International Journal of Industrial Engineering & Production Research (2018)

March 2018, Volume 29, Number 1
pp. 53- 63

DOI: 10.22068/ijiepr.29.1.53 http://IJIEPR.iust.ac.ir/

Study of Buffer Effects on the Grouping Efficacy Measure of Stochastic Cell Formation Problem

Bahman Esmailnezhad & Mohammad Saidi-Mehrabad*

Bahman Esmailnezhad, *Ph.D. Candidate, Department of Industrial Engineering, Iran University of Science and Technology*
Mohammad Saidi-Mehrabad, *Professor, Department of Industrial Engineering, Iran University of Science and Technology*

KEYWORDS

Stochastic cell formation problem,
Grouping efficacy measure,
Metaheuristic.

ABSTRACT

This paper deals the stochastic cell formation problem (SCFP) and presents a new nonlinear integer programming model for the SCFP in which the effect of buffer size on the grouping efficacy of cells has been investigated. The objective function is the maximization of the grouping efficacy of cells. A chance constraint is applied to explore the effect of buffer on the SCFP. Processing time and arrival time of the part for each cell are considered stochastic which follow exponential probability distribution. To find the optimal solution in a reasonable time, a heuristic approach is used to linearize the proposed nonlinear model. This problem has been known as an NP-hard problem. Therefore, two metaheuristic methods, namely genetic algorithm and particle swarm optimization, are employed to solve examples. The parameters of the algorithms are calibrated using Taguchi and full factorial methods, and the performances of the algorithms in the examples of various sizes are analyzed against global solutions obtained from Lingo software's branch and bound (B&B) in terms of quality of solutions and computational time.

1. Introduction

Cellular manufacturing (CM) is considered as the application of group technology (GT) philosophy and its principle, which focuses on the identification of similar parts to the benefit of a particular production. It offers promising alternative solutions to manufacturing systems. The CM approach uses both the job and mass production. The main enhancement of cellular manufacturing implementation incorporates reduction in set-up time, throughput time, material handling, and improved quality management. One of the basic problems that has to be solved before implementing CM is the cell formation (CF) problem. The objective of the CF is the establishment of the family of parts and the group of machines for subsequent processes [1]. Addressing the CF in the uncertainty

* **Corresponding author:** ***Mohammad Saidi-Mehrabad***
Email: mehrabad@iust.ac.ir
Received 26 September 2017; revised 4 December 2017; accepted 22 January 2018

conditions is closer to reality because demand, machine availability, processing time, setup time, etc. are uncertain in the real world. In this research, processing time is stochastic. Study of buffer effects has not been ever considered in the SCFP; hence, buffer effects will be discussed in this research. Whenever there is variation, inventory is needed to compensate for the variation if we wish to maintain the production rate. This is not quite a true statement. Specifically, when there is variation, this variation needs to be compensated in order to a maintain rate. Inventory is discussed as a countermeasure for variation; however, in a more general sense, a buffer is required. A buffer is the excess resource designed to account for the fact that production cannot be in perfect lock-step with consumption [2]. In the following, a review of new researches is presented about the SCFP.

Egilmez et al. [3] addressed the impact of probabilistic demand and processing times on cell formation. The objective function was the maximization of the total similarity among products that are formed as families to be produced in dedicated cells, while minimizing the total number of cells. Rabbani et al. [4] proposed a bi-objective cell formation problem with demand of products expressed in a number of probabilistic scenarios. To deal with the uncertain demand of products, a framework of a two-stage stochastic programming model was presented. The first crucial objective function minimized the machine constant cost, expected machine variable cost, cell fixed-charge cost, and expected inter-cell movement cost; the second crucial objective function minimized the expected total cell loading variation. Egilmez et al. [5] proposed a non-linear mathematical model to solve stochastic capacitated CM system. The problem was observed in both machine and labor-intensive cells, where operation times were probabilistic in addition to uncertain customer demand. They assumed that processing time and customer demand are normally distributed. Their objective was to design a CM system with product families formed with the most similar products and minimum number of cells and machines for a specified risk level. Aghajani et al. [6] presented a dynamic multi-objective mixed integer mathematical model for a cell formation problem with probabilistic demand and machine reliability analysis. Their objective functions included the minimization of the system production costs for meeting the demand, machine underutilization cost, and system failure rate simultaneously. The objective function of system production costs included machine operating, internal part production, intercellular material handling, and subcontracting costs. Egilmez et al. [7] studied a stochastic skill-based manpower allocation problem, where operation times and customer demand were uncertain. Their proposed methodology optimized the manpower levels, product-cell formations, and individual worker assignment hierarchically with respect to a specified risk level. They developed three stochastic nonlinear mathematical models to deal with manpower level determination, cell loading, and individual worker assignment phases. Salarian et al. [8] developed a mathematical model to formulate a cellular manufacturing system with uncertain parameters. The processing time and demand were considered random variables following normal probability distribution, and the inter-arrival time for part was considered a random variable following exponential probability distribution. The objective of their proposed mathematical model was to configure machines' layout in cells so that the inter-cell movements are minimized. Egilmez and Suer [9] researched the stochastic cell loading problem. The objective function was to minimize the number of tardy jobs subject to maximum acceptable probability of tardiness (risk level). They developed a stochastic nonlinear mathematical model in which processing times had a normal distribution and due dates were deterministic. Esmailnezhad and Fattahi [10] presented a mathematical model considering the queuing theory and reliability concept for the SCFP. Their mathematical model considered inter-arrival times, processing times, and machines' breakdown as probabilistic. Zohrevand et al. [11] proposed a new bi-objective stochastic model in order to consider human-related issues in dynamic cell formation problem. They used fuzzy stochastic programming to cope with the vagueness involved in part demands, machine capacities in regular time and overtime, and machine selling prices. The first objective function of the developed model was to minimize total cost of machine procurement, machine relocation, inter-cell moves, overtime utilization, worker hiring/laying-off, and worker moves between cells; the second objective function was to maximize labor utilization of the cellular manufacturing system. Egilmez et al. [12] offered a nonlinear stochastic p-median for the SCFP. The objective was to maximize the total similarity and minimize the total number of cells. In this model, capacity requirement is probabilistic, because the product's demand and processing time are probabilistic. It should be noted that capacity requirement is equal to the multiplication of the product's demand and the corresponding processing time.

This study seeks to model the SCFP with queuing theory approach through the maximization of the grouping efficacy of cells. In addition, buffer effects are considered on the grouping efficacy measure. The rest of the paper is organized as follows: In the next section, mathematic modeling is presented. The experimental results, solution procedure, and computational results are described in sections 3, 4 and 5, respectively. Finally, conclusion is presented in section 6.

2. Mathematic Modeling

2-1. Problem description

In this section, The SCFP will be formulated as a queue system. Each part is assumed as a customer and each machine is assumed as a server. The arrival rate of each part is represented by λ and the service rate of each machine is represented by μ.

2-2. Assumptions

- M/M/1 queuing model is used to formulate the SCFP; moreover, distribution of processing times of each part and the time interval between two consecutive arrivals of parts follow exponential distribution;
- Only one part can be processed by a machine at each time;
- The order of service is based on first-come, first-service method;
- Exceptional elements will be out-sourced to operate

2-3. Notation

Indexing sets

i: Index for parts $i = 1, \ldots, P$

j: Index for machines $j = 1, \ldots, M$

k: Index for cells $k = 1, \ldots, C$

Parameters

λ_i: Mean arrival rate for part i (mean number of parts entered per unit time).

μ_j: Mean service rate for machine j (mean number of customers served per unit time by machine j).

N: The number of inventory buffer for each machine

α: Maximum allowed probability that the queue length behind each machine can be more than the number of inventory buffer for each machine

M_{max}: The maximum number of machines per cell.

$$a_{ij} = \begin{cases} 1 & \text{if part i is to be processed on machine j} \\ 0 & \text{otherwise.} \end{cases}$$

Decision variables

$$x_{ik} = \begin{cases} 1 & \text{if part i is assigned to cell k} \\ 0 & \text{otherwise} \end{cases}$$

$$y_{jk} = \begin{cases} 1 & \text{if machine j is assigned to cell k} \\ 0 & \text{otherwise} \end{cases}$$

2-4. Model formulation

There are several measures for machine-part matrix clustering in the cell formation problem. Kumar and Chandrasekharan [13] introduced grouping efficacy to compare the quality rates of machine-part matrix clustering. Special property of grouping efficacy is that it is not affected by the size of the machine-part matrix. In this measure, high grouping efficacy will result in a good CF. Grouping efficacy is calculated through $\frac{e-e_0}{e+e_v}$ where e is the total number of ones in the given machine-part matrix, e_0 is the number of exceptional elements (exceptional elements are defined as parts which must be processed in different cells and, therefore, have intercellular movements), and e_v is the number of voids (a void indicates that a machine assigned to a cell is not required for the processing of a part in the cell). Based on the presented description, the proposed model can be formulated as follows:

$$\text{Max } Z = \frac{\sum_{i=1}^{P}\sum_{j=1}^{M} a_{ij} - \sum_{k=1}^{C}\sum_{i=1}^{P}\sum_{j=1}^{M} a_{ij}y_{jk}(1-x_{ik})}{\sum_{i=1}^{P}\sum_{j=1}^{M} a_{ij} + \sum_{k=1}^{C}\left(\sum_{i=1}^{P}\sum_{j=1}^{M} x_{ik}y_{jk} - \sum_{i=1}^{P}\sum_{j=1}^{M} x_{ik}y_{jk}a_{ij}\right)} \quad (1)$$

$$s.t.: \sum_{k=1}^{C} x_{ik} = 1 \qquad \forall i \quad (2)$$

$$\sum_{k=1}^{C} y_{jk} = 1 \qquad \forall j \quad (3)$$

$$\sum_{j=1}^{M} y_{jk} \le M_{max} \qquad \forall k \quad (4)$$

$$P\{queue\ length\ of\ machine\ j > N\} \le \alpha \qquad \forall j \quad (5)$$

$$x_{ik}, y_{jk} \in \{0,1\} \qquad \forall i,j,k \quad (6)$$

Equation (1) is the objective function used to maximize the grouping efficacy of produced parts in cells in the planning horizon. Equation (2) restricts allocation of each part to only one cell. Similarly, Equation (3) ensures that each machine is allocated to only one cell. Equation (4) guarantees that the number of machines to be allocated to each cell should be less than the maximum number of machines allowed in each cell. Equation (5) is a chance constraint that limits the probability through which the length of queue behind each machine exceeds the number of inventory buffer. According to the assumptions and notations mentioned, the arrival time of part for processing on a special machine can be found. It is equal to the minimum arrival time of parts. On the other hand, the minimum of some independent exponential random variables is also exponential at a rate equal to the sum of arrival rates. In addition, the utilization factor is the ratio of mean arrival rate over mean service rate for each machine. Hence, the utilization factor for machine j is $\frac{\sum_{k=1}^{C}\sum_{i=1}^{P}\lambda_i x_{ik}y_{jk}a_{ij}}{\mu_j}$ (or the probability that machine j is busy). On the other hand, $\text{P}\{\text{queue length of machine j} > \text{N}\}$ is equal to $(\text{the utilization factor of machine j})^{N+2}$ [14]. Finally, left side of Equation (5) is equal to $\left(\frac{\sum_{k=1}^{C}\sum_{i=1}^{P}\lambda_i x_{ik}y_{jk}a_{ij}}{\mu_j}\right)^{N+2}$. Equation (5) can be rewritten as $\sum_{k=1}^{C}\sum_{i=1}^{P}\lambda_i x_{ik}y_{jk}a_{ij} \le \mu_j\alpha^{\frac{1}{N+2}}$ for each j.

Equation (5) avoids the infinite queue length behind each machine. Equation (6) specifies the type of decision variables.

2-5. Linearization of the proposed model

Objective function (1) and Equation (5) need to be linearized. In the first step, define binary variable V_{ijk}, which is equal to

$$V_{ijk} = x_{ik} \times y_{jk} \quad \forall i, j, k$$

By considering the above equation, the following set constraints should be added to the proposed model:

$$V_{ijk} - x_{ik} - y_{jk} + 1.5 \geq 0 \quad \forall i, j, k \tag{7}$$

$$1.5V_{ijk} - x_{ik} - y_{jk} \leq 0 \quad \forall i, j, k \tag{8}$$

Two other auxiliary variables are introduced in order to linearize objective function (1) along with additional constraints. The new variables can be defined by the following equations:

$$Q = \frac{\sum_{i=1}^{P}\sum_{j=1}^{M} a_{ij} - \sum_{k=1}^{C}\sum_{i=1}^{P}\sum_{j=1}^{M} a_{ij}y_{jk}(1 - x_{ik})}{\sum_{i=1}^{P}\sum_{j=1}^{M} a_{ij} + \sum_{k=1}^{C}\left(\sum_{i=1}^{P}\sum_{j=1}^{M} x_{ik}y_{jk} - \sum_{i=1}^{P}\sum_{j=1}^{M} x_{ik}y_{jk}a_{ij}\right)}$$

$$R_{ijk} = Q \times V_{ijk} \quad \forall i, j, k$$

Where $Q \geq 0$ and $R_{ijk} \geq 0 \ \forall i, j, k$.

Considering the above equations, the following constraints should be added to the mathematical model:

$$Q\sum_{i=1}^{P}\sum_{j=1}^{M} a_{ij} + \sum_{k}^{C}\sum_{i}^{P}\sum_{j}^{M} R_{ijk} - \sum_{k}^{C}\sum_{i}^{P}\sum_{j}^{M} R_{ijk}\, a_{ij} = \sum_{i=1}^{P}\sum_{j=1}^{M} a_{ij} - \sum_{k}^{C}\sum_{i}^{P}\sum_{j}^{M} a_{ij}y_{jk} + \sum_{k}^{C}\sum_{i}^{P}\sum_{j}^{M} V_{ijk}\, a_{ij} \tag{9}$$

$$R_{ijk} \leq L \times V_{ijk} \quad \forall i, j, k \tag{10}$$

$$R_{ijk} \leq Q \quad \forall i, j, k \tag{11}$$

$$R_{ijk} \geq Q - \left(1 - V_{ijk}\right) \times L \quad \forall i, j, k \tag{12}$$

Which L is sufficiently large number.

3. Experimental Results

An example is considered with $P = 7, M = 7, C = 3,$ and $M_{max} = 4$. This randomly generated example is used to examine the effects of N and α on the grouping efficacy measure. Fig. 1 shows the relation between N and the grouping efficacy measure with constant amount for $\alpha = 0.1$. When N increases, the grouping efficacy measure rises, too. Based on the result, by increasing N, more parts can be processed on each machine. Indeed, the number of operation for exceptional parts and that of voids will reduce in each cell, and more parts will be done on each machine. Therefore, the grouping efficacy measure will increase. Fig. 2 shows the relation between α and the grouping efficacy measure with constant amount for $N = 4$. This Figure illustrates that, for a fixed N, if α increases, the upper bound of linearized equation (5) will increase, where this growth will decrease exceptional elements and voids in each cell. Then, more parts will be done on each machine; therefore, the grouping efficacy measure will increase.

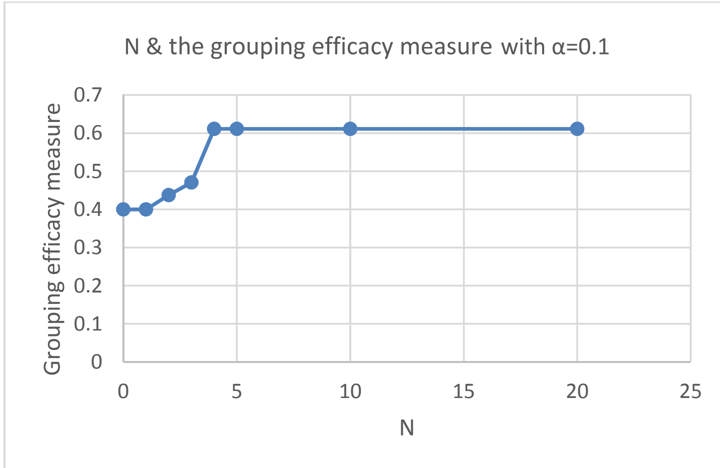

Fig. 1. The relation between the number of inventory buffer and the grouping efficacy measure with constant amount for $\alpha = 0.1$

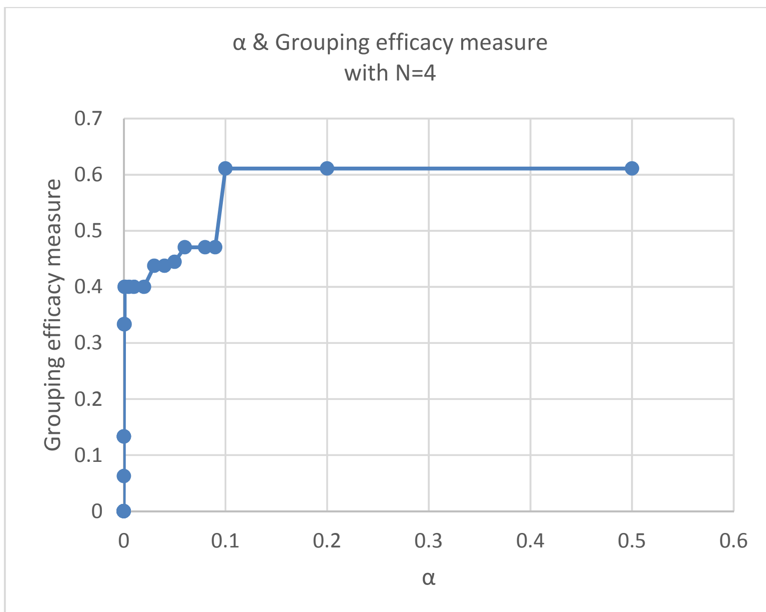

Fig. 2. The relation between the maximum allowed probability and the grouping efficacy measure with constant amount for $N = 4$

4. Solution procedure

The CF is known as NP-hard. For that reason, two metaheuristic algorithms are presented. The following Figure (3) shows the particle (or chromosome). The first part of particle relates to the cells assigned to machines, and the second part relates to the cells assigned to parts. The following heuristic algorithm is used to generate an initial solution. Steps of heuristic algorithm, particle swarm optimization algorithm, and genetic algorithm are presented in the following, respectively:

	Machine1	Machine2	...	Machine M	Part1	Part2	...	Part P
the cell number	1	2	...	1	3	2	...	3

Fig. 3. Sample of particle (or chromosome) structure

4-1. Heuristic algorithm

Heuristic algorithm steps involve the following steps:

Step 1: Set $r = 1$ (population index);

Step 2: Set $the\ number\ of\ machines\ inside\ each\ cell = 0$;
Step 3: Set $i = 1$ (index for machine and part);
Step 4: If "$i \leq M + P$" (M is the number of machines and P is the number of parts);
4-1: If "$i \leq M$";
Generate a random integer between 1 to $C \rightarrow d$, and go to step 8;
4-2: Else, generate a random integer between 1 to $C \rightarrow d$, and go to step 5;
Else, if "$i > M + P$";
If "$i \leq M + P + 1$";
Find the value of the fitness function and go to step 6;
Else, go to step 6;
Step 5: If there is a machine in all cells whose queue length is infinite;
$r \leftarrow r - 1$ and $i = M + P + 1$. Then, go to step 9;
Else, if the machine queue length is infinite in cell d, go back to sub-step 4-2;
Else, make clear machines where part i needs to be processed on them, and then go to step 10;
Step 6: $r \leftarrow r + 1$;
Step 7: If "$population > r$" ($population$ is the number of particles);
Go back to step 2;
Else "End";
Step 8: If "the number of assigned machines to cell $d \leq M_{max}$";
8-1: (The number of machines inside the cell d) + 1 → (the number of machines inside the cell d) and $X(i) \leftarrow d$ (X is the row vector of particle number and $X(i)$ is the i^{th} position of the row vector X). Then, go to step 9;
Else go back to sub-step 8-1;
Step 9: $i \leftarrow i + 1$ and go back to step 4;
Step 10: Adding the arrival rate of part i to the queue length of machines that are needed to be processed on them;
Step 11: If there is the machine whose queue length is infinite;
$X(i) \leftarrow d$ and go to step 9;
Else, Subtract the arrival rate of part i of the queue length of machines that are needed to be processed on them and, then, go back to sub-step 4-2;

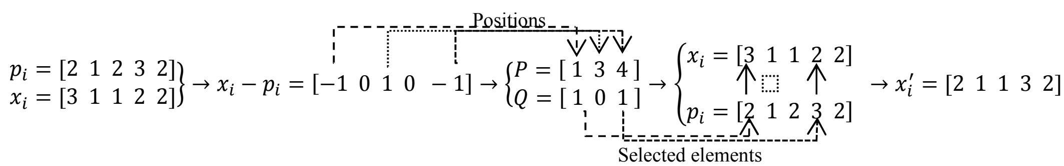

Fig. 4. A sample of how to convert x_i into x_i'

4-2. Particle swarm optimization (PSO)

The particle swarm is a population-based stochastic algorithm for optimization, which is based on social–psychological principles. Unlike evolutionary algorithms, the particle swarm does not use selection; typically, all population members survive from the beginning of a trial to the end. Their interactions result in iterative improvement of the quality of problem solutions over time [15]. The PSO algorithm steps consist of the following steps:
Step 1: Generate initial population by the presented heuristic algorithm and set $i = 1$;
Step 2: Evaluate the fitness (objective function (1)) of each particle;
Step 3: Calculate the values of the best function result so far for particle i (pbesti) and for all particles (gbest);
Step 4: Calculate positions where elements $\vec{x}_i$ (each potential solution is called a particle) and $\vec{p}_i$ (the particle is better than any that has been found for particle i so far) are not equal →P;
Step 5: Generate Q (the vector with the same length and vector P) and set j=1;
Step 6: If "$Q(j) = 1$", the change is made, namely $\vec{x}_i(P(j)) \leftarrow \vec{p}_i(P(j))$);
Else, do nothing;
Step 7: If Equations (4) and (5) are satisfied;
7-1: If the next position exists for P;
7-1-1: $j \leftarrow j + 1$ and go back to step 6;
Else, positions where elements $\vec{x'}_i$ (calculation method of $\vec{x'}_i$ is shown in figure (4)) and $\vec{p}_g$ (the vector of the best particle has been found so far) are not equal → P and go to step 8;
Else, return the change made and go to sub-step 7-1;
Step 8: If " $Q(j) = 1$", the change is made, namely $\vec{x'}_i(P(j)) \leftarrow \vec{p}_g(P(j))$);

Else, do nothing;
Step 9: If Equations (4) and (5) are satisfied;
9-1: If the next position exists for P;
9-1-1: $\mathrm{j} \leftarrow \mathrm{j} + 1$ and go back to step 8;
Else, if "$i \geq population$" ($population$ is the number of particles);
$\mathrm{i} \leftarrow \mathrm{i} + 1$ and go back to step 4;
Else, if "$i \leq population$";
Evaluate the fitness of each particle and go to step 10;
Else return the made change and go to sub-step 9-1;
Step 10: Update gbest and $\mathrm{pbest_i}$;
Step 11: If "$Iteration \leq Maximum\ iteration$"
Go back to step 4;
Else "End";

4-3. Genetic algorithm (GA)

GA is a directed random search technique, which can find the global optimal solution in complex multi-dimensional search spaces. GA is modelled on natural evolution in that the operators it employs are inspired by the natural evolution process. These operators, known as genetic operators, manipulate individuals in a population over several generations to improve their fitness gradually [16]. The selection method of initial generation is based on the tournament selection method in the presented genetic algorithm. GA steps consist of the following steps:
Step 1: Initial population is generated using the presented heuristic algorithm and set $i = 1$, k (the tournament size), p (the fraction of selected population for crossover), and q (the fraction of selected population for mutation);
Step 2: The fitness value of each chromosome is calculated by objectives function (1);
Step 3: Select two parent chromosomes from the selected population (i.e. p× population);
Step 4: Choose k chromosomes from the selected population at random;
Step 5: Choose the best chromosome from the selected k chromosomes and, then, insert it in the new selected population;
Step 6: If "$i > p \times population$";
Go to step 7 and set $j = 1$;
Else, $i \leftarrow i + 1$ and go back to step 3;
Step 7: $the\ select\ population \leftarrow the\ new\ selected\ population$;
Step 8: Select two parents from the selected population;
Step 9: Generate a random number between 1 and $M + P$ (M is the number of machines and P is the number of parts);
Step 10: Select a single crossover point on both parents' chromosomes;
Step 11: Swap all data beyond that point in either chromosome between the two parent chromosomes (the derived combinations are the children);
Step 12: If "$j > p \times population$";
Go to step 13 and set t=1;
Else, if Equations (4) and (5) are satisfied;
$\mathrm{j} \leftarrow \mathrm{j} + 1$ and go back to step 8;
Else, return the made change;
Step 13: Select the fraction of the initial population with a probability q (i.e., q× population);
Step 14: Select a random number between 1 and $M + P$;
Step 15: Alter selected array value in a chromosome from the selected population (mutation operator of Mahdavi et al. [17] is used for the mutation);
Step 16: If "$t > q \times population$";
Go to step 17;
Else, if Equations (4) and (5) are satisfied;
$\mathrm{t} \leftarrow \mathrm{t} + 1$ and go back to step 13;
Else, return the made change;
Step 17: Select the best solutions by comparing the previous generation and the solutions generated by the crossover (steps 7 to 11) and the mutation (steps 13 to 16) and, then, insert them into a new generation (the size of the new generation or the next population is the same as the previous one);
Step 18: If "$Iteration \leq Maximum\ iteration$";
Go back to step 3;
Else "End";

5. Computational Results

This section describes some computational experiments, which are applied to evaluate the efficiency and performance of GA and PSO algorithms. For this reason, 9 examples are defined and, then, solved by Lingo software B&B algorithm, PSO and GA. Finally, the generated solutions will be compared with each other according to the criteria of solution quality and solving time. The proposed model is coded in LINGO 11.0 optimization software, and the used metaheuristic algorithms are coded in MATLAB

2013a on a computer with 4.00 GB RAM and core i5 with 2.5 GHz processor. For each example, 5400 seconds (1.5 hours) are allowed to run. In B&B algorithm (achieved by Lingo software package), if the example was solved in less than 5400 seconds (1.5 hour), it is classified as small-medium size examples; otherwise, it is classified as large-sized examples. This procedure is similar to Safaei et al. [18]. Since the efficiency of the metaheurstic algorithms depends strongly on the operators and parameters, the design of experiments is done to set parameters. Design of experiments features the combination of control factors with the lowest variation, aiming for robustness in solutions. For protection different sizes, examples with small size (5×4), medium size (8×8), and large size (20×37) have been selected. PSO and GA parameters are set using the full factorial design and Taguchi technique design, respectively. A summary of all obtained PSO and GA parameters is given in
Tab. **1** and Tab. 2, respectively.

Tab. 1. details of the examples

example No.	No. of parts	No. of machines	No. of cells	Mmax
1	4	4	2	3
2	5	4	2	3
3	4	5	2	3
4	9	8	3	4
5	8	8	3	4
6	8	9	3	4
7	35	20	4	7
8	37	20	5	7
9	43	22	5	7

Tab. 2. GA parameter settings

Size	5×4	8×8	20×37
Population	350	1250	5600
Iteration	55	90	100
Probability of crossover	0.5	0.6	0.7
Probability of mutation	0.3	0.4	0.6
Number of members competition in the tournament	2	2	2

Tab. 3. PSO parameter settings

Size	5×4	8×8	20×37
Population	350	1000	3000
Iteration	55	100	130

Tab. 4. Comparison of B&B, PSO, and GA results

example No.	B&B			PSO					GA				
	Fbest	Fbound	TB&B(s)	Zave	Zbest	TPSO(s)	Gave(%)	Gbest(%)	Zave	Zbest	TGA(s)	Gave(%)	Gbest(%)
1	0.7	0.7	0	0.7	0.7	1	0.00	0.00	0.7	0.7	1	0.00	0.00
2	0.3846	0.3846	1	0.3846	0.3846	2	0.00	0.00	0.3846	0.3846	1	0.00	0.00
3	0.6154	0.6154	2	0.6154	0.6154	1	0.00	0.00	0.6154	0.6154	1	0.00	0.00
4	0.5122	0.5122	501	0.4901	0.5122	6.9	-4.31082	0.00	0.5085	0.5122	13.6	-0.71	0.00
5	0.4688	0.4688	551	0.4477	0.4688	7.8	-4.50133	0.01	0.4618	0.4688	13	-1.49	0.01
6	0.3953	0.3953	1099	0.3927	0.3953	11.9	-0.65014	0.00	0.3936	0.3953	13	-0.44	0.00
7	0.0464	0.4428	5400	0.2239	0.25	89.20	382.9	439.29	0.2075	0.2286	409.8	347.67	393.12
8	0.0170	0.4055	5400	0.2258	0.2407	76.7	1224.5	1312.11	0.2128	0.2203	367.2	1148.31	1192.43
9	0.1338	0.4328	5400	0.1877	0.2018	82.9	40.2	50.79	0.1787	0.193	406.7	33.54	44.21

According to the Lingo software's documents, F_{best} shows the best feasible objective function value (OFV) found so far. F_{bound} shows the bound on the objective function value. Thus, a possible domain for the optimum value of objective function (F^*) is limited between $F_{best} \leq F^* \leq F_{bound}$. The details of 9 examples are displayed in Tab. 3. Tab. 4 indicates the comparison of the Lingo software's B&B algorithm results with PSO and GA corresponding to 9 examples. Each example is run 10 times; the average of OFV (Z_{ave}), the best OFV (Z_{best}), and average of run time (T_{PSO}) are represented in this Table. The relative gap between the best OFV found by Lingo (F_{best}) and Z_{ave} found by the metaheurstic algorithms is displayed in column ‘‘G_{ave}’’. G_{ave} is calculated as:$G_{ave} = [(Z_{ave} - F_{best})/F_{best}] \times 100$. In addition, the relative gap between F_{best} and Z_{best} is displayed in column ‘‘G_{best}’’. In a similar manner, G_{best} is calculated as:$G_{best} = [(Z_{best} - F_{best})/F_{best}] \times 100$. In Lingo software's B&B algorithm, if $F_{bound} = F_{best}$, the optimal solution is achieved. In Tab. 4, in some cases, Z_{ave} and Z_{best} are between F_{bound} and F_{best} that show a feasible better solution; under this condition, G_{ave} and G_{best} are positive.

As explained earlier, in small-medium size examples, a limited run time (1.5 h) is considered for Lingo solver to find optimal solutions. Therefore, as concluded from Tab. 4, the percent error of the optimal solution is very small for medium-sized example and is zero for small-sized example in both metaheurstic algorithms. In addition, in large-sized examples, PSO and GA outperform the Lingo software B&B algorithm in all examples in a limited time. It implies that PSO and GA algorithms are so effective in solving the proposed model in all classes of examples. A paired t test was conducted to analyze a significant difference between the obtained solutions of the metaheurstic algorithms.
The statistical details are shown in

Tab. **5**. Tests show that there is no statistically significant difference between solutions obtained by PSO and GA.

Tab. 5. Detailed statistics of paired t test

		Paired Differences							
					95% Confidence Interval of the Difference				
		Mean	Std. Deviation	Std. Error Mean	Lower	Upper	t	df	Sig. (2-tailed)
Pair 1	PSO - GA	.0005556	.0114802	.0038267	-.0082689	.0093800	.145	8	.888

6. Conclusions

In this research, the probability nonlinear integer model was proposed for cell formation problem. Buffer effects on the grouping efficacy measure of cells were studied using the buffer chance constraint. The proposed model maximized the grouping efficacy measure of cells through queuing system approach. Each part as a customer and each machine as a server were assumed. To find the optimal solution in a reasonable time, the proposed model was linearized with a heuristic method. Experimental results showed buffer effects on the grouping efficacy measure of cells. Nine examples were generated randomly in order to explore of proposed model. As the proposed model is known as a NP-hard optimization problem, GA and PSO algorithms were used to solve the model efficiently. The results showed that the two metaheurstic algorithms have better performance based on the computational time and the solution quality against the method of Lingo software's B&B. Finally, the paired t test showed that there was no statistically significant difference between solutions obtained by PSO and GA. The purchase of the bottleneck machine would decrease outsourcing operations, yet it would increase purchase costs. Therefore, the purchase of the bottleneck machine and outsourcing operations need to be integrated, left for future works.

References

[1] V. Modrák, *Operations Management Research and Cellular Manufacturing Systems: Innovative Methods and Approaches: Innovative Methods and Approaches*: IGI Global, (2011).

[2] L. Wilson, "How to implement lean manufacturing," *Estados Unidos: McGraw-Hill,* (2010).

[3] G. Egilmez, G. A. Süer, and O. Ozguner, "Stochastic capacitated cellular manufacturing system design with hybrid similarity coefficient," *Manufacturing System. Intech,* (2012), pp. 381-404.

[4] M. Rabbani, F. Jolai, N. Manavizadeh, F. Radmehr, and B. Javadi, "Solving a bi-

objective cell formation problem with stochastic production quantities by a two-phase fuzzy linear programming approach," *The International Journal of Advanced Manufacturing Technology,* Vol. 58, (2012), pp. 709-722.

[5] G. Egilmez, G. A. Süer, and J. Huang, "Stochastic cellular manufacturing system design subject to maximum acceptable risk level," *Computers & Industrial Engineering,* Vol. 63, (2012), pp. 842-854.

[6] A. Aghajani, S. A. Didehbani, M. Zadahmad, M. H. Seyedrezaei, and O. Mohsenian, "A multi-objective mathematical model for cellular manufacturing systems design with probabilistic demand and machine reliability analysis," *The International Journal of Advanced Manufacturing Technology,* Vol. 75, (2014), pp. 755-770.

[7] G. Egilmez, B. Erenay, and G. A. Süer, "Stochastic skill-based manpower allocation in a cellular manufacturing system," *Journal of Manufacturing Systems,* Vol. 33, (2014), pp. 578-588.

[8] R. Salarian, H. Fazlollahtabar, and I. Mahdavi, "Inter-cell movement minimisation in a cellular manufacturing system having stochastic parameters," *International Journal of Services and Operations Management,* Vol. 17, (2014), pp. 67-87.

[9] G. Eğilmez and G. A. Süer, "Stochastic cell loading to minimize n T subject to maximum acceptable probability of tardiness," *Journal of Manufacturing Systems,* Vol. 35, (2015), pp. 136-143.

[10] B. Esmailnezhad and P. Fattahi, "Formation of Manufacturing Cell Using Queuing Theory and Considering Reliability," *International Journal of Industrial Engineering & Production Research,* Vol. 27, (2016), pp. 121-139.

[11] A. Zohrevand, H. Rafiei, and A. Zohrevand, "Multi-objective dynamic cell formation problem: A stochastic programming approach," *Computers & Industrial Engineering*, Vol. 98, (2016), pp. 323-332.

[12] G. Egilmez, S. Singh, and O. Ozguner, "Cell formation in a cellular manufacturing system under uncertain demand and processing times: a stochastic genetic algorithm approach," *International Journal of Services and Operations Management*, Vol. 26, (2017), pp. 162-185.

[13] C. Suresh Kumar and M. Chandrasekharan, "Grouping efficacy: a quantitative criterion for goodness of block diagonal forms of binary matrices in group technology," *International Journal of Production Research,* Vol. 28, (1990), pp. 233-243.

[14] D. Gross, *Fundamentals of queueing theory*: John Wiley & Sons, (2008).

[15] C. Sammut and G. I. Webb, *Encyclopedia of machine learning*: Springer Science & Business Media, (2011).

[16] D. Pham and D. Karaboga, *Intelligent optimisation techniques: genetic algorithms, tabu search, simulated annealing and neural networks*: Springer Science & Business Media, (2012).

[17] I. Mahdavi, M. M. Paydar, M. Solimanpur, and A. Heidarzade, "Genetic algorithm approach for solving a cell formation problem in cellular manufacturing," *Expert Systems with Applications,* Vol. 36, (2009), pp. 6598-6604.

[18] N. Safaei, M. Saidi-Mehrabad, and M. Jabal-Ameli, "A hybrid simulated annealing for solving an extended model of dynamic cellular manufacturing system," *European Journal of Operational Research,* Vol. 185, (2008), pp. 563-592,

Follow This Article at The Following Site

Esmailnezhad B, saidi mehrabad M. Study of buffer effects on the grouping efficacy measure of stochastic cell formation problem. IJIEPR. 2018; 29 (1) :53-63
URL: http://ijiepr.iust.ac.ir/article-1-788-en.html