



# A General Variable Neighborhood Search approach based on a p-median model for cellular manufacturing problems

Saber Ibrahim<sup>1</sup> · Bassem Jarboui<sup>2</sup>

Received: 27 February 2020 / Accepted: 30 October 2020 / Published online: 10 November 2020

© Springer-Verlag GmbH Germany, part of Springer Nature 2020

## Abstract

One of the practical application in cellular manufacturing systems is the cell formation problem (CFP). Its main idea is to group machines into cells and parts into part families in a way that the number of exceptional elements and the number of voids are minimized. In literature, it is proved that p-median is an efficient mathematical programming model for solving CF problems. In the present work, we develop a modified p-median based model dedicated to solve CFP respecting the objective of minimizing the sum of dissimilarities of machines. For this aim, we applied a General Variable Neighborhood Search algorithm and we collaborated it with an Estimation of Distribution Algorithm maximizing the group capability index and the grouping efficacy evaluation criteria. Thirty CF problems are taken from the literature and tested by our proposed algorithm and the experimental study demonstrated that the proposed method guided by p-median model provides high quality cells in speed running times and beats other state-of-the-art algorithms particularly for CF instances with large sizes.

**Keywords** Cell formation problem · p-Median model · General Variable Neighborhood Search · Estimation of Distribution Algorithm · Grouping efficacy · Group capability index

## 1 Introduction

In production environment, group technology (GT) is known as a manufacturing philosophy that utilizes similarities in product processes and design. One of the main applications of this concept is cellular manufacturing (CM) in which parts are

---

✉ Saber Ibrahim  
saber.ibrahim@gmail.com

<sup>1</sup> Department of Management Information Systems and Production Management, College of Business and Economics, Qassim University, P.O. Box: 6640, Buraydah 51452, Saudi Arabia

<sup>2</sup> Department of Industrial Management, Higher Colleges of Technology, Abu Dhabi, UAE

grouped into part families according to their similarities and machines are clustered into machine groups according to their dissimilarities which leads to the process of one or more part families within a single cell of machines. As stated in [26] and [60], this application has several advantages which can be resumed into finding better quality and production control, growth in flexibility and reduction of setup time, throughput time, material handling costs and work-in-process inventories. The problem of finding, optimally, machine groups and part families is referred to the part cell formation problem (PCFP). Generally, the input of the CFP is a binary part-machine incidence matrix (PMIM) where rows and columns have to be rearranged to produce part families and their corresponding cells. During the last 30 years, researchers proved that the best way to optimize this problem is to diagonalize the PMIM and the ideal cell configuration is reached when each part is processed in only one cell and no part is transferred from one cell to another one.

CFP is an NP-hard problem [4,5]. Also, in [6], the authors proved the NP-completeness of CFP with the fractional grouping efficacy objective function. The literature review showed that a large number of methods has been developed on the subject of CFP. In [31], a comprehensive review of the application of various algorithms for solving medium and large CFP is presented. These methods include hierarchical methods [40], non-hierarchical methods [11,53], graph theory [57], genetic algorithms [13,25], simulated annealing [7,48,52,65], neural networks [30,64], and mathematical models [14,33]. Among these methods, the approaches based on mathematical programming have proven their success to form machine groups and the associated part families by the formulation of the CFP using a linear or nonlinear programming model. In [8], a mathematical programming model is developed to minimize the total costs of a CM system with exceptional elements over inter-cellular assignment, using machine duplication and subcontracting considering machine capacities. Several studies on CFP have shown that the combination of mathematical programming and heuristics or meta-heuristics leads to an improvement of the quality of solutions. One of the successful mathematical models is the p-median problem (PMP) which has as objective the maximization of the similarity coefficients sum defined between pairs of parts. The first suggestion of a linear integer programming p-median model for solving CFP was in [33]. Then, a number of researchers developed modified versions and formulations such as the works presented in [3,19,24,29,58,61,63]. The p-median model was initially designed to build part families first then to form machine groups based on the information taken from the PMIM. It can be reformed also to form groups of machines first and then part families so that we know if the efficacy of the model is better for the formation of machine cells or for part families.

This paper is organized as follows: Sect. 2 presented our modified p-median method which focused on a particular procedure to determine the seed machines. The new p-median version guarantees a speed implementation for CF problems. Sect. 3 described our proposed methodology in which the p-median model is developed and combined with the EDA and simultaneously improved by the GVNS method, a variant of Variable Neighborhood Search (VNS) method, in such a way that can solve large-sized problems in a very competitive CPU time (less than 1s). Section 4 illustrates computational results and discussions and reports comparisons with the best known earlier

findings respecting to the grouping efficacy and the Group Capability Index criteria. Section 5 includes concluding remarks and future researches.

## 2 Problem statement

According to [44], p-median algorithm is revealed very helpful for clustering applications. It was applied by several authors to solve medium and large sizes of CFP. Numerous papers proposed a number of modified versions of p-median models [3,19,24,58,62,63]. In these works, p-median problem is solved by combination of some heuristic methods. For more details about PMP, the works developed in [41] and [51] gave a review of the different meta-heuristics applied to the p-median problem.

Recently in [1], a new mixed integer linear programming (MILP) model is proposed for the p-Median problem minimizing simultaneously the total linking cost distances between facilities and customers, and also the cost distances between facilities when linked under a tree, a ring or under a star network topology. Furthermore, a new formulation viewed as a MILP is proposed in [22] solving the problem using branch-and-cut algorithm. Also, in [23], it has been applied more than one method for the Hamiltonian p-median problem including a constructive heuristic algorithm, an exact algorithm and a metaheuristic and the authors called it HpMP and they proved that it dominates previous p-median based formulations.

In this work, an innovative formulation for the PMP is developed. The proposed algorithm is based on dissimilarities between machines unlike the other works that used the similarities coefficients of parts. Furthermore, the collaboration of the GVNS method with the EDA will allow the solving process more efficient even for large CF instances (CPU time does not exceed one second).

### 2.1 The classical formulation

Typically, the p-median model for CFP is developed in [33] for part family formation and it was defined as follows: Given  $p$  the number of parts,  $n$  the required number of part families and  $x_{ij}$  is a binary decision variable such that  $x_{ij} = 1$  if the part  $i$  is assigned to the family  $j$  and  $x_{ij} = 0$  otherwise. The variable  $S_{ij}^p$  denotes the similarity coefficient between two parts  $i$  and  $j$  and expressed as follows:

$$S_{ij}^p = \sum_{k=1}^m \delta(a_{ki}, a_{kj}); \quad i \neq j; \quad j = 1, 2, \dots, p$$

Where:  $\delta(a_{ki}, a_{kj})$  represents the Kronecker product defined as:

$$\delta(a_{ki}, a_{kj}) = \begin{cases} 1 & \text{if } a_{ki} = a_{kj} \\ 0 & \text{otherwise} \end{cases}$$

The objective is to find the binary variables  $x_{ij} \in \{0, 1\}$  for  $i, j = 1, 2, \dots, p$  corresponding to the following mathematical model:

$$\text{Maximize } \sum_{i=1}^p \sum_{j=1}^p S_{ij}^p x_{ij} \quad (1)$$

$$\text{subject to } \sum_{j=1}^p x_{ij} = 1; \quad i = 1, 2, \dots, p \quad (2)$$

$$\sum_{j=1}^p x_{jj} = n; \quad j = 1, 2, \dots, p \quad (3)$$

$$x_{ij} \leq x_{jj}; \quad i, j = 1, 2, \dots, p \quad (4)$$

$$x_{ij} \in \{0, 1\}; \quad i, j = 1, 2, \dots, p \quad (5)$$

The objective function (1) is defined to maximize the sum of similarity coefficients between parts such that a part is assigned to only one family. Constraint (2) guarantees that each part has to be an element of exactly one family. Constraint (3) identifies the part families required number. Constraint (4) guarantees that part  $i$  belongs to its associated part family  $j$  only when this part family  $j$  is built. The last constraint (5) ensures that the solution is binary. This model can be generalized and applied according to dissimilarities of machines instead of similarities of parts. In the context of cell formation, the PMP indicates the finding of  $p$  central machines which are the most representatives of the  $p$  groups. Once  $p$  median machines are fixed, cells can be identified by assigning any other machine to the principal one such that the sum of their dissimilarity coefficients is minimized.

## 2.2 The modified p-median model for CFP

The following additional notations are used in the modified p-median model for CFP:

$d_{hi}$  denotes the dissimilarity coefficient between machines  $h$  and  $i$ .

$\sigma_{ij}$  denotes the location of the  $i^{th}$  machine within the cell  $j$ .

The decision variable  $x_{h\sigma_{ij}}$  is defined according to the location  $\sigma_i$  as follows:

$$x_{h\sigma_{ij}} = \begin{cases} 1 & \text{if machine } h \text{ is placed on location } \sigma_{ij} \\ 0 & \text{otherwise} \end{cases}$$

where:

$M$  denotes a large positive number,  $L$  and  $U$  denotes respectively lower limit and upper limit on the capacity of a cell,  $|S|$  represents the number of required cells, and  $\sigma_i$  denotes the different positions of the machine  $i$ . Thus, the proposed modified p-median model for CF problem can be written as follows:

$$\text{Maximize } \sum_{h=1}^m \sum_{i=1}^{|S|} \left( \sum_{j=1}^L (M + d_{hi}) x_{h\sigma_{ij}} + \sum_{j=L+1}^U d_{hi} x_{h\sigma_{ij}} \right) \quad (6)$$

$$\text{subject to } \sum_{i \in S} \sum_{\sigma_{ij} \in \sigma_i} x_{h\sigma_{ij}} = 1 \quad (7)$$

$$\sum_{h=1}^m x_{h\sigma_{ij}} \leq 1; \quad \sigma_{ij} \in \sigma_i; \quad i = 1, 2, \dots, |S| \quad (8)$$

$$x_{h\sigma_{ij}} \in \{0, 1\}; \quad i, h = 1, 2, \dots, m; \quad j = 1, 2, \dots, |S| \quad \text{and } \sigma_{ij} \in \sigma_i \quad (9)$$

The objective function (6) looks to maximize the sum of dissimilarity coefficients between machines such that one machine should be assigned to only one cell. Constraint (7) makes sure that each machine has to be incumbent on one cell. Constraint (8) identifies the required number of machine groups. The last constraint (9) ensures that the solution is binary. The new proposed model is based on the modification of the p-median classical formulation while considering an objective function which maximizes the sum of dissimilarities between the assigned machines in the same group. Thus, the cell formation configuration decision can be improved.

### 3 Collaboration of EDA with GVNS

In the proposed algorithm, we tried to solve CF problems with the EDA. A new probabilistic model which aims to allow a certain degree of diversification in the search is developed. This technique has been fascinated much attention by practitioners and researchers due to their search facilities. However, it has the limit of getting trapped in local maximum values. In order to escape from these maxima, it is suitable to integrate randomness or to repeat different executions or to use a non-deterministic search within the algorithm. Thus, we suggest integrating the General Variable Neighborhood Search in the improvement phase of the algorithm to lead to an efficient diversification of the search space. This latter represents the assignment of machines. At this level, each part can be easily allocated to the cell which contains more machines required for this part. In the following subsection, we detail the different steps of the EDA. In order to evaluate the proposed EDA, we used two efficient measures which are Grouping Efficacy (GE) and group capability index (GCI).

The first grouping performance criterion was developed in [32]. It has been considered as the best measure that distinguishes between well-structured matrices and ill-structured matrices. It is given by the following formula:

$$GE = \frac{e(X) - e_0(X)}{e(X) - e_v(X)}$$

where:

$e(X)$  is the number of the 1s in the Part Machine Incidence Matrix.

$e_0(X)$  is the number of the 1s outside diagonal blocks (Exceptional Elements).

$e_v(X)$  is the number of the 0s inside diagonal blocks voids in the solution  $X$ .

The second measure named the group capability index (GCI) is developed in [27]. This criterion excludes voids from the calculation of goodness of the solution found and it is defined as follows:

$$GCI = 1 - \frac{e_0(X)}{e(X)} \times 100$$

### 3.1 The EDA procedure

The EDA was developed initially in [43]. In literature, it is defined as a class of evolutionary algorithms that resembles to genetic algorithms. Instead of crossover and mutation operations, the EDA uses probabilistic functions to form and produce new individuals in the next steps of the algorithm. In this article, EDA is applied to solve CFP and the important steps are described in what follows.

#### *Solution representation and initial population*

In CF problems, a solution is displayed by the vector  $x_i = [x_1, x_2, \dots, x_n]$ , where  $x_i$  signifies the assignment of machine  $i$  to its associated cell. The problem consists in building partitions from the  $m$  machines assignments into some groups (cells). We have to assure that the obtained solutions respect the constraints developed in Sect. 2.2. In our algorithm, starting population is generated randomly with a uniform distribution.

#### *Selection step*

The idea is to permit to selected individuals to have high probabilities to be reproduced. In this level, the procedure of truncated selection is adopted to make new individuals. In each iteration, we take  $P_1$  individuals randomly from the half of the best individuals placed in the current population. Then, in the next generation, these  $P_1$  individuals will be duplicated using our proposed probabilistic algorithm to acquire new ones.

#### *Probabilistic model and creation of new individuals phase*

In order to generate efficiently new individuals, we applied a probabilistic model to the  $P_1$  selected individuals with respect to a given fitness function. The probabilistic equation gives the assignment probability of each machine  $i$  to its associated cell  $j$  and it is defined as follows:

$$P_{ij} = \frac{\text{number of times when machine } i \text{ appears in cell } j + \epsilon}{\text{number of selected individuals} + C \times \epsilon}$$

where the parameter  $\epsilon > 0$  is defined to assure that the proposed algorithm gives a non-null probability  $P_{ij}$ .

#### *Replacement step*

In this step, we compared each new individual with the worst individual placed in the current population. Then, we keep the best individual to use it in the next generation.

#### *Fitness function*

In this step, we used an evaluation function to assess the aptitude of each individual to be reserved or to be involved in constructing new individuals. As we described above, two fitness functions that we denoted  $FF_1$  and  $FF_2$  are introduced to accomplish respectively the objectives of reducing exceptional elements percentage and maximizing grouping efficacy measure. Let give  $m_i$  the number of machines allocated to cell  $i$ . The functions  $FF_1$  and  $FF_2$  are expressed as follows:

$FF_1(X) = e_0(X) + Pen(X)$  and  $FF_2(X) = GE(X) - Pen(X)$  where  $Pen(X)$  denotes the distance between each solution  $X$  and the feasible region and defined as follows:

$$Pen(X) = \alpha_1 \sum_{i=1}^C \max \{0, m_i - k_{max}\} + \alpha_2 \sum_{i=1}^C \max \{0, 1 - m_i\}$$

We note that the penalty under-evaluates the fitness value of each solution  $X$  when the latter violates the problem constraints. A penalty is met when the number of allocated machines exceeds the capacity of a cell  $i$  ( $k_{max}$ ). Penalty can be met also when machines are assigned to a number of cells greater than the defined number of cells  $C$ .

#### *Stopping criterion*

For the selected stopping condition of our algorithm, we fixed a maximum number of iterations and a maximal computational time which must not be violated.

### **3.2 The GVNS procedure**

The VNS is a meta-heuristic that was developed initially by [42]. It has proved its success in exploring the search space by finding the favorable areas seeking for better solution quality. In our case, we implemented one of the variants of the VNS method namely the GVNS which is based on exploiting several neighborhood structures and varying the obtained solution through local search processes. At this level, we considered the pipe Variable Neighborhood Descent (pipe VND) as a local search technique. Two neighborhood structures are used in our algorithm namely the insert and the swap operators. Thus, the moves of insertion of a machine within another cell or the permutation of two machines that belong to two different cells permitted us to assure the speedy convergence of the algorithm even when applied to CFP with large sizes (less than one second). The main steps of this procedure are described in Algorithm 1.

#### *Shaking phase*

The idea of this phase consists of defining the neighborhood structures that permit us to acquire a distance between the solution  $X$  and the new neighbor solution  $X'$  that must be equal to  $k$ . We defined this distance as the number of differences between both vectors  $X$  and  $X'$ . Then, we denoted  $N_k$  as the number of used neighborhood structures specified by applying randomly  $k$  insertion moves.

#### *VND procedure*

Through local search processes, finding a local minimum following a given neighborhood structure does not necessarily imply that we can reach a local optimum using

**Algorithm 1:** General Variable Neighborhood Search

Select the set of neighborhood structures  $N_k, k = 1, 2, \dots, k_{max}$  that will be used in the search, find an initial solution  $x$ , choose a stopping condition.

Repeat the following steps until the stopping condition is met:

Set  $k = 1$

Repeat the following steps until all neighborhood structures are used:

- (1) Shaking: generate a point  $x'$  at random from  $k^{th}$  neighborhood of  $x (x' \in N_k(x))$
- (2) Local Search: apply pipe VND with  $x'$  as an initial solution; denote with  $x''$  the obtained local optimum.
- (3) Move or not: if this local optimum  $x''$  is better than the incumbent, or if some acceptance criterion is met, move there ( $x \leftarrow x''$ ), and continue the search with  $N_1 (k = 1)$ ; otherwise, set ( $k \leftarrow k + 1$ ).

another one. Recently, in [21], the authors provided a clear VND survey with its variants and its neighborhood structures classification and applications in deterministic context. In the proposed algorithm, we applied the pipe VND where the insert and the swap operators are used. The first structure tried to select one machine and then insert it in a new cell. Then, the second one aims to choose two machines from two different cells and switch them. The following act is to apply iteratively both structures until there is no chance for the current solution to be improved.

### 3.3 The role of the p-median model

As an exact mathematical tool, the p-median model is used in order to assure better quality of the obtained solutions. This technique is used as a guide for the proposed algorithm until we reach optimality even for medium and large sized CF problems. The candidate median machines play an important role by allowing the algorithm convergent respecting the feasible region. This procedure is very efficient because it adjusts solutions path and do not allow them to derive from the optimality region which makes the process speedier comparing with the classical implementation.

## 4 Computational results

The objective of our numerical experiments is to show that the model based on p-median problem provides high-quality solutions for CF problems and outperforms other recent approaches in most cases. Furthermore, we tried to prove that the collaboration between the GVNS and the EDA methods guarantees the improvement of the quality of solutions as well as CPU times even for large instances. In order to evaluate the performance of the proposed algorithm (p-EDA-GVNS), we tested 30 CF instances selected from the literature. These data sets include various sizes starting from 5 machines and 7 parts to 50 machines and 150 parts. The proposed p-EDA-GVNS algorithm is coded with C++ and tested to several different sizes of CF benchmarks.



For all problems and throughout all tests, the optimal solution was acquired in the first generation, which demonstrates the power of the GVNS during the improvement phase. Table 1 shows problem sources, sizes and number of required cells. The best results and CPU times obtained by the p-EDA-GVNS algorithm are also illustrated in the same table. We note that the best number of cells is determined after extensive experiments.

#### 4.1 Comparison respecting the grouping efficacy criterion

Table 1 reports the obtained grouping efficacy values used for evaluating the goodness of the obtained solutions. A wide comparative study with the well-known methods in the literature is reported. The best solutions found through these methods are also mentioned in the table. The methods used for comparison are the following:

- ZODIAC method [12]
- GRAFICS method [53]
- GA-Genetic Algorithm [47]
- HGGA-Grouping genetic algorithm [28]
- GRASP-GRASP heuristic [20]
- ACO-Ant colony optimization [39]
- SAYLL-Simulated annealing with variable neighborhood [65]
- SA-Simulated Annealing [50]
- GA-VNS-hybrid Genetic VNS algorithm [49]

Furthermore, we can see that our p-median model outperforms clustering methods such as ZODIAC and GRAPHICS in all instances. The evolutionary algorithms and metaheuristics provided results better than our proposed model in some instances especially the SA method [resented in [50] and the GA-VNS developed in [49]. Also, We can observe that the best solution is obtained by our p-median model in 18 cases out of 30 test problems. Whereas, SA and GA-VNS methods reached the best solution in 22 cases and 24 cases, respectively, out of the tested 30 problems.

#### 4.2 Comparison respecting the group capability index criterion

In order to give a significant evaluation for the performances of our proposed model, we prepared several CF benchmarks taken from the literature and summarized in Table 2. Particularly, large-size problems have been tested by our modified p-median model to show the solvability and the convergence of our algorithm. These data sets contain more than 30 machines and 40 parts. The number of required cells is determined on the basis of the number of cells fixed by the existing results in literature especially those fixed in [63] and in [24].

During experiments, we focused on the largest CF problems and our results are compared with those reported in several recent papers. Compared results are described as follows: The first paper is that developed in [63] using a p-median approach and applying a heuristic procedure. They used Wei and Kern similarity measure and GCI criterion. The second one is that offered in [64] applying the ART1 neural network

**Table 1** Computational results of the proposed algorithm

Source	$m \times n$	C	GA	GGA	EA	HGGA	p-EDA-GVNS	CPU
[34]	$5 \times 7$	2	–	82.35	73.68	82.35	82.35	0.000
[59]	$5 \times 7$	2	62.50	69.57	52.50	69.57	69.57	0.000
[54]	$5 \times 18$	2	77.36	79.59	79.59	79.59	79.59	0.000
[35]	$6 \times 8$	2	76.92	76.92	76.92	76.92	76.92	0.000
[36]	$7 \times 11$	2	50.00	60.87	53.13	60.87	58.62	0.000
[7]	$7 \times 11$	3	70.37	70.83	70.37	70.83	70.37	0.000
[55]	$8 \times 12$	5	–	69.44	68.30	69.44	68.30	0.000
[11]	$8 \times 20$	3	85.25	85.25	85.25	85.25	85.25	0.000
[11]	$8 \times 20$	2	55.91	55.32	58.72	58.72	58.72	0.000
[45]	$10 \times 10$	3	72.79	75.00	69.86	75.00	70.59	0.000
[16]	$10 \times 15$	3	92.00	92.00	92.00	92.00	92.00	0.015
[56]	$14 \times 24$	4	63.48	72.06	69.33	72.06	70.51	0.015
[46]	$16 \times 24$	6	–	51.58	52.58	52.75	51.96	0.046
[37]	$16 \times 43$	4	86.25	55.48	54.86	57.53	54.86	0.031
[45]	$20 \times 20$	5	34.16	40.74	42.96	41.18	43.18	1.232
[17]	$20 \times 35$	4	66.30	77.02	76.22	77.91	76.27	0.078
[10]	$20 \times 35$	5	44.44	57.14	58.07	57.98	57.98	0.093
[17]	$24 \times 40$	7	100.0	100.0	100.0	100.0	100.00	0.031
[17]	$24 \times 40$	7	85.11	85.11	85.11	85.11	85.11	0.092
[17]	$24 \times 40$	7	73.51	73.51	73.51	73.51	76.97	5.171
[17]	$24 \times 40$	7	37.62	52.41	51.97	53.29	72.92	0.732
[17]	$24 \times 40$	7	34.76	46.67	47.06	48.95	53.74	0.670
[17]	$24 \times 40$	7	34.06	45.27	44.87	47.26	48.95	0.233
[46]	$27 \times 27$	6	–	52.53	54.27	54.02	54.98	7.260
[38]	$30 \times 41$	14	40.96	61.39	58.48	63.31	61.43	0.580
[56]	$30 \times 50$	13	48.28	57.95	59.66	59.77	59.43	0.447
[56]	$30 \times 50$	14	37.55	50.00	50.51	50.83	50.78	1.406
[34]	$36 \times 90$	17	–	43.78	42.64	46.35	45.94	5.094
[46]	$37 \times 53$	2	–	52.47	56.42	60.64	55.43	4.318
[12]	$40 \times 100$	10	83.90	82.25	84.03	84.03	83.81	7.421

method to solve cell formation problem using the Grouping Efficiency and GCI criteria. The next research is presented in [2] where the TOPSIS technique is proposed to solve the cell formation problem reporting the values of Grouping Efficiency from where we extracted the values of GCI. The fourth paper is that established in [9] proposing a model using a constraint on the number of cells and taking into account that each cell have to contain two machines at least. The following paper is that presented in [24] where a MILP is formulated and authors have analyzed the error and the computational efficiency measure of their proposed model and finally, the last paper to be compared with is that developed in [5].

**Table 2** Computational results of the proposed algorithm

Problem source	size $m \times n$	p	GCI	GCI <sub>Gold</sub>	GCI <sub>p-EDA-GVNS</sub>
[63]	$30 \times 41$	3	92.2	95.3	95.31
		4	93.0	93.0	93.75
		5	91.4	91.4	92.19
		6	89.8	90.6	91.41
		7	81.3	89.8	82.03
	$30 \times 50$	3	77.2	77.3	80.24
		4	74.9	76.1	76.65
	$30 \times 90$	3	79.9	77.5	84.43
	$40 \times 100$	2	79.5	93.6	96.90
		3	93.1	91.5	95.46
		4	89.8	88.8	93.55
		5	89.3	87.4	91.65
		6	89.3	88.1	91.41
		7	87.6	88.6	90.93
		8	85.5	89.1	89.76
	$50 \times 150$	2	96.5	96.5	96.79
		3	86.4	90.1	94.62
		4	88.4	92.7	93.36
		5	89.7	91.5	92.10
		6	87.3	93.1	87.18
[64]	$8 \times 20$	6	73.7	73.7	73.84
	$46 \times 105$	7	84.1	84.9	85.13
[2]	$8 \times 20$	3	83.6	83.6	83.6
	$12 \times 19$	4	66.2	66.2	66.21
	$20 \times 20$	6	67.1	82.3	79.75
	$18 \times 35$	4	77.2	77.2	78.62
	$25 \times 40$	7	61.5	76.2	76.15
	$20 \times 51$	6	67.8	77.2	75.00

Usually, EDA is a metaheuristic that established its effectiveness to explore solution space, but it is unsuccessful in the search intensification stage. However, the collaboration with local search techniques may overcome this limit and allow us to obtain good search patterns. For that reason, GVNS technique is used to find the global equilibrium of solution exploration during the evolutionary process. In the used GVNS approach, we applied the one interchange technique for machines in order to guarantee better solution after finding solutions by the EDA.

Table 2 provides computational results for the test problems found in literature [3,25,63,64]. The first column shows the source of CFP data. Columns 2 and 3 display the problem size (number and machines and parts) and the required number of cells. Columns 4, 5 and 6 indicate the obtained solution by compared papers and those

obtained by our proposed p-median model for each problem respecting to the GCI criterion.

As we can see in Table 2, in most of the considered cases, our results outperform those reported in literature. However, the comparing methods dominated some cases of the studied problems. This finding is due to the fact that any p-median model has a specific heuristic nature which implies obtaining better solutions in a reasonable computational time is a hard task. Through experiments, the major strength of our proposed algorithm is the speedy CPU time that did not exceed one second for all instances.

In [24], the authors proved that their results are better than or equal to the compared values obtained in [63,64] and [2] for 96 % of test problems. Our results showed that our proposed p-EDA-GVNS is better than or equal to the values obtained by the algorithm of [24] for 85 % of the cases. For the rest of problems, our results fail in front of the compared algorithms but they are so close to them and our computational times are always the best according to their convergence speed. Regarding this fact, we can conclude that the proposed p-EDA-GVNS is very powerful to solve large problems.

## 5 Conclusion

A modified p-median model is developed with the collaboration of both GVNS and EDA methods to solve the cell formation problem. The proposed model determined the cell configuration according to the dissimilarities between machines. An efficient GVNS algorithm was combined with the EDA method in order to improve the quality of the obtained solutions. In order to examine the performance of the modified algorithm, we tested several CF benchmarks taken from the literature including the largest sized problems. We considered two well-known performance measures namely the grouping efficacy and the group capability index. Computational experiments showed that the applying the GVNS approach collaborated with p-median model is very successful in terms of running time (less than one second) and moderately efficient in terms of quality of solution comparing with recent findings in the literature especially for medium and large CF problem sizes.

## References

1. Adasme, P.: P-median based formulations with backbone facility locations. *Appl. Soft Comput.* **67**, 261–275 (2018)
2. Ahi, A., Aryanezhad, M.B., Ashtiani, B., Makui, A.: A novel approach to determine cell formation, intracellular machine layout and cell layout in the CMS problem based on TOPSIS method. *Comput. Oper. Res.* **36**(5), 1478–1496 (2009)
3. Ashayeri, J., Heuts, R., Tammel, B.: A modified simple heuristic for the p-median problem with facilities design applications. *Robot. Comput. Integr. Manuf.* **21**(4–5), 451–464 (2005)
4. Ballakur, A.: An investigation of part family/machine group formation in designing a cellular manufacturing system. University of Wisconsin, Madison, WI (1985). PhD thesis
5. Ballakur, A., Steudel, H.J.: A within in cell utilization based heuristic for designing cellular manufacturing system. *Int. J. Prod. Res.* **25**, 639–655 (1987)

6. Batsyn, M.V., Batsyna, E.K., Bychkov, I.S.: NP-completeness of cell formation problem with grouping efficacy objective. *Int. J. Prod. Res.* **111** (2019)
7. Boctor, F.F.: A linear formulation of the machine part cell formation problem. *Int. J. Prod. Res.* **29**(2), 343–356 (1991)
8. Brown, J.R.: A capacity constrained mathematical programming model for cellular manufacturing with exceptional elements. *J. Manuf. Syst.* **37**(1), 227–232 (2015)
9. Bhatnagar, R., Saddikuti, V.: Models for cellular manufacturing systems design: Matching processing requirements and operator capabilities. *J. Oper. Res. Soc.* **61**(5), 827–839 (2010)
10. Boe, W.J., Cheng, C.H.: A close neighbour algorithm for designing cellular manufacturing systems. *Int. J. Prod. Res.* **29**(10), 2097–2116 (1991)
11. Chandrasekharan, M.P., Rajagopalan, R.: An ideal seed non-hierarchical clustering algorithm for cellular manufacturing. *Int. J. Prod. Res.* **24**(2), 451–464 (1986b)
12. Chandrasekharan, M.P., Rajagopalan, R.: ZODIAC: an algorithm for concurrent formation of part-families and machine-cells. *Int. J. Prod. Res.* **25**(6), 835–850 (1987)
13. Cheng, C., Gupta, Y., Lee, W., Wong, K.: A TSP-based heuristic for forming machine groups and part families. *Int. J. Prod. Res.* **36**, 1325–1337 (1998)
14. Choobineh, F.: A framework for the design of cellular manufacturing systems. *Int. J. Prod. Res.* **26**, 1161–1172 (1988)
15. Chandrasekharan, M.P., Rajagopalan, R.: MODROC: an extension of rank order clustering for group technology. *Int. J. Prod. Res.* **24**(5), 1221–1233 (1986a)
16. Chan, H.M., Milner, D.A.: Direct clustering algorithm for group formation in cellular manufacturing. *J. Manuf. Syst.* **1**(1), 65–74 (1982)
17. Carrie, A.S.: Numerical taxonomy applied to group technology and plant layout. *Int. J. Prod. Res.* **11**(4), 399–416 (1973)
18. Chandrasekharan, M.P., Rajagopalan, R.: GROUPABILITY: an analysis of the properties of binary data matrices for group technology. *Int. J. Prod. Res.* **27**(6), 1035–1052 (1989)
19. Deutsch, S.J., Freeman, S.F., Helander, M.: Manufacturing cell formation using an improved p-median model. *Comput. Ind. Eng.* **34**(1), 135–146 (1998)
20. Diaz, J.A., Luna, D., Luna, R.: A GRASP heuristic for the manufacturing cell formation problem. *Top* **20**(3), 679–706 (2012)
21. Duarte, A., Snchez-Oro, J., Mladenovi, N., Todosijevi, R.: Variable neighborhood descent. In: *Handbook of Heuristics*, pp. 341–367 (2018)
22. Elloumi, S.: A tighter formulation of the p-median problem. *J. Comb. Optim.* **19**(1), 69–83 (2010)
23. Erdoan, G., Laporte, L., Rodriguez Cha, A.M.: Exact and heuristic algorithms for the Hamiltonian p-median problem. *Eur. J. Oper. Res.* **253**(2), 280–289 (2016)
24. Goldengorin, B., Krushinsky, D., Slomp, J.: Flexible PMP approach for large-size cell formation. *Oper. Res.* **60**(5), 1157–1166 (2012)
25. Goncalves, J., Resende, M.: An evolutionary algorithm for manufacturing cell formation. *Comput. Ind. Eng.* **47**, 247–273 (2004)
26. Heragu, S.S.: Group technology and cellular manufacturing. *IEEE Trans. Syst. Man Cybern.* **24**(2), 203–214 (1995)
27. Hsu, C.P.: Similarity coefficient approaches to machine component cell formation in cellular manufacturing: a comparative study. Department of Industrial and Systems Engineering, University of Wisconsin, Milwaukee (1990). PhD thesis
28. James, T.L., Brown, E.C., Keeling, K.B.: A hybrid grouping genetic algorithm for the cell formation problem. *Comput. Oper. Res.* **34**(7), 2059–2079 (2007)
29. Kaparthi, S., Suresh, N.C.: Performance of selected part-machine grouping techniques for data sets of wise ranging sizes and imperfection. *Dec. Sci.* **25**(4), 515–539 (1994)
30. Kaparthi, S., Suresh, N.C.: Machine-component cell formation in group technology: a neural network approach. *Int. J. Prod. Res.* **30**(6), 1353–1357 (1992)
31. Kesavan, V., Kamalakannan R., Sudhakarapandian, R., Sivakumar, P.: Heuristic and meta-heuristic algorithms for solving medium and large scale sized cellular manufacturing system NP-hard problems: a comprehensive review. In: *Materials Today: Proceedings* (2019)
32. Kumar, C., Chandrasekharan, M.: Grouping efficacy: a quantitative criterion for goodness of block diagonal forms of binary matrices in group technology. *Int. J. Prod. Res.* **28**, 233–243 (1990)
33. Kusiak, A.: The generalized group technology concept. *Int. J. Prod. Res.* **25**, 561–569 (1987)

34. King, J.R., Nakornchai, V.: Machine-component group formation in group technology - review and extension. *Int. J. Prod. Res.* **20**(2), 117–133 (1982)
35. Kusiak, A., Cho, M.: Similarity coefficient algorithm for solving the group technology problem. *Int. J. Prod. Res.* **30**(11), 2633–2646 (1984)
36. Kusiak, A., Chow, W.S.: Efficient solving of the group technology problem. *J. Manuf. Syst.* **6**(2), 117–124 (1987)
37. King, J.R.: Machine-component grouping in production flow analysis: an approach using rank order clustering algorithm. *Int. J. Prod. Res.* **18**(2), 213–232 (1980b)
38. Kumar, K.R., Vannelli, A.: Strategic subcontracting for efficient disaggregated manufacturing. *Int. J. Prod. Res.* **25**, 1715–1728 (1983)
39. Li, X., Baki, M.F., Aneja, Y.P.: An ant colony optimization metaheuristic for machine-part cell formation problems. *Comput. Oper. Res.* **37**, 2071–2081 (2010)
40. McAuley, J.: Machine grouping for efficient production. *Prod. Eng.* **51**(2), 53–57 (1972)
41. Mladenovi, N., Brimberg, J., Hansen, P., Moreno-Prez, J.A.: The p-median problem: a survey of metaheuristic approaches. *Eur. J. Oper. Res.* **179**(3), 927–939 (2008)
42. Mladenovi, N., Hansen, P.: Variable neighborhood search. *Comput. Oper. Res.* **24**, 1097–1100 (1997)
43. Muhlenbein, H., Paas, G.: From recombination of genes to the estimation of distributions I, binary parameters. *Parallel Probl. Solv. Nat.* **4**, 178–187 (1996)
44. Mulvey, J., Crowder, H.: Cluster analysis: an application of Lagrangian relaxation. *Manag. Sci.* **25**, 329–340 (1979)
45. Mosier, C.T., Taube, L.: Weighted similarity coefficient heuristics for the group technology machine clustering problem. *Int. J. Manag. Sci.* **13**(6), 577–583 (1985)
46. McCormick, W.T., Schweitzer, P.J., White, T.W.: Problem decomposition and data reorganization by a cluster technique. *Oper. Res.* **20**, 993–1009 (1972)
47. Onwubolu, G.C., Mutingi, M.: A genetic algorithm approach to cellular manufacturing systems. *Comput. Ind. Eng.* **39**(12), 125–144 (2001)
48. Pailla, A., Trindade, A.R., Parada, V., Ochi, V.L.: A numerical comparison between simulated annealing and evolutionary approaches to the cell formation problem. *Exp. Syst. Appl.* **37**, 547–683 (2010)
49. Paydar, M.M., Saidi-Mehrabad, M.: A hybrid genetic-variable neighborhood search algorithm for the cell formation problem based on grouping efficacy. *Comput. Oper. Res.* **40**, 980–990 (2013)
50. Paydar, M.M., Mahdavi, I., Sharafuddin, I., Solimanpur, M.: Applying simulated annealing for designing cellular manufacturing systems using MDmTSP. *Comput. Ind. Eng.* **59**(4), 929–936 (2011)
51. Reese, J.: Solution methods for the p-median problem: an annotated bibliography. *Networks* **48**(3), 125–142 (2006)
52. Sofianopoulou, S.: Application of simulated annealing to a linear model for the formation of machine cells in group technology. *Int. J. Prod. Res.* **35**(2), 501–511 (1997)
53. Srinivasan, G., Narendran, T.T.: GRAFICS—a non hierarchical clustering-algorithm for group technology. *Int. J. Prod. Res.* **29**(3), 463–478 (1991)
54. Seifoddini, H.: A note on the similarity coefficient method and the problem of improper machine assignment in group technology applications. *Int. J. Prod. Res.* **27**(7), 1161–1165 (1989c)
55. Seifoddini, H., Wolfe, P.M.: Application of the similarity coefficient method in group technology. *IIE Trans.* **18**(3), 271–277 (1986)
56. Stanfel, L.E.: Machine clustering for economic production. *Eng. Costs Prod. Econ.* **9**, 73–81 (1985)
57. Vannelli, A., Kumar, K.R.: A method for finding minimal bottleneck cells for grouping part—machine families. *Int. J. Prod. Res.* **24**, 387–400 (1986)
58. Wang, J., Roze, C.: Formation of machine cells and part families: a modied p-median model and a comparative study. *Int. J. Prod. Res.* **35**(5), 1259–1286 (1997)
59. Waghodekar, P.H., Sahu, S.: Machine-component cell formation in group technology, MACE. *Int. J. Prod. Res.* **22**(6), 937–948 (1984)
60. Wemmerlov, U., Hyer, N.L.: Cellular manufacturing in the US industry: a survey of users. *Int. J. Prod. Res.* **27**(9), 1511–1530 (1989)
61. Won, Y.: New p-median approach to cell formation with alternative process plans. *Int. J. Prod. Res.* **38**(1), 229–240 (2000)
62. Won, Y., Currie, K.R.: An effective p-median model considering production factors in machine cell/part family formation. *J. Manuf. Syst.* **25**(1), 58–64 (2006)
63. Won, Y., Lee, K.C.: Modied p-median approach for efficient GT cell formation. *Comput. Ind. Eng.* **46**(3), 495–510 (2004)

64. Yang, M.S., Yang, J.H.: Machine part cell formation in group technology using a modified ART1 method. *Eur. J. Oper. Res.* **188**, 140–152 (2008)
65. Ying, K.C., Lin, S.W., Lu, C.C.: Cell formation using a simulated annealing algorithm with variable neighborhood. *Eur. J. Ind. Eng.* **5**(1), 22–42 (2011)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.