborhood solutions is selected as 100 for small size problems and 150 for large size problems. For all test problems, SA terminates when the temperature value becomes lower than 0.5. Selected parameters are given in Table 4.

Table 4. Parameters for Hybrid GA-SA

Parameters	Small-scale	Large-scale
Population size	50	150
Max. Generation	75	150
Crossover Ratio	0.9	0.9
Mutation Ratio	0.1	0.1
Initial Temperature	500	750
Minimum Temperature	0.5	0.5
Number of Iteration	100	150

3.2. Comparison of the Results

The aforementioned eleven test problems were solved by GAMS, GA, and GA-SA hybrid techniques. First of all, problems were solved with 25 repetitions by the chosen GA parameters, the best, mean, and worst values and the mean of solution time were recorded for each problem. For the GA-SA hybrid method, the best solution of the repeated 25 trials obtained with the best parameter levels of GA was assumed as the initial solution in SA. Afterwards, solution results and solution times on average of 25 replicates were recorded. The results obtained are included in Table 5. Since all constraints are satisfied and the function does not receive a penalty, $F_{fitness}$ and z_3 are equal to each other in Table 5 (see Equation (11)).

Table 5. Comparison of mathematical model, GA and GA-SA result

Size	Test Problem (part, machine, worker, cell)	GAMS		Proposed GA				Proposed GA-SA			
		Solution	Elapsed Time (s)	SolBest	SolAvg	SolWorst	Avg. Elapsed Time (s)	SolBest	SolAvg	SolWorst	Avg. Elapsed Time (s)
Small	p1 (4,4,4,2)	12*	< 1	12	12.24	13	3.63	12	12.08	13	3.20
	p2 (5,4,5,2)	15*	< 1	15	16.04	17	3.38	15	15.80	17	3.11
	p3 (6,5,5,2)	20*	< 1	20	20.36	22	4.39	20	20.72	24	3.81
	p4 (10,7,4,2)	34*	< 1	34	35.00	49	10.16	34	35.24	47	9.27
	p5 (10,7,6,3)	29*	333	29	32.00	37	5.44	29	30.68	37	5.28
Large	p1 (12,8,6,3)	32*	367	32	35.76	43	19.62	32	34.12	41	17.39
	p2 (12,8,7,3)	35**	> 10800	34	38.44	45	19.38	34	37.40	43	19.38
	p3 (15,10,6,3)	51**	> 10800	51	59.00	64	27.78	51	55.64	63	28.88
	p4 (15,10,6,4)	43**	> 10800	43	47.48	53	26.42	42	45.20	52	22.74
	p5 (20,10,6,3)	68**	> 10800	63	70.20	80	37.38	62	69.96	78	43.07
	p6 (20,10,6,4)	58**	> 10800	54	61.72	77	34.12	51	57.48	64	36.96

* Global optimum ** Best integer found after 5 hours

The best solution was found with GAMS in a limited time for all of the small problems and for the first of the large problems. For these problems, the best solutions were found in a short time with the GA and GA-SA approaches. This situation can be considered as an indication that the proposed meta-heuristics are working. It was seen that as the size of the problem increased, the best results that could be obtained with the GA-SA technique were better. Besides, the hybrid method seems to be more successful than the GA technique in terms of average elapsed time.

In addition, the performances for the proposed methods were compared with the RPD criterion [47]. As listed in Table 6, RPDs and mean RPDs (ARPDs) were calculated by Equations (23) - (24) with repeated test results

$$RPD = \frac{|Algorithm_{Sol} - Best_{Sol}|}{Best_{Sol}} \times 100 \quad (23)$$

$$ARPD = \overline{RPD} = \frac{\sum_{i=1}^{number\ of\ run} RPD_i}{number\ of\ run} \quad (24)$$

While the solution of each problem obtained by the algorithm was expressed as $Algorithm_{sol}$, the best solution of the problem among the trials was expressed as $Best_{sol}$. In order to compare the performance of the methods, RPD and \overline{RPD} values were calculated in all problem dimensions. In addition, the arithmetic mean of the mean of \overline{RPD} values based on the problem dimensions was also examined. Table 6 shows the \overline{RPD} values for each problem in the framework of methods.

Table 6. Comparison of proposed methods according to \overline{RPD} values

	Solution Techniques	
	GA	GA-SA
Small Instances	\overline{RPD}	\overline{RPD}
P1	2.00	0.67
P2	6.93	5.33
P3	1.80	3.60
P4	2.94	3.65
P5	10.34	5.79
Average	4.80	3.81
Large Instances		
P1	11.75	6.63
P2	13.06	10.00
P3	15.69	9.09
P4	13.05	7.62
P5	13.23	12.84
P6	21.02	12.71
Average	14.63	9.82

According to Table 6, GA has reached a solution with an average deviation of 4.80% for small size problems and 14.63% for large size problems from the best solutions reached. On the other hand, in the GA-SA hybrid method, an average deviation of 3.81% for small size problems and 9.82% for large size problems was obtained. As a result, the lowest average \overline{RPD} values were obtained for the GA-SA hybrid method. The results showed that the proposed GA-SA hybrid method was more successful in convergence to the best solution in iterative problems of different dimensions.

4. CONCLUSIONS

Considering the importance of the ability of the workers in using the machines and processing the part, the worker has been included in the studies as a dimension in addition to the part and machine size in recent years. This study is important in terms of considering the worker as a dimension in the CMS. Part quality is included in the problem objectives as well as void and EE, and it is important for improving the quality of the produced parts and customer satisfaction. GA and GA-SA meta-heuristics, which were also developed in the RPD indicator, proved the success of large-scale.

For the small-scale problems, GAMS gives the proven optimal results for the considered problem in a reasonable time period and easily outperforms both GA and GA-SA in terms of average solution time. It is obvious that the mathematical model of the considered problem is successfully represented by the proposed chromosome structure and the proposed constraint handling mechanism. Besides, the proposed procedure for generating neighborhood solutions helps SA to search the solution space more efficiently. When the problem size is increased, GAMS starts to have difficulties obtaining the best solutions, and the solution times are considerably increased. For these large-scale problems, GA and proposed GA-SA are successfully performed and they both give better solutions in shorter durations compared to GAMS.

The lowest average \overline{RPD} values were obtained for the GA-SA hybrid method. Neither GA nor GA-SA becomes dominant in terms of RPD values for small-scale problems and they tend to have similar average RPD values. But GA-SA starts to outperform GA with having lower RPD values for each large-scale

problem. Lower RPD values for GA-SA is an indication of the success of the algorithm to find the best or best possible in different runs.

This study reveals also many avenues for future work. It may be considered to integrate issues closely related to human factors such as teamwork, synergy, competition, and incentives, into models. The psychological factors of the operators have not been taken into account in CMS studies due to the difficulty of being included in the problem. Environmental factors are also disregarded. To consider the environmental aspect in cell formation problem, minimization of different kinds of wastes such as CO₂ emission, energy loss, raw material scrap, etc. can be considered in future cubic cell work. In particular, the inclusion of ergonomic factors that may have an impact on work performance is also open to investigation. Additionally, noise dosage to prevent worker hear loss can be considered as a social constraint to minimize and balance noise exposure that may cause social and physical problems. The validity and success of the model will increase by including the mentioned situations.

CONFLICTS OF INTEREST

No conflict of interest was declared by the authors.

REFERENCES

- [1] Danilovic, M., Ilic, O., "A novel hybrid algorithm for manufacturing cell formation problem", *Expert Systems with Applications*, 135: 327-350, (2019).
- [2] Li, M., "A novel algorithm of cell formation with alternative machines and multiple-operation-type machines", *Computers & Industrial Engineering*, 154: 107172, (2021).
- [3] Rafiee, M., Kayvanfar, V., Mohammadi, A., Werner, F., "A robust optimization approach for a cellular manufacturing system considering skill-leveled operators and multi-functional machines", *Applied Mathematical Modelling*, 107: 379-397, (2022).
- [4] Ameli, M. S. J., Arkat, J., "Cell formation with alternative process routings and machine reliability consideration", *The International Journal of Advanced Manufacturing Technology*, 35: 761-768, (2008).
- [5] Dalfard, V.M., "New mathematical model for problem of dynamic cell formation based on number and average length of intra and intercellular movements", *Applied Mathematical Modelling*, 37(4): 1884-1896, (2013).
- [6] Rafiee, M., Mohamaditalab, A., "Investigation into skill leveled operators in a multi-period cellular manufacturing system with the existence of multi-functional machines", *Scientia Iranica*, 27(6): 3219-3232, (2020).
- [7] Brown, J. R., "A capacity constrained mathematical programming model for cellular manufacturing with exceptional elements", *Journal of Manufacturing Systems*, 37: 227-232, (2015).
- [8] Bychkov, I., Batsyn, M., "An efficient exact model for the cell formation problem with a variable number of production cells", *Computers & Operations Research*, 91: 112-120, (2018).
- [9] Sahin, Y. B., Alpay, S., "A New Mathematical Model for the Integrated Solution of Cell Formation and Part Scheduling Problem", *Gazi University Journal of Science*, 32(4): 1196-1210, (2019).
- [10] Danilovic, M., Ilic, O., "A novel hybrid algorithm for manufacturing cell formation problem", *Expert Systems with Applications*, 135: 327-350, (2019).

- [11] Sahin, Y. B., Alpay, S., "A metaheuristic approach for a cubic cell formation problem", *Expert Systems with Applications*, 65: 40-51, (2016).
- [12] Firouzian, S., Mahdavi, I., Paydar, M. M., Saadat, M., "Simulated annealing and artificial immune system algorithms for cell formation with part family clustering", *Journal of Industrial Engineering and Management*, 7(1): 191-219, (2020).
- [13] Feng, H., Da, W., Xi, L., Pan, E., Xia, T., "Solving the integrated cell formation and worker assignment problem using particle swarm optimization and linear programming", *Computers & Industrial Engineering*, 110: 126-137, (2017).
- [14] Mahmoodian, V., Jabbarzadeh, A., Rezazadeh, H., Barzinpour, F., "A novel intelligent particle swarm optimization algorithm for solving cell formation problem", *Neural Computing and Applications*, 31(2): 801-815, (2019).
- [15] Liu, C., Wang, J., Leung, J. Y. T., Li, K., "Solving cell formation and task scheduling in cellular manufacturing system by discrete bacteria foraging algorithm", *International Journal of Production Research*, 54(3): 923-944, (2016).
- [16] Hazarika, M., Laha, D., "Genetic algorithm approach for machine cell formation with alternative routings", *Materials Today*, 5(1): 1766-1775, (2018).
- [17] Adinarayanan, A., Uthayakumar, M., Prabhakaran, G., "Machine cell formation using simulated annealing algorithm in cellular manufacturing system", *International Journal of Computer Aided Engineering and Technology*, 10(1-2): 111-125, (2018).
- [18] Kamalakannan, R., Pandian, R. S., Sivakumar, P., "A simulated annealing for the cell formation problem with ratio level data", *International Journal of Enterprise Network Management*, 10(1): 78-90, (2019).
- [19] Deep, K., "Machine cell formation for dynamic part population considering part operation trade-off and worker assignment using simulated annealing-based genetic algorithm", *European Journal of Industrial Engineering*, 14(2): 189-216, (2020).
- [20] Ariafar, S., Firoozi, Z., Ismail, N., A triangular stochastic facility layout problem in a cellular manufacturing system. In *international conference on Mathematical Sciences and Statistics*, Springer, Singapore, 45-52, (2013).
- [21] Zohrevand, A. M., Rafiei, H., Zohrevand, A. H., "Multi-objective dynamic cell formation problem: A stochastic programming approach", *Computers & Industrial Engineering*, 98: 323-332, (2016).
- [22] Islier, A., "Cellular Manufacturing Systems: Organization, Trends and Innovative Methods", *Alphanumeric Journal*, 3(2): 13-26, (2015).
- [23] Min, H., Shin, D., "Simultaneous formation of machine and human cells in group technology: A multiple objective approach", *International Journal of Production Research*, 31(10): 2307-2318, (1993).
- [24] Bootaki, B., Mahdavi, I., Paydar, M. M., "A hybrid GA-AUGMECON method to solve a cubic cell formation problem considering different worker skills", *Computers & Industrial Engineering*, 75: 31-40, (2014).

- [25] Bootaki, B., Mahdavi, I., Paydar, M. M., “New bi-objective robust design-based utilisation towards dynamic cell formation problem with fuzzy random demands”, *International Journal of Computer Integrated Manufacturing*, 28(6): 577-592, (2015).
- [26] Delgoshaei, A., Ariffin, M. K. A., Ali, A., “A multi-period scheduling method for trading-off between skilled-workers allocation and outsource service usage in dynamic CMS”, *International Journal of Production Research*, 55(4), 997-1039 (2017).
- [27] Goli, A., Tirkolaee, E. B., Aydın, N. S., “Fuzzy integrated cell formation and production scheduling considering automated guided vehicles and human factors”, *IEEE Transactions on Fuzzy Systems*, 29(12), 3686-3695 (2021).
- [28] Delgoshaei, A., Ali, A., “An applicable method for scheduling temporary and skilled-workers in dynamic cellular manufacturing systems using hybrid ant colony optimization and tabu search algorithms”, *Journal of Industrial and Production Engineering*, 34(6): 425-449, (2017).
- [29] Delgoshaei, A., Mirzazadeh, A., Ali, A., “A hybrid ant colony system and tabu search algorithm for the production planning of dynamic cellular manufacturing systems while confronting uncertain costs”, *Brazilian Journal of Operations & Production Management*, 15(4): 499-516, (2018).
- [30] Mahdavi, I., Aalaei, A., Paydar, M. M., Solimanpur, M., “Multi-objective cell formation and production planning in dynamic virtual cellular manufacturing systems”, *International Journal of Production Research*, 49(21), 6517-6537, (2011).
- [31] Nouri, H., “Development of a comprehensive model and BFO algorithm for a dynamic cellular manufacturing system”, *Applied Mathematical Modelling*, 40(2): 1514-1531, (2016).
- [32] Bagheri, F., Safaei, A. S., Kermanshahi, M., Paydar, M. M., “Robust Design of Dynamic Cell Formation Problem Considering the Workers Interest”, *International Journal of Engineering*, 32(12): 1790-1797, (2019).
- [33] Niakan, F., Baboli, A., Moyaux, T., Botta-Genoulaz, V., “A bi-objective model in sustainable dynamic cell formation problem with skill-based worker assignment”, *Journal of Manufacturing Systems*, 38: 46-62, (2016).
- [34] Mahdavi, I., Aalaei, A., Paydar, M. M., Solimanpur, M., “A new mathematical model for integrating all incidence matrices in multi-dimensional cellular manufacturing system”, *Journal of Manufacturing Systems*, 31(2): 214-223, (2012).
- [35] Mahdavi, I., Bootaki, B., Paydar, M. M., “Manufacturing Cell Configuration Considering Worker Interest Concept Applying a Bi-Objective Programming Approach”, *International Journal of Industrial Engineering & Production Research*, 25(1): 41-53, (2014).
- [36] Bouaziz, H., Berghida, M., Lemouari, A., “Solving the generalized cubic cell formation problem using discrete flower pollination algorithm”, *Expert Systems with Applications*, 113345, (2020).
- [37] Delgoshaei, A., Gomes, C., “A multi-layer perceptron for scheduling cellular manufacturing systems in the presence of unreliable machines and uncertain cost”, *Applied Soft Computing*, 49, 27-55, (2016).
- [38] Chen, A., Jiang, T., Chen, Z., Zhang, Y., “A genetic and simulated annealing combined algorithm for optimization of wideband antenna matching networks”, *International Journal of Antennas and Propagation*, 251624, (2012).

- [39] Delgoshaei, A., Ali, A., Ariffin, M. K. A., Gomes, C., “A multi-period scheduling of dynamic cellular manufacturing systems in the presence of cost uncertainty”, *Computers & Industrial Engineering*, 100, 110-132, (2016).
- [40] Holland, J. H., *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*, 1st ed., U Michagan Press, Oxford, England, (1992).
- [41] Gen, M., Cheng, R., *Genetic Algorithms and Engineering Optimization*, 1st ed., John Wiley and Sons, New York, 53-61, (2000).
- [42] Rabunal, J. R., Dorado, J., *Artificial neural networks in real-life applications*, 1st ed., IGI Global, United Kingdom, London, 105-112, (2006).
- [43] Kirkpatrick, S., Gelatt, C. D., Vecchi, M. P., “Optimization by simulated annealing”, *Science*, 220(4598): 671-680, (1983).
- [44] Sahab, M., Toropov, V., Gandomi, A., *A Review on Traditional and Modern Structural Optimization* 1st ed., Waltham, USA, 25-47, (2013).
- [45] Eglese, R. W., “Simulated annealing: a tool for operational research”, *European Journal of Operational Research*, 46(3): 271-281, (1990).
- [46] <https://www.mathworks.com/help/gads/genetic-algorithm-options.html>. Access date: 15.10.2021
- [47] Attar, S. F., Mohammadi, M., Tavakkoli-Moghaddam, R., “Hybrid flexible flowshop scheduling problem with unrelated parallel machines and limited waiting times”, *International Journal of Advanced Manufacturing Technology*, 68(5-8): 1583-1599, (2013).

Appendix A. Quality Index Data Values**Small Problems****p1 (4,4,4,2)**

part. machine	workers			
	1	2	3	4
1.2	2	4	4	3
1.3	5	3	1	2
1.4	0	2	0	0
2.1	1	3	0	1
2.3	1	2	3	3
3.1	0	2	0	5
3.2	0	1	2	2
3.3	0	1	3	5
3.4	0	2	0	0
4.1	1	0	0	3
4.3	4	0	2	2

p2 (5,4,5,2)

part. machine	workers				
	1	2	3	4	5
1.1	2	0	2	0	4
1.3	5	0	3	0	0
2.1	1	0	0	0	2
3.4	0	0	4	0	0
4.2	0	0	0	2	0
5.3	0	0	3	0	0
5.4	0	0	3	0	5

p3 (6,5,5,2)

part. machine	workers				
	1	2	3	4	5
1.1	0	3	0	5	0
1.3	0	0	0	2	0
1.4	0	4	0	0	0
2.1	0	2	0	5	0
2.4	0	1	0	0	0
3.2	2	0	0	0	4
3.5	1	0	3	0	3
4.2	0	0	0	0	2
5.3	0	0	0	3	0
5.4	0	5	0	0	0
6.1	0	4	0	1	0
6.3	0	0	0	3	0

p4 (10,7,4,2)

part. machine	workers			
	1	2	3	4
1.1	3	0	0	5
1.3	0	3	0	0
1.5	0	0	0	2
1.6	5	2	0	0
1.7	0	0	0	3
2.2	0	0	3	0
2.4	0	4	5	0
2.7	0	0	0	2
3.3	0	3	0	0
3.4	0	3	0	0
3.6	0	1	0	0
4.2	0	0	5	0
4.3	0	3	0	0
4.4	3	1	4	0
5.1	0	0	0	2
5.4	0	5	0	0
5.7	0	0	0	3
6.1	1	0	0	3
6.4	2	0	0	0
6.6	3	0	0	0
6.7	0	0	0	1
7.1	0	0	0	2
7.2	0	0	2	0
7.3	0	2	0	0
8.5	0	0	3	5
8.6	1	4	0	0

p5 (10,7,6,3)

part. machine	workers					
	1	2	3	4	5	6
1.2	2	0	0	0	0	4
1.5	0	0	0	0	0	3
2.1	0	2	0	0	5	0
2.6	0	3	0	0	0	0
3.3	0	0	2	3	0	0
3.4	0	0	4	0	0	0
4.4	0	0	3	0	0	0
5.5	0	0	0	0	0	4
5.7	2	0	0	0	0	0
6.6	0	2	0	0	0	0
7.1	0	0	0	0	5	0
7.7	2	0	0	0	0	0
8.1	0	2	0	0	0	0
9.3	0	0	0	3	0	0
10.2	2	0	0	0	0	5
10.5	0	0	0	0	0	4
10.7	2	0	0	0	0	0

Large Problems**p1 (12,8,6,3)**

part. machine	workers					
	1	2	3	4	5	6
1.1	0	2	0	5	0	0
1.4	0	1	0	0	0	0
1.5	0	0	0	2	0	0
2.2	1	0	0	0	3	0
2.3	0	0	0	0	3	0
3.1	0	2	0	3	0	0
3.4	0	4	0	0	0	0
4.3	0	0	4	1	0	0
4.5	0	0	3	2	0	0
4.8	0	0	1	5	0	0
5.6	1	0	0	0	0	0
5.7	0	0	0	0	3	0
6.2	3	0	0	0	0	0
6.4	0	4	0	0	0	0
7.1	0	0	0	1	0	0
8.3	0	0	2	0	0	0
8.4	0	1	0	0	0	0
9.8	0	0	0	2	4	0
10.5	0	0	1	0	0	0
11.1	0	0	0	1	0	0
11.3	0	0	0	3	5	0
12.4	0	3	0	0	0	0

p2 (12,8,7,3)

part. machine	workers						
	1	2	3	4	5	6	7
1.1	0	3	0	0	4	0	0
1.5	0	0	0	0	0	0	2
1.7	0	1	0	0	0	0	3
2.2	0	3	0	0	0	5	0
2.8	0	0	0	2	0	4	0
3.4	0	0	0	0	0	2	3
3.5	0	0	0	0	0	0	5
4.3	3	0	0	4	0	0	0
5.6	0	0	2	0	5	0	0
5.7	0	0	3	0	0	0	0
6.2	0	1	0	0	0	4	0
6.4	0	0	0	0	0	2	0
7.1	0	0	2	0	3	0	0
8.2	0	4	0	0	0	2	0
8.5	0	0	0	0	0	0	2
9.1	0	1	3	0	5	0	0
9.6	0	0	4	0	2	0	0
10.4	0	0	0	0	0	5	0
11.1	0	0	2	0	3	0	0
12.7	0	1	0	0	0	0	3

p3 (15,10,6,3)/ p4 (15,10,6,4)

part. machine	workers					
	1	2	3	4	5	6
1.1	0	3	0	4	0	0
1.4	0	2	0	0	0	0
1.5	0	0	0	3	0	0
2.2	1	0	0	0	4	0
2.3	0	0	0	0	2	0
3.1	0	1	0	5	0	0
3.4	0	2	0	0	0	0
4.3	0	0	3	5	0	0
4.5	0	0	2	1	0	0
4.8	0	0	1	3	0	0
5.6	1	0	0	0	0	0
5.7	0	0	0	0	2	0
6.2	2	0	0	0	0	0
6.4	0	3	0	0	0	0
7.1	0	2	0	4	0	0
7.9	0	1	0	0	0	0
8.3	0	0	5	0	0	0
8.4	0	4	0	0	0	0
9.2	5	0	0	0	3	0
9.8	0	0	0	4	2	0
9.9	0	0	0	0	4	0
10.3	0	0	1	0	0	0
10.10	0	0	2	0	0	0
11.1	0	0	0	4	0	0
11.3	0	0	0	2	4	0
12.4	0	3	0	0	0	0
12.10	0	0	0	0	0	2
13.6	2	0	0	0	0	0
13.7	0	0	0	0	1	0
14.1	0	2	0	0	0	0
15.3	0	0	3	5	0	0

Large Problems (cont.)**p5(20,10,6,3)/ p6 (20,10,6,4)**

part. machine	workers					
	1	2	3	4	5	6
1.1	0	3	0	4	0	0
1.4	0	2	0	0	0	0
1.5	0	0	0	3	0	0
2.2	1	0	0	0	4	0
2.3	0	0	0	0	2	0
3.1	0	1	0	5	0	0
3.4	0	2	0	0	0	0
4.3	0	0	3	5	0	0
4.5	0	0	2	1	0	0
4.8	0	0	1	3	0	0
5.6	1	0	0	0	0	0
5.7	0	0	0	0	2	0
6.2	2	0	0	0	0	0
6.4	0	3	0	0	0	0
7.1	0	2	0	4	0	0
7.9	0	1	0	0	0	0
8.3	0	0	2	0	0	0
8.4	0	4	0	0	0	0
9.2	0	0	0	0	3	0
9.8	0	0	0	4	2	0
9.9	0	0	0	0	4	0
10.3	0	0	1	0	0	0
10.10	0	0	2	0	0	0
11.1	0	0	0	4	0	0
11.3	0	0	0	2	4	0
12.4	0	3	0	0	0	0
12.10	0	0	0	0	0	2
13.6	2	0	0	0	0	0
13.7	0	0	0	0	1	0
14.1	0	2	0	0	0	0
15.3	0	0	3	5	0	0
16.5	0	0	2	4	0	0
16.6	2	0	0	0	0	0
17.1	0	3	0	4	0	0
18.3	0	0	2	0	0	0
18.4	0	1	0	0	0	0
19.8	0	0	4	0	0	0
19.10	0	0	1	0	0	5
20.2	3	0	0	0	5	0