Scientia Iranica E (2014) 21(6), 2326–2346

SCIENTIA IRANICA

Sharif University of Technology
Scientia Iranica
Transactions E: Industrial Engineering
www.scientiairanica.com

A genetic algorithm for solving integrated cell formation and layout problem considering alternative routings and machine capacities

K. Forghani and M. Mohammadi*

Department of Industrial Engineering, Faculty of Engineering, Kharazmi University, Tehran, P. O. Box 15719-14911, Iran.

Received 24 August 2012; received in revised form 6 January 2013; accepted 21 October 2013

KEYWORDS
Cellular manufacturing system;
Cell formation;
Layout problem;
Alternative routings;
Lower bound;
Genetic algorithm.

Abstract. In this paper, an integrated approach was presented to simultaneously solve the cell formation and layout problems. Design parameters, such as part demands, alternative process routings, machine capacities, cell dimensions, multi-row arrangement of machines within cells, aisle distances, etc. were considered in this approach to make it more realistic. Also, in order to measure the material handling cost more precisely, the actual position of the machines within the cells was used (instead of the center-to-center distances between the cells). Due to the complexity of the proposed problem, a genetic algorithm was developed to efficiently solve it in a reasonable computational time. Finally, the performance of the genetic algorithm was evaluated by solving several numerical examples from the literature. The results indicated that when decisions about cell formation, inter and intra-cell layouts and routing of parts are simultaneously made, the total material handling costs may reduce significantly in comparison with the sequential design approach.

1. Introduction

In today's competitive manufacturing environment, companies have to adopt innovative manufacturing strategies and technologies in order to respond rapidly to changes in product design and demand, with lower or even no costs. Furthermore, manufacturing systems must be able to produce products with low production costs and high quality, as quickly as possible, in order to deliver products to customers on time [1]. Cellular Manufacturing (CM) is an efficient tool for this purpose. CM is known as one of the Group Technology (GT) applications. The basic idea of GT is to decompose a manufacturing system into several subsystems for facilitating shop floor control [2]. CM improves manufacturing productivity and flexibility by allowing small batch-type production to gain economic advantages, as in mass production, while still retaining the flexibility of job-shop production [3]. According to Wemmerlöv and Hyer [4], the design of a Cellular Manufacturing System (CMS) includes: Cell Formation (CF), group layout (including inter and intra-cell layouts), group scheduling and resource allocation. CF is one of the first and most important steps in designing CMSs, in which products are grouped into part families based on their processing similarities, and machines are grouped into machine cells based on parts manufactured by them. An efficient CMS design requires concurrent consideration of real life features relevant to the system. In recent years, researchers have noticed potential benefits when the layout problem is considered within the CF process [2]. In this context, Akturk [5] presented a mathematical model to solve the part-family and machine-cell formation problem and to determine the linear arrangement of machines within each cell. Ramabhatta and Nagi [6]

*. *Corresponding author. Tel./Fax: +98 21 88830891*
E-mail addresses: kamran21f@gmail.com (K. Forghani); mohammadi@khu.ac.ir (M. Mohammadi)

developed a branch-and-bound algorithm for solving the integrated problem of CF and its inter-cell layout with the common objective of minimizing the resulting inter-cell movements cost. The proposed formulation incorporates critical production planning issues that include long term projected production requirements, resource capacity constraints, functionally identical machines and alternative process plans in the CF problem. Tavakkoli-Moghaddam et al. [7] developed a new mathematical model based on the Quadratic Assignment Problem (QAP) for solving the facility layout problem in CMSs with stochastic demands and pre-specified machine cells. Kia et al. [8] presented an integrated mathematical model for solving the CF and intra-cell layout problems in a dynamic uncertain environment. This model incorporates several design features, including part demands, operation sequences and times, alternative process routings, duplicate machines, machine capacities and cell reconfiguration.

Due to the complexity and NP-hard nature of the CF and layout problems, most researchers have focused on implementation of heuristics and metaheuristics. For example, Adil and Rajamani [9] developed a simulated annealing algorithm to minimize the total inter and intra-cell movement costs. Aktürk and Turkcan [10] proposed a local search heuristic to solve the CF and intra-cell layout problems by considering various manufacturing issues, such as production volumes, processing times, operation sequences, alternative routings and machine utilization levels. In this study, a holonistic approach was used to maximize the profit of not only the overall system, but also individual cells. Chiang and Lee [2] developed a simulated annealing algorithm enhanced by dynamic programming to simultaneously solve the CF and linear inter-cell layout problems. Also, Saghafian and Akbari Jokar [11] integrated the same problem with the intra-cell layout and solved it by using a hybrid algorithm based on the simulated annealing algorithm, dynamic programming and ant colony optimization. Hicks [12] presented a layout design tool based on the GA that could be used to solve the cellular layout or non-cellular layouts. This tool could solve the layout problems directly or indirectly by optimizing the results obtained from a CF algorithm. Similarly, Forghani et al. [13] presented a layout design model, which could be used to solve the layout problem of different manufacturing systems, such as job shop and CMS. Also, a heuristic method was developed to efficiently solve large-sized layout problems. Chan et al. [14] proposed a two-stage approach for solving the CF and cell layout problems. Machine cells and part families are determined at the first stage, and the linear sequence of machine cells is determined in the second stage. Both the problems were solved by the GA. Wu et al. [15] developed a GA to simultaneously form manufacturing cells and determine the inter and intra-cell layouts. Many design factors, such as operation sequences, machine capacities, part demands, batch sizes and layout type, were considered in the problem formulation. Also, Jolai et al. [16] employed an electromagnetism-like algorithm to solve the modified version of the proposed problem in [15]. Yalaoui et al. [17] solved a combined group technology problem with a facility layout problem by using a three-stage method. In the first stage, a GA was used to create part families and machine cells, in the second stage, an ant colony optimization algorithm was used to obtain the arrangement of machines, and in the third stage a global evaluation of all the solutions was carried out to choose the appropriate number of cells.

According to Car and Mikac [18], the performance of CMS depends heavily on the cell structure. On the other hand, studies show that 30-75% of production cost is due to materials handling [19]. So, in order to gain all the advantages of CM, its layout should be designed efficiently. To design an efficient layout in CMS, appropriate computational algorithms should be applied. Nevertheless, most recent studies regarding the CF and layout problems have some shortcomings, which can be listed as follows:

- Considering only intra-cell layout or inter-cell layout in the problem formulation;
- Considering only linear layout (flow line layout) in the layout of CM;
- Minimizing the total number of movements, instead of the actual material handling cost;
- Ignoring the dimensions of the cells and aisle distances between them in the inter-cell layout;
- Calculating the inter-cell material handling cost in terms of the center-to-center distances between the cells, instead of the actual position of machines within the cells;
- Considering only one routing for each part type;
- Assuming infinite capacity for machines;
- Using the sequential design approach for solving the integrated CF and layout problem.

To overcome these shortcomings, in this paper a new integrated approach has been proposed to design CMS with more realistic assumptions. In the proposed approach, the decisions about the routing of parts, CF and its layout (inter and intra-cell layouts) are simultaneously made by considering part demands, alternative process routings, operation sequences, processing times, machine capacities, cell dimensions and aisle distances. The consideration of alternative routings in the CM design may improve the clustering ability in the CF process and decrease the number of required machines [20]. This integrated problem has been formulated as a nonlinear mathematical model, so as to minimize the total material handling cost, which is calculated in terms of the actual position of machines within the cells. To solve such a complex problem, a GA has been developed that can be used to solve large-sized problems in a reasonable computational time. Also, a lower bound is constructed to evaluate the quality of the GA in solving small and medium-sized problems. After setting the parameters of the GA, its performance is examined by solving numerical examples from the literature. Finally, the proposed integrated approach is compared to the sequential design approach.

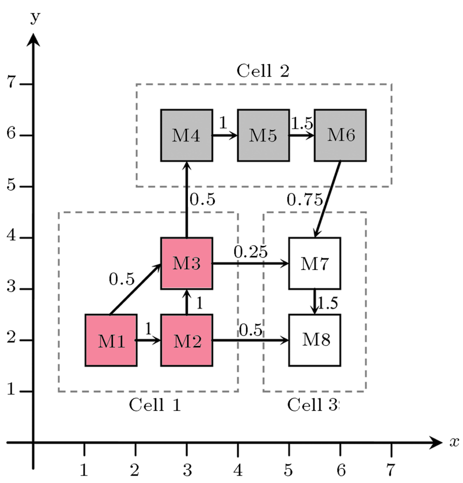
2. Problem formulation

In this section, the routing selection, CF, and inter and intra-cell layout problems are formulated as an integrated mathematical model. The multi-row arrangement of machines within the cells (intra-cell layout) is represented by the QAP, and the continuous layout model is used to represent the cells layout (inter-cell layout). The material handling cost is calculated based on the actual position of the machines within the cells, which is more precise than the common approaches that use the center-to-center distances between the cells. Figure 1 illustrates the difference between these two approaches. There are 8 machines, which have been arranged within 3 cells. The numbers on the arcs show the amount of material flow between the machines. The unit inter and intra-cell material handling costs are assumed to be 1 and 0.5, respectively. Based on the actual position of the machines, the total material handling cost is calculated as 13.625. However, if we use the center-to-center distances between the cells, it is calculated as 11.75 with an error equal to 15.96%.

According to this simple example, we figure out that calculating the material handling cost, in terms of the center-to-center distances between the cells, may result in inappropriate cell design and, consequently, poor system performance. Therefore, in order to design the layout of CM more precisely, the material handling cost should be calculated based on the actual position of the machines within the cells.

Generally, the main assumptions of this paper are as follows:

- The demand and processing routings of each part type is known in advance;
- Each part should be processed by one routing;
- The process of parts has to be done on the given sequence in the corresponding routing;
- The processing times for all operations of each part type on different machine types are known and deterministic;
- Each cell has a pre-specified rectangular shape;
- Each cell has been divided into equally spaced locations (candidate points) with pre-specified rows and columns (such as a grid);
- Cells should be located vertically or horizontally on the planar area;
- The capacity of the machines is limited;
- The dimensions of the machines are assumed to be equal in size, each with 1 unit of area;



Distances between machines based on the actual position of machines

	M1	M2	M3	M4	M5	M6	M7	M8
M1	–	**1.5**	**3**	5.5	7	8.5	5.5	4
M2	–	–	**1.5**	4	5.5	7	2.5	4
M3	–	–	–	2.5	4	5.5	4	2.5
M4	–	–	–	–	**1.5**	**3**	5	6.5
M5	–	–	–	–	–	**1.5**	3.5	5
M6	–	–	–	–	–	–	3	4.5
M7	–	–	–	–	–	–	–	**1.5**
M8	–	–	–	–	–	–	–	–

Distances between machines based on the center-to-center distance between the cells

	M1	M2	M3	M4	M5	M6	M7	M8
M1	–	**1.5**	**3**	5.5	5.5	5.5	3.25	3.25
M2	–	–	**1.5**	5.5	5.5	5.5	3.25	3.25
M3	–	–	–	5.5	5.5	5.5	3.25	3.25
M4	–	–	–	–	**1.5**	**3**	4.25	4.25
M5	–	–	–	–	–	**1.5**	4.25	4.25
M6	–	–	–	–	–	–	4.25	4.25
M7	–	–	–	–	–	–	–	**1.5**
M8	–	–	–	–	–	–	–	–

Figure 1. An example illustrating the difference between the two distance measurement approaches.

- Machines are arranged within the candidate locations in the cells;
- The rectilinear distances between the machines are calculated based on their actual positions.

Also, the following notations are used in the problem formulation:

Sets

i	Parts index $i = 1, ..., P$ (P is the total number of parts);
j	Routings index $j = 1, ..., AR_i$ (AR_i is the number of alternative routings of part i);
o	Operations index $o = 1, ..., Op_{ij}$ (Op_{ij} is the number of operations involving part i in routing j);
k, k'	Machines index $k, k' = 1, ..., Ma$ (Ma is the total number of machines);
l, l'	Cells index $l = 1, ..., C^{\max}$ ($C^{\max}$ is the maximum number of cells allowed);
m	Columns index $m = 1, ..., Co_l$ (Co_l is the number of columns in cell l);
n	Rows index $n = 1, ..., Ro_l$ (Ro_l is the number of rows in cell l).

Parameters

d_i	Demand of part i;
AT_k	Available time of machine k;
c_i^{Intra}	Unit intra-cell material handling cost of part i per unit distance;
c_i^{Inter}	Unit inter-cell material handling cost of part i per unit distance;
a_{ijko}	1 if oth operation of part i in routing j needs machine k; 0 otherwise;
t_{ijk}	Processing time of part i on machine k in routing j;
$f_{ijkk'}$	Amount of material flows between machines k and k' for jth routing of part i;
w_l	Width of cell l;
h_l	Height of cell l;
$ld_{ll'}$	Aisle distance between cells l and l';
A_{lm}	Length of the center of mth column of cell l in the x axis with respect to the left side of cell l;
B_{ln}	Length of the center of nth row of cell l in the y axis with respect to the down side of cell l;
BM	A large enough number.

To clarify parameter a_{ijko}, consider the following sequence for routing j of part $i : 2 \rightarrow 8 \rightarrow 3 \rightarrow 2 \rightarrow 5$. The number of operations for this sequence is 5 (i.e., $Op_{ij} = 5$). In this sequence, part i first visits machine 2 (i.e., $a_{ij21} = 1$), then moves to machine 8 for the second operation (i.e., $a_{ij82} = 1$). Afterward it goes to machine 3 for the third operation (i.e., $a_{ij33} = 1$), and so on.

Decision variables

r_{ij}	=1 if routing j of part i is selected; 0 otherwise;
z_{kl}	=1 if machine k is assigned to cell l; 0 otherwise;
z_{klm}^X	=1 if machine k is assigned to mth column of cell l; 0 otherwise;
z_{kln}^Y	=1 if machine k is assigned to nth row of cell l; 0 otherwise;
u_l	=1 if cell l is located vertically; 0 otherwise;
(x_l, y_l)	horizontal and vertical coordinates of the centroid of cell l, respectively.

2.1. Mathematical model

According to the description given above, the problem can be formulated as the following Mixed-Integer Non-Linear Programming (MINLP) model:

$$\min \sum_{l} \sum_{l'} \sum_{k} \sum_{k'>k} \sum_{i} \sum_{j} d_i.c_{ill'}.f_{ijkk'}.r_{ij}.z_{kl}.z_{k'l'} \times (dx_{kk'll'} + dy_{kk'll'}). \quad (1)$$

Subject to:

$$dx_{kk'll'} = \Bigg| x_l - \frac{w_l + (h_l - w_l)u_l}{2} + (1-u_l) \sum_{m=1}^{Co_l} A_{lm}.z_{kml}^X + u_l \sum_{n=1}^{Ro_l} B_{ln}.z_{kln}^Y - \Bigg(x_{l'} - \frac{w_{l'} + (h_{l'} - w_{l'})u_{l'}}{2} + (1-u_{l'}) \sum_{m=1}^{Co_{l'}} A_{l'm}.z_{k'l'm}^X + u_{l'} \sum_{n=1}^{Ro_{l'}} B_{l'n}.z_{k'l'n}^Y \Bigg) \Bigg|, \qquad \forall k' > k, l, l', \quad (2)$$

$$dy_{kk'll'} = \Bigg| y_l - \frac{h_l + (w_l - h_l)u_l}{2} + (1-u_l) \sum_{n=1}^{Ro_l} B_{lm}.z_{kln}^Y + u_l \sum_{m=1}^{Co_l} A_{lm}.z_{klm}^X$$

$$- \left(y_{l'} - \frac{h_{l'} + (w_{l'} - h_{l'})u_{l'}}{2} + (1 - u_{l'}) \sum_{n=1}^{Ro_{l'}} B_{l'n}.z^Y_{k'l'n} + u_{l'} \sum_{m=1}^{Co_{l'}} A_{l'm}.z^X_{k'l'm} \right) \Bigg|, \quad \forall k' > k, l, l', \tag{3}$$

$$\sum_j r_{ij} = 1, \qquad \forall i, \tag{4}$$

$$\sum_i \sum_j d_i.t_{ijk}.r_{ij} \le AT_k, \quad \forall k, \tag{5}$$

$$\sum_l z_{kl} = 1, \quad \forall k, \tag{6}$$

$$\sum_l \sum_m z^X_{klm} = 1, \quad \forall k, \tag{7}$$

$$\sum_l \sum_n z^Y_{kln} = 1, \quad \forall k, \tag{8}$$

$$\sum_k z^X_{klm}.z^Y_{kln} \le 1, \quad \forall l, m, n, \tag{9}$$

$$z_{kl} = \sum_m \sum_n z^X_{klm}.z^Y_{kln}, \quad \forall k, l, \tag{10}$$

$$|x_l - x_{l'}| + BM.v_{ll'} \ge \frac{w_l + (h_l - w_l)u_l + w_{l'} + (h_{l'} - w_{l'})u_{l'}}{2} + ld_{ll'}, \quad \forall l' > l, \tag{11}$$

$$|y_l - y_{l'}| + BM(1 - v_{ll'}) \ge \frac{h_l + (w_l - h_l)u_l + h_{l'} + (w_{l'} - h_{l'})u_{l'}}{2} + ld_{ll'}, \quad \forall l' > l, \tag{12}$$

$$x_l, y_l \ge 0, \quad \forall l, \tag{13}$$

$$u_l \in \{0,1\}, \quad \forall l, \tag{14}$$

$$v_{ll'} \in \{0,1\}, \quad \forall l' > l, \tag{15}$$

$$z^X_{klm} \in \{0,1\}, \quad \forall k, l, m, \tag{16}$$

$$z^Y_{kln} \in \{0,1\}, \quad \forall k, l, n, \tag{17}$$

$$z_{kl} \in \{0,1\}, \quad \forall k, l, \tag{18}$$

$$dx_{kk'll'}, dy_{kk'll'} \ge 0, \quad \forall k > k', l, l', \tag{19}$$

where $c_{ill'}$ and $f_{ijkk'}$ are calculated as follows:

$$c_{ill'} = \begin{cases} c_i^{\text{Inter}}, & \forall l \ne l' \\ c_i^{\text{Intra}}, & \forall l = l' \end{cases} \tag{20}$$

$$f_{ijkk'} = \sum_{o=1}^{op_{ij}-1} \left(a_{ijko}.a_{ijk'o+1} + a_{ijk'o}.a_{ijko+1} \right), \quad \forall k' > k. \tag{21}$$

Objective function (1) minimizes the total material handling cost, including the total inter and intra-cell material handling costs. Constraints (2) and (3) measure the vertical and horizontal distances between the machines, respectively. Constraint (4) ensures that only one routing is selected for each part type. Constraint (5) ensures that the capacity limitation of each machine is satisfied. Constraints (6)-(8) ensure that each machine is assigned to one cell, one column and one row, respectively. Constraint (9) ensures that each location in each cell can be occupied by, at most, one machine. Constraint (10) determines the cell to which machine k has been assigned. Constraints (11) and (12) jointly ensure that cells do not overlap, where auxiliary binary variable, $v_{ll'}$, makes sure that only one of Constraints (11) and (12) holds. Finally, the set of Constraints (13)-(19) indicate the type of decision variable.

2.2. A lower bound for the integrated model

As the CF, QAP and continuous layout problems are NP-hard [12,21-23], the proposed integrated problem is NP-hard also. It means that the time taken to search the optimal solution becomes unreasonably large as the problem size grows. From the other side, the proposed model is MINLP, and converting it into a Mixed-Integer Linear Programming (MILP) model dramatically increases the number of integer and positive variables. Consequently, it becomes difficult to obtain the optimal solution in a reasonable computational time, even for small-sized problems. These problems have motivated us to develop a GA for solving the proposed problem. Also, a Lower Bound (LB) model is developed for the integrated problem, which can be used for evaluating the GA. In fact, instead of the main model, its LB model will be solved to evaluate the results of the GA in small and medium-sized problems. The following MINLP model represents the proposed LB:

$$\text{LB} : \min \sum_k \sum_{k'>k} \sum_i \sum_j d_i.c_i^{\text{Intra}}.f_{ijkk'}.r_{ij} \times \left(\sum_l z_{kl}.z_{k'l} \right) (dmx_{kk'} + dmy_{kk'})$$

$$+\sum_k \sum_{k'>k} \sum_i \sum_j d_i.c_i^{\text{Inter}}.f_{ijkk'}.r_{ij} \times \left(1-\sum_l z_{kl}.z_{k'l}\right) \times \max\left\{\text{LD}, (dmx_{kk'}+dmy_{kk'})\right\}. \quad (22)$$

Subject to Constraints (4)-(6) and (18):

$$dmx_{kk'} = \left|\sum_m A_m(z_{km}^X - z_{k'm}^X)\right|, \qquad \forall k' > k, \quad (23)$$

$$dmy_{kk'} = \left|\sum_n B_n(z_{kn}^Y - z_{k'n}^Y)\right|, \qquad \forall k' > k, \quad (24)$$

$$\sum_m Z_{km}^X = 1, \qquad \forall k, \quad (25)$$

$$\sum_n Z_{kn}^Y = 1, \qquad \forall k, \quad (26)$$

$$\sum_k Z_{km}^X.Z_{kn}^Y \leq 1, \qquad \forall m, n, \quad (27)$$

$$\sum_k z_{kl} \leq Ro_l.Co_l, \qquad \forall l, \quad (28)$$

$$Z_{km}^X \in \{0,1\}, \qquad \forall k, m, \quad (29)$$

$$Z_{kn}^Y \in \{0,1\}, \qquad \forall k, n, \quad (30)$$

$$dmx_{kk'll'}, dmy_{kk'll'} \geq 0, \qquad \forall k > k', l, l', \quad (31)$$

where $\text{LD} = 1 + \min_{l'>l}\{ld_{ll'}\}$.

Proposition 1. The optimum objective function value of model LB is a lower bound for the integrated problem.

Proof. Enforcement of facilities to get arranged within the pre-specified layout shapes may increase the total material handling cost [24]. So, regardless of the cell structures (i.e., Constraints (7)-(12)), each machine can be placed anywhere within the planar area and, consequently, the total material handling cost may decrease. On the other hand, as it was assumed that the dimensions of the machines are equal in size (i.e., 1×1), the problem can be formulated as the QAP [25]. It means that instead of binary variables, z_{klm}^X and z_{kln}^Y, two sets of binary variables, Z_{km}^X and Z_{kn}^Y, are defined to specify the horizontal and vertical location of the machines, respectively. Thus, by relaxing Constraints (11) and (12) and defining binary variables, Z_{km}^X and Z_{kn}^Y, Constraints (2) and (3) are reduced to Constraints (23) and (24) (where variables $dmx_{kk'll'}$ and $dmy_{kk'll'}$ measure the horizontal and vertical distances between machines k and k', respectively), and Constraints (7)-(10) are changed to Constraints (25)-(28). Also, since the distance between two machines in two distinct cells is always greater than LD, the following expression can be applied to estimate it: $\max\{\text{LD}, (dmx_{kk'} + dmy_{kk'})\}$. Consequently, since model LB is a reduced form of the integrated model, the optimum objective function value of model LB is always less than or equal to that of the integrated model. □

Model LB is nonlinear, due to the presence of the product and absolute operators in Objective function (22) and Constraints (27), (23) and (24). This model can be linearized in three stages. In the first stage, the absolute operator is linearized by replacing Constraints (23) and (24) with Constraints (33)-(36). In the second stage, a new positive auxiliary variable is defined as follows: $\gamma_{ijkk'l} = r_{ij}(\sum_l Z_{kl}.Z_{k'l})(dmx_{kk'} + dmy_{kk'})$. Now, by adding Constraint (37) to the model, the first term of Objective function (22) is linearized. In the third stage, three sets of positive auxiliary variables, $\delta_{ijkk'l}$, $Z_{kk'l}$ and Z_{kmn}^{XY}, are introduced to replace the $r_{ij}(1-\sum_l z_{kl}.z_{k'l}) \max\{\text{LD}_{kk'}(dmx_{kk'}+dmy_{kk'})\}$, $z_{kl}.z_{k'l}$ and $Z_{km}^X.Z_{kn}^Y$, respectively. Finally, by adding the set of Constraints (38)-(49) to the model, the second term of Objective function (22) is linearized. The linearized model is given below:

$$\text{LB}: \min \sum_k \sum_{k'>k} \sum_i \sum_j d_i.c_i^{\text{Intra}}.f_{ijkk'}.\gamma_{ijkk'} + \sum_k \sum_{k'>k} \sum_i \sum_j d_i.c_i^{\text{Inter}}.f_{ijkk'}.\delta_{ijkk'}. \quad (32)$$

Subject to Constraints (4)-(6), (18), (25), (26) and (28)-(31).

$$dmx_{kk'} \geq \sum_m A_m\left(z_{km}^X - z_{k'm}^X\right), \qquad \forall k' > k, \quad (33)$$

$$dmx_{kk'} \geq \sum_m A_m\left(z_{k'm}^X - z_{km}^X\right), \qquad \forall k' > k, \quad (34)$$

$$dmy_{kk'} \geq \sum_n B_n\left(z_{kn}^Y - z_{k'n}^Y\right), \qquad \forall k' > k, \quad (35)$$

$$dmy_{kk'} \geq \sum_n B_n\left(z_{k'n}^Y - z_{kn}^Y\right), \qquad \forall k' > k, \quad (36)$$

$$\gamma_{ijkk'} \geq dmx_{kk'} + dmy_{kk'} + \text{BM}\left(r_{ij} + \sum_l Z_{kk'l} - 2\right), \qquad \forall i, j, k' > k, \quad (37)$$

$$\delta_{ijkk'} \geq dmx_{kk'} + dmy_{kk'} + \text{BM}\left(r_{ij} - \sum_{l} Z_{kk'l} - 1\right), \quad \forall i,j,k' > k, \qquad (38)$$

$$\delta_{ijkk'} \geq \text{LD} + \text{BM}\left(r_{ij} - \sum_{l} Z_{kk'l} - 1\right), \quad \forall i,j,k' > k, \qquad (39)$$

$$z_{kk'l} \leq z_{kl}, \qquad \forall k' > k, l, \qquad (40)$$

$$Z_{kk'l} \geq z_{k'l}, \qquad \forall k' > k, l, \qquad (41)$$

$$Z_{kk'l} \geq z_{kl} + z_{k'l} - 1, \qquad \forall k' > k, l, \qquad (42)$$

$$\sum_{k} Z_{kmn}^{XY} \leq 1, \qquad \forall m,n, \qquad (43)$$

$$Z_{kmn}^{XY} \leq Z_{km}^{X}, \qquad \forall k,m,n, \qquad (44)$$

$$Z_{kmn}^{XY} \leq Z_{kn}^{Y}, \qquad \forall k,m,n, \qquad (45)$$

$$Z_{kmn}^{XY} \geq Z_{km}^{X} + Z_{kn}^{Y} - 1, \qquad \forall k,m,n, \qquad (46)$$

$$Z_{kmn}^{XY} \geq 0, \qquad \forall k,m,n, \qquad (47)$$

$$Z_{kk'l} \geq 0, \qquad \forall k' > k, l, \qquad (48)$$

$$\gamma_{ijkk'}, \delta_{ijkk'} \geq 0, \qquad \forall i,j,k' > k. \qquad (49)$$

3. Developing a genetic algorithm for solving the proposed problem

Genetic Algorithms (GAs), introduced by Holland [26], are stochastic adaptive search methods which employ the principles of Darwin's genetics and natural evolution theory. GAs are an excellent methodology/technique for solving combinatorial optimization problems in a wide variety of application domains including engineering, biology, economics, agriculture, business, telecommunications, and manufacturing [27-29]. In recent years, the GA has been successfully used to solve the CF and layout problems (see [14,15,17, 30,31]). The advantage of the GA relies on the competition between a group of solutions called the population. By selecting good chromosomes from the original population, and implementing genetic operators (including crossover and mutation), a new population with better fitness values will then be produced. This evolution is carried out repeatedly until the stopping criterion is met. GAs do not guarantee optimal solutions, however, quite good (suboptimal) solutions can be found efficiently within a short period of time. A detailed explanation of the proposed GA are given in the following subsections.

3.1. Multi section chromosome structure

In the GA, a coding scheme should be used to represent a solution. In order to encode the routing information for parts, CF information for machines and layout information for machines and cells, a multi section scheme with direct representation for chromosomes has been proposed. The first section of this scheme shows the routing information for parts, the second one represents the machine cells and layout of machines within the cells based on the assignment of machines to the locations (in this section, in order to specify which machine has been assigned to which cell, a procedure is used to convert location numbers to cell numbers), and, finally, the third section shows the layout information of cells on the planar area. In this section, the first two layers indicate the coordinates of cells in the x and y axis, respectively; the latter one specifies the orientation of cells. For illustration, Figure 2 shows a typical chromosome with its equivalent facility layout considering a data set with 5 parts, 8 machines and 3 cells, where the dimensions of cells 1, 2, and 3 (no. columns × no. rows) have been assumed as: 2×2, 3×1 and 3×1, respectively.

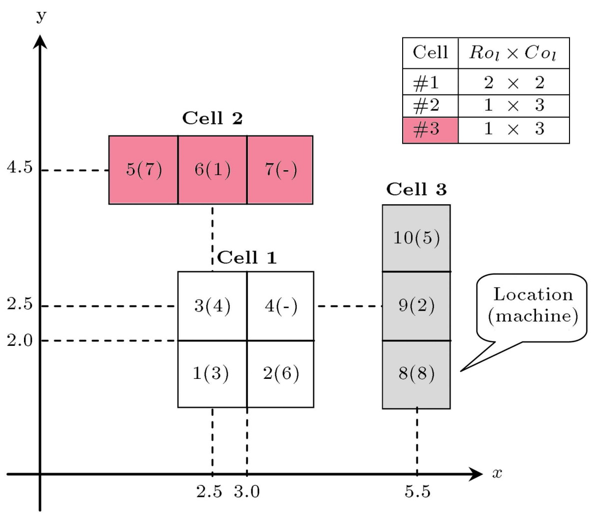
3.2. Generating an initial population

Populations may be generated randomly or problem specific knowledge can be used to construct an initial population. The latter approach may increase the

Section	Part routings					CF and machine layou										Cell layout		
Gene	1	2	3	4	5	1	2	3	4	5	6	7	8	9	10	1	2	3
Layer 1	2	1	3	2	2	3	6	4	9*	7	1	10	8	2	5	3	2.5	5.5
Layer 2																2	4.5	2.5
Layer 3																0	0	1

*: The superscripted numbers show the empty candidate points.

(a)



(b)

Figure 2. Proposed scheme for encoding a solution with 5 parts, 8 machines and 3 cells: (a) Chromosome; and (b) layout of machines and cells based on the given chromosome.

likelihood of producing good solutions [12]. To increase the performance of the GA, a heuristic algorithm has been proposed to form machine cells. The genes in the first and third sections (i.e., the part routings and cell layout sections) are generated randomly. However, the genes in the second section (i.e., the CF and machine layout section) are generated according to the routing information, available in the first section. The steps of the proposed heuristic are as follows:

Step 1. Set $S^c=\{1,2,...,C^{\max}\}$ and $S=\{1,2,...,Ma\}$ (where S^c is the set of cells with free capacities and S is the set of machines that have not been arranged), let $Ca_l=Ro_l \times Co_l$, $\forall l \in S^c$ and $F_{kk'}=\sum_i \sum_j f_{ijkk'}.c_i^{\text{Intra}}.\tilde{r}_{ij}$, $\forall k' > k$ (where $F_{kk'}$ measures the material movement/cost between machines k and k' based on the given part routings. Also, $\tilde{r}_{ij}$ is the selected routing of parts in the corresponding chromosome), go to Step 2.

Step 2. Select the cell with maximum Ca_l, $\forall l \in S^c$ and call this cell C. If $Ca(C)=1$, then go to Step 3. Otherwise, go to Step 4.

Step 3. Randomly arrange machines belonging to S in the remaining locations and stop.

Step 4. Set $S^m=\{\phi\}$ (where S^m is the set of machines that will be arranged within cell C). If the number of machines in S is equal to 1, i.e. $|S|=1$, then go to Step 3. Else, if $|S| \geq 2$, then go to Step 5. Otherwise, i.e., $|S|=0$, stop.

Step 5. Select two machines with maximum $F_{kk'} \forall k' > k, k', k \in S$ and call these machines M and N. If $F(M,N)=0$, then go to Step 3. Otherwise, go to Step 6.

Step 6. Add M and N to S^c, remove M and N from S and go to Step 7.

Step 7. If the number of machines in S^m is equal to $Ca(C)$, i.e. $|S^m|=Ca(C)$, then go to Step 8. Otherwise, go to Step 9.

Step 8. Randomly arrange machines belonging to S^m in the remaining locations of cell C, remove C from S^c and go to Step 2.

Step 9. If $|S| \geq 1$, then let $\pi_k=\sum_{\substack{k' \in S^m \\ k'>k}}(F_{kk'}+F_{k'k})$, $\forall k \in S$ and go to Step 10. Otherwise, randomly arrange machines belonging to S^m in the remaining locations of cell C and stop.

Step 10. Find the machine with maximum π_k, $\forall k \in S$ and call it O. If $\pi(O)=0$, then go to Step 11. Otherwise, add O to S^m, remove O from S and go to Step 7.

Step 11. Find the cell with minimum Ca_l, $\forall l \in S^c$, satisfying the inequality: $Ca_l \geq |S^m|$, $\forall l \in S^c$, and call it C. Randomly arrange machines belonging to S^m in the remaining locations of cell C. Let $Ca(C) \leftarrow Ca(C)-|S^m|$. If $Ca(C)=0$, then remove cell C from S^c. Go to Step 2.

In fact, the proposed heuristic algorithm aims at maximizing the total intra-cell material movement costs (this results in the maximization of the total inter-cell movement costs [9]). To illustrate the implementation of the proposed heuristic algorithm, a numerical example with 8 parts, 10 machines and 4 cells has been provided. The dimensions of cells are assumed to be 2×2, 3×1, 2×1 and 2×1; and the unit intra-cell material handling cost is assumed to be 1. The demand and processing routings of parts have been presented in Table 1.

Let us assume that the routing information (i.e., $\tilde{r}_{ij}$) for the chromosome under study is as follows: 1, 1, 2, 1, 1, 2, 2 and 2. According to the steps of the proposed heuristic algorithm, the machine cells are obtained as follows.

Table 1. Demand and processing information of parts for illustrative example.

Part	Demand	Routing	Op. 1	Op. 2	Op. 3	Op. 4
1	170	1	1	5	4	
		2	2	7	6	
2	161	1	8	1	9	10
		2	3	7	2	
		3	6	4	5	
3	161	1	7	2	8	6
		2	9	1	3	
4	169	1	10	7	5	
5	181	1	3	6	2	
6	139	1	8	9	4	
		2	1	6		
7	127	1	5	2	9	
		2	3	7	10	
8	144	1	6	4	8	
		2	9	5	1	
		3	4	6	3	8

Table 2. Material movement/cost between machines ($F_{kk'}$) based on the information presented in Table 1 and considering the following routings for parts: 1, 1, 2, 1, 1, 2, 2 and 2.

Machine	1	2	3	4	5	6	7	8	9	10
1	—	0	161	0	314	139	0	161	322	0
2		—	0	0	0	181	0	0	0	0
3			—	0	0	181	127	0	0	0
4				—	170	0	0	0	0	0
5					—	0	169	0	144	0
6						—	0	0	0	0
7							—	0	0	296
8								—	0	0
9									—	161
10										—

In Step 1, we set $S^c = \{1,2,3,4\}$ and $S = \{1,2,...,10\}$, then, we calculate $F_{kk'}$, as shown in Table 2, and the capacity of cells as follows: $Ca_1 = 4$, $Ca_2 = 3$, $Ca_3 = 2$ and $Ca_4 = 2$. Next, we go to Step 2. In Step 2, we select cell 1 (i.e., $C = 1$) as the cell with maximum Ca_l, $\forall l \in S^x$. Since $Ca(1){=}4{\neq}1$, we go to Step 4. In Step 4, we set $S^m = \{\phi\}$. Since $|S| = 10 \geq 2$, we go to Step 5. In Step 5, machines 1 and 9 are selected (i.e., $M = 1$ and $N = 9$) as the machines with maximum $F_{kk'}$ $\forall k' > k, k', k \in S$. Since $F(M = 1, N = 9) = 322 \neq 0$, we go to Step 6. In Step 6, we set $S^m = \{1,9\}$, $S = \{2,3,...,8,10\}$ and go to Step 7. Since, in Step 7, $|S^m| = 2 \neq Ca(C = 1) = 4$, we go to Step 9. Since, in Step 9, $|S| = 8 \geq 1$, we calculate π_k, $\forall k \in S$ as follows: $\pi_2 = \pi_4 = \pi_7 = 0$, $\pi_3 = 161$, $\pi_5 = 314 + 144$, $\pi_6 = 139$, $\pi_8 = 161$ and $\pi_{10} = 161$. Next, we go to Step 10. In Step 10, machine 5 is selected (i.e., $O = 5$) as the machine with maximum π_4, $\forall k \in S$. Since $\pi(O = 5) = 458 \neq 0$, we set $S^m = \{1,5,9\}$, $S = \{2,3,4,6,7,8,10\}$ and go to Step 7. Since $|S^m| = 3 \neq Ca(C = 1) = 4$, we go to Step 9. Since $(|S| = 7) \geq 1$, we calculate π_k, $\forall k \in S$ as follows: $\pi_2 = 0$, $\pi_3 = 161$, $\pi_4 = 170$, $\pi_6 = 139$, $\pi_7 = 169$, $\pi_8 = 161$ and $\pi_{10} = 161$. Next, we go to Step 10. In Step 10, machine 4 is selected (i.e., $O = 4$) as the machine with maximum π_k, $\forall k \in S$. Since $\pi(O = 4) = 170 \neq 0$, we set $S^m = \{1,4,5,9\}$, $S = \{2,3,6,7,8,10\}$ and go to Step 7. Since $|S^m| = 4 = Ca(C = 1) = 4$, we go to Step 8. In Step 8, machines belonging to S^m, i.e. machines 1, 4, 5 and 9, are randomly arranged within the locations of cell $C = 1$; then, we set $S^c = \{2,3,4\}$ and go to Step 2. In Step 2, cell 2 is selected (i.e., $C = 2$) as the cell with maximum Ca_l, $\forall l \in S^c$. Since $Ca(C = 2) = 3$, we go to Step 4. In Step 4, we set $S^m = \{\phi\}$. Since $|S| = 6 \geq 2$, we go to Step 5. In Step 5, machines 7 and 10 are selected (i.e., $M = 7$ and $N = 10$) as the machines with maximum $F_{kk'}$ $\forall k' > k, k', k \in S$. Since $F(M = 7N = 10) = 322 \neq 0$, we go to Step 6. In Step 6, we set $S^m = \{7,10\}$, $S = \{2,3,6,8\}$ and go to Step 7. Since $|S^m| = 2 \neq Ca(C = 2) = 3$, we go to Step 9. Since $(|S| = 4) \geq 1$, we calculate π_k, $\forall k \in S$ as follows: $\pi_2 = \pi_6 = \pi_8 = 0$ and $\pi_3 = 127$. Next, we go to Step 10. In Step 10, machine 3 is selected (i.e., $O = 3$) as the machine with maximum π_k, $\forall k \in S$. Since $\pi(O = 3) = 127 \neq 0$, we set $S^m = \{3,7,10\}$, $S = \{2,6,8\}$ and go to Step 7. Since $|S^m| = 3 = Ca(C = 2) = 3$, we go to Step 8. In Step 8, the machines belonging to S^m, i.e., machines 7, 3 and 10, are randomly arranged within the locations of cell C, i.e., cell 2; then, we set $S^c = \{3,4\}$ and go to Step 2. In Step 2, cell 3 is selected (i.e., $C = 3$) as the cell with maximum Ca_l, $\forall l \in S^c$. Since $Ca(C = 3) = 2 \neq 1$, we go to Step 4. In Step 4, we set $S^m = \{\phi\}$. Since $|S| = 3 \geq 2$, we go to Step 5. In Step 5, machines 2 and 6 are selected (i.e., $M = 2$ and $N = 6$) as the machines with maximum $F_{kk'}$. Since $F(M = 2, N = 6) = 181 \neq 0$, we go to Step 6. In Step 6, we set $S^m = \{2,6\}$, $S = \{8\}$ and go to Step 7. Since $|S^m| = 2 = Ca(C = 3) = 2$, we go to Step 8. In Step 8, we randomly arrange machines belonging to S^m (i.e., machines 2 and 6) in the locations of cell $C = 3$; then we set $S^c = \{4\}$ and go to Step 2. In Step 2, cell 4 is selected (i.e., $C = 4$) as the cell with maximum Ca_l, $\forall l \in S^c$. Since $Ca(4) = 2 \neq 1$, we go to Step 4. In Step 4, we set $S^m = \{\phi\}$. Since the number of machines in S is equal to 1 (i.e., $|S| = 1$), we go to Step 3. In Step 3, the only machine belonging to S, i.e. machine 8, is randomly arranged within the remaining locations and the algorithm is stopped.

In brief, after implementing the proposed heuristic algorithm, the machines cells are determined as follows: $\{(1,4,5,9),(3,7,10),(2,6),(8)\}$.

3.3. Fitness evaluation

Before performing the crossover function to produce new chromosomes, each solution in the population pool should be calculated to determine its fitness value, and, according to their fitness values, a probability will be assigned to each of them. The higher the probability, the better are the chances to be chosen for crossover. The objective is to minimize the total material handling cost. In order to prevent an infeasible solution, due to the overlap between the cells or violation of the capacity constraint of machines, a penalty cost fitness function is defined as follows:

$$\text{fit}_s = \alpha_1.\text{TC}_s + \alpha_2.\sum_k \text{CV}_{sk} + \alpha_3.\sum_l \sum_{l'>l} \text{OV}_{sll'} + \alpha_4.\sum_l \sum_{l'>l} \text{CL}_{sll'}. \quad (50)$$

For each chromosome (solution), s, fit_s is the fitness value, TC_s is the total material handling cost calculated using Eq. (1), CV_{sk} is the amount of capacity constraint violation for machine k calculated using Eq. (51), $\mathrm{OV}_{sll'}$ is the overlapped area between cells l and l' computed using Eq. (52), and $\mathrm{CL}_{sll'}$ is the total closeness factor between cells l and l' calculated using Eq. (53). The closeness factor function prevents the independent cells (the cells that have no material flow together) to be located far away from each other. Also, in Eq. (50), coefficients, α_1, α_2, α_3 and α_4, represent the weight of the corresponding criterion.

$$\mathrm{CV}_{sk} = \max\left\{0, \sum_i \sum_j t_{ijk}.r_{ij} - \mathrm{AT}_k\right\}, \quad \forall s,k, \tag{51}$$

$$\mathrm{OV}_{sll'} = \max\left\{0, \left(\frac{w_l+(h_l-w_l)u_l+w_{l'}+(h_{l'}-w_{l'})u_{l'}}{2} + ld_{ll'} - |x_l - x_{l'}|\right)\right\} \times \max\left\{0, \left(\frac{h_l+(w_l-h_l)u_l+h_{l'}+(w_{l'}-h_{l'})u_{l'}}{2} + ld_{ll'} - |y_l - y_{l'}|\right)\right\}, \quad \forall s,l,l', \tag{52}$$

$$\mathrm{CL}_{sll'} = |y_l - y_{l'}| + |x_l - x_{l'}|, \quad \forall s,l,l'. \tag{53}$$

3.4. Selection method

The selection scheme is used to select chromosomes to be exposed to genetic operations. Several approaches are used for this purpose. The 'roulette wheel' approach, selects chromosomes in accordance with their fitness (i.e., chromosomes with a high fitness have relatively large segments of the roulette wheel, while chromosomes with low fitness have small segments). An alternative is the 'tournament' approach that randomly selects the k chromosome ($k \geq 2$) and the chromosome with the lowest fitness survives to the next generation. Random selection has also been used widely [12]. From the literature, the results show that the roulette wheel procedure is one of the most common procedures used in the GA to solve the CF problems [32]. Hence, the roulette wheel procedure, despite the elitist approach, is used as the selection strategy. The elitist approach, which always carries the fittest chromosome through to the next generation, makes sure that the best solution in each generation is not lost.

3.5. Crossover operators

Crossover is a probabilistic evolutionary mechanism which seeks to combine chromosomes, selected by a selection strategy, in order to produce a pool of new offspring. It allows the algorithm to explore new regions in the solution space by exchanging genes between parent chromosomes. Due to the multi section structure of chromosomes, three types of crossover operator have been considered. In the following sections, these crossover operators are presented.

3.5.1. One-point crossover on the part routings section

The one point crossover is performed on the first section (part routings section) of the chromosomes. A cut point is randomly selected over this section (i.e., from $\{1, 2, ..., P\}$) and then the elements of the parent chromosomes are directly copied to the children. Next, from the cut point, the contents of Parent 1 are copied to Child 2 and the contents of Parent 2 are copied to Child 1. An example of this operator has been shown in Figure 3.

3.5.2. Partially matched crossover on the CF and machine layout section

In the proposed scheme, for encoding the layout information of machines, a permutation of integer numbers is used to determine which location has been allocated to which machine. Applying simple crossover operators, similar to the one presented for the part routings section, may result in illegal chromosomes (infeasible solutions). To overcome this problem, a partially mapped crossover (PMX) is applied to this section [12,15]. At first, a cut point is randomly selected over the second section of chromosomes, then, the PMX is applied to prevent production of illegal chromosomes. In order to clarify the function of the PMX operator, consider the example presented in Figure 3. $P1$ and $P2$ are the parent chromosomes selected for crossover. If only the one-point crossover is used, the reproduced chromosomes, $C1'$ and $C2'$, may contain some redundant alleles (i.e., 7, 6 and 4) and may miss some other necessary alleles (e.g. 8, 2 and 5). To correct these illegal chromosomes, the PMX operator is applied. For chromosome $C1'$, the mapping relationships are determined as follows: $7 \leftrightarrow 9 \leftrightarrow 8$, $6 \leftrightarrow 5$ and $4 \leftrightarrow 3 \leftrightarrow 1 \leftrightarrow 2$. After determining the mapping relationships, the illegal alleles are partially replaced with their corresponding mapping relationships. For instance, in chromosome $C1'$, the illegal alleles 7, 6 and 4 are replaced with alleles 8, 5 and 2, respectively, to produce chromosome $C1$. Similarly, in chromosome $C2'$, the illegal alleles 8, 2 and 5 are replaced with alleles 7, 4 and 6, respectively, to produce chromosomes $C2$.

3.5.3. Uniform crossover on the cell layout section

The uniform crossover is applied to the third section of chromosomes. For each gene in this section, a random number is selected from $\{0, 1\}$. Then, if the random number chosen is less than or equal to 0.5, the elements

		Part routings section							CF and machine layout section										Cell layout section				
Crossover type		One-point							PMX										Uniform				
Crossover points						↓									↓					↓		↓	
Gene		1	2	3	4	5	6	7	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5
P1	Layer 1	2	4	1	2	3	1	3	3	6	4	^{9*}	7	1	^{10}	8	2	5	3	2.5	5.5	6	4.5
	Layer 2																		2	4.5	2.5	1	3
	Layer 3																		0	0	1	1	0
P2	Layer 1	3	3	2	1	4	2	1	1	5	3	8	^{9}	2	7	6	4	^{10}	5	3.5	2	3	1.5
	Layer 2																		1.5	2	4.5	1	6
	Layer 3																		1	1	0	0	1

C1′	Layer 1								3	6	4	^{9}	7	1	7	6	4	^{10}					
C2′	Layer 1								1	5	3	8	^{9}	2	^{10}	8	2	5					

Mapping relationships: $(7 \leftrightarrow 9 \leftrightarrow 8)$, $(6 \leftrightarrow 5)$ and $(4 \leftrightarrow 3 \leftrightarrow 1 \leftrightarrow 2)$

C1	Layer 1	2	4	1	2	4	2	1	3	6	4	^{9}	7	1	8	5	2	^{10}	3	3.5	5.5	3	4.5
	Layer 2																		2	2	2.5	1	3
	Layer 3																		0	1	1	0	0
C2	Layer 1	2	3	2	1	3	1	3	1	5	3	8	^{9}	2	^{10}	7	4	6	5	2.5	2	6	1.5
	Layer 2																		1.5	4.5	4.5	1	6
	Layer 3																		1	0	0	1	1

∗: The superscripted numbers show the empty locations.

Figure 3. Numerical illustrations of crossover operators.

	Section	Part routings					CF and machine layout															Cell layout				
	Gene	1	2	3	4	5	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	1	2	3	4	5
Before mutation	Layer 1	4	2	1	3	2	12	5	1	7	2	13	3	15	9	4	11	14	6	8	10	1.5	0	2	2.5	5.5
	Layer 2																					5	2.5	3	2	2.5
	Layer 3																					1	0	1	1	1
After mutation	Layer 1	4	2	1	1	2	12	3	1	4	11	13	5	15	9	7	2	10	8	6	14	1	0	5	2.5	6.5
	Layer 2																					4.5	2.5	1.5	2	3.5
	Layer 3																					1	0	1	1	0

	Changing the routing of part #4
	Swapping the contents of locations #2 and 7
	Exchanging the contents of cells #2 and 4
	Moving cell #1, 0.5 units to the left and 0.5 units to the down
	Changing the position of cell #3
	Rotating cell #5 using scenario 4 as well as invert operator

Figure 4. Numerical illustrations of general mutation.

of the parent chromosomes are directly copied to the children. Otherwise, the content of Parent 1 is copied to Child 2 and the content of Parent 2 is copied to Child 1. An example of this operator has been shown in Figure 3.

Note that each of the proposed crossover operators are independently implemented on the parent chromosomes. Therefore, the total crossover rate, Tcr, is calculated as follows:

$$Tcr = 1 - (1 - Cr1)(1 - Cr2)(1 - Cr3), \qquad (54)$$

where $Cr1$, $Cr2$ and $Cr3$ are the crossover rates which are applied to the first, second and third sections, respectively.

3.6. Mutation operators

Mutations aim at maintaining diversity in the population so that new points in the solution space may be randomly considered as a solution to the problem at hand. They work with a low probability of occurrence and are applied to each chromosome that enters the new population. Due to the multi-section structure of the proposed scheme for chromosome representation, various mutations need to be performed. In this way, seven operators have been considered which are classified into *General mutation* and *Heuristic mutation*. The details of these operators are given below.

3.6.1. General mutation

General mutation comprises the following operators:

I. *Changing part routings.* This operator is applied to the first section of chromosomes and changes the routing of the selected part at random. A sample of this operator has been illustrated in Figure 4. Let us assume that part #4 (the mutated gene) has 4 alternative routings. The current routing for this part is routing 3. After mutation (choosing a random number from $\{1, 2, 3, 4\}$), this routing has been changed to routing 1.

II. *Swapping the contents of locations.* This operator affects the second section of chromosomes and swaps the mutated gene with a randomly selected gene over this section. For example, consider the chromosome presented in Figure 4. The content of gene 3 (the mutated gene) has been swapped with the content of gene 7 (the randomly selected gene).

III. *Exchanging the contents of cells.* This operator is performed on the second section of chromosomes and exchanges the contents of two cells. To illustrate this operator, an example has been given in Figure 4. Assume that the selected cells for mutation are cells #2 and 4. There are 4 and 2 locations in cells #2 and 4, respectively. The content of the cell with the smaller number of locations (i.e., cell #4) is exchanged with that of the cell with the greater number of locations (i.e., cell #2) from a randomly selected location over the cell with the greater number of locations.

IV. *Moving cells.* This operator is performed on the third section of chromosomes. In this operator, the selected cell is randomly moved by 0.5 units across the main directions, i.e. left, right, up and down. An example of this operator has been given in Figure 4.

V. *Changing the position of cells.* This operator which is performed on the third section of chromosomes, randomly changes the position of the selected cell within the planar area (see Figure 4, cell #3).

VI. *Rotating cells.* This operator simultaneously affects the second and third sections of chromosomes. To rotate a cell, 6 scenarios have been considered (see Table 3), one of which is chosen at random. For the first scenario, the contents of the selected cell (i.e., the intra-cell layout) are inverted (i.e., the elements within the locations of the corresponding cell are placed in reverse order) with a probability of 1, and for the other scenarios, it is inverted with a probability of 0.5. To clarify the function of this operator, consider the example presented in Figure 4. Assume that the width and height of cell #5 are 3 and 1, respectively (i.e., $w_5 = 3$ and $h_5 = 1$). If the fourth scenario with an invert operator is selected for rotation, the layout information of this cell after rotation is obtained as follows: $x_l^1 = 5.5 + 1\left(\frac{3-1}{2}\right) = 6.5$, $y_l^1 = 2.5 + 1\left(\frac{3-1}{2}\right) = 3.5$ and $u_l^1 = |1-1| = 0$.

3.6.2. Heuristic mutation

Besides the general mutation operators, the proposed heuristic algorithm in Section 3.2 is also employed as a mutation operator to form a machine cell. In fact, this operator only affects the second section of the selected chromosome for mutation, and, based on the routing information in the first section, the assignment of machines to the cells is determined again.

3.7. Stopping criteria

The GA is terminated when there is no further improvement in the best solution for a specified number of consecutive generations.

4. Setting GA parameters

The proposed GA was coded in Embarcadero Delphi XE and implemented on a PC with 2.4 GHz CPU and 2 GB RAM. Since the performance of GA depends on its parameters, a statistical test using the Design Of Experiment (DOE) method has been considered to determine the best combination of GA parameters. In this way, Minitab 16 software is used to analyze the results. Two design measures, namely, solution quality

Table 3. Scenarios considered for calculating the layout information of a cell after rotation.

Scenarios	x_l^1	y_l^1	u_l^1	Inverting probability
1	x_l^0 [a]	y_l^0 [b]	u_l^0 [c]	1
2	x_l^0	y_l^0	$\lvert 1-u_l^0 \rvert$	0.5
3	$x_l^0 + (1-u_l^0)\left(\frac{h_l-w_l}{2}\right) + u_l^0\left(\frac{w_l-h_l}{2}\right)$	$y_l^0 + (1-u_l^0)\left(\frac{w_l-h_l}{2}\right) + u_l^0\left(\frac{h_l-w_l}{2}\right)$	$\lvert 1-u_l^0 \rvert$	0.5
4	$x_l^0 + (1-u_l^0)\left(\frac{h_l-w_l}{2}\right) + u_l^0\left(\frac{w_l-h_l}{2}\right)$	$y_l^0 + (1-u_l^0)\left(\frac{h_l-w_l}{2}\right) + u_l^0\left(\frac{w_l-h_l}{2}\right)$	$\lvert 1-u_l^0 \rvert$	0.5
5	$x_l^0 + (1-u_l^0)\left(\frac{w_l-h_l}{2}\right) + u_l^0\left(\frac{h_l-w_l}{2}\right)$	$y_l^0 + (1-u_l^0)\left(\frac{w_l-h_l}{2}\right) + u_l^0\left(\frac{h_l-w_l}{2}\right)$	$\lvert 1-u_l^0 \rvert$	0.5
6	$x_l^0 + (1-u_l^0)\left(\frac{w_l-h_l}{2}\right) + u_l^0\left(\frac{h_l-w_l}{2}\right)$	$y_l^0 + (1-u_l^0)\left(\frac{h_l-w_l}{2}\right) + u_l^0\left(\frac{w_l-h_l}{2}\right)$	$\lvert 1-u_l^0 \rvert$	0.5

[a] x_l^0: initial coordinate of cell l in the x axis; [b] y_l^0: initial coordinate of cell l in the y axis; [c] u_l^0: initial orientation of cell l.

Table 4. Feature and reference of numerical examples selected for parameters setting.*

Reference	$Ma \times P \times \sum_i AR_i$	Cells dimensions	Description and added information
[33]	$14 \times 24 \times 183$	$3 \times (5 \times 1)$ $4 \times (2 \times 2)$	—
[6]	$15 \times 15 \times 27$	$3 \times (3 \times 2)$ $4 \times (2 \times 2)$	—
[14]	$20 \times 20 \times 51$	$3 \times (3 \times 3)$ $4 \times (3 \times 2)$	$d_i = 1$, $\forall i$, $t_{ijk} = 0.1$, $\forall i, j, k$ and $AT_k = 0.5$, $\forall k$ It is assumed that parts visit machines in increasing order of machine indices

*: For all the problems $c_i^{\text{Intra}} = 1$, $c_i^{\text{Inter}} = 1.5$, $\forall i$ and $ld_{ll'} = 0.5, \forall l, l'$

Table 5. Experimental factors and their levels in each stage of parameters setting.

	Stage 1				Stage 2				Stage 3			
Parameters	**L1**	**L2**	**L3**	**L4**	**L1**	**L2**	**L3**	**L4**	**L1**	**L2**	**L3**	**L4**
Ps: Population size	200				200				50	100	150	200
Mg: Maximum number of generations	300				300				50	100	200	300
$Cr1$: Crossover rate on the first section	0.3				0.3	0.4	0.5	0.6	Best combination from Stage 2			
$Cr2$: Crossover rate on the second section	0.6				0.3	0.4	0.5	0.6				
$Cr3$: Crossover rate on the third section	0.3				0.3	0.4	0.5	0.6				
Gmr: General mutation rate	0	0.01	0.03	0.05	Best combination from Stage 1				Best combination from Stage 1			
Hmr: Heuristic mutation rate	0	0.03	0.05	0.1								

(the value of the objective function) and computational time are introduced to measure the performance of the GA. The parameters of the GA are classified into three main categories, including: (1) mutation operators; (2) crossover operators; and (3) population size and maximum number of generations. Three numerical examples have been selected from the literature and each problem is solved for two different cell dimensions (in total, six problems are solved). The characteristics of these problems are shown in Table 4.

A primary test with two levels for each parameter was carried out and the results showed that there are no meaningful significances between the proposed categories. In the next step, to determine the parameters of the GA with more accuracy, four levels were considered for each parameter (obtained via literature and primary test), and then the DOE method was sequentially applied to determine the value of the parameters in each category. The proposed values for each parameter are given in Table 5.

For each combination of parameters in each category, the proposed problems are solved 20 times by using the GA (in total, $6 \times 20 \times (2 \times 4^2 + 4^3) = 11520$ experiments were done). Table 6 demonstrates the ANOVA results for the solution quality and computational time. The results of the solution quality indicate that all the factors in the mutation category, as well as population size and maximum number of generations, are significant at level 0.001. The meaningful level of the heuristic mutation rate in the solution quality demonstrates the efficiency of the proposed heuristic in improving the performance of the GA. To select the best value for each parameter in order to obtain high quality solutions, it is necessary to examine the average of factors. For instance, Figure 5 shows the main effects and their interaction plots in the first stage of the parameter setting. The selected value for each parameter based on the DOE plots has been presented in Table 7. It should be noted that for single routing problems (the problems in which all the parts have one processing routing), $Hmr = Cr1 = 0$. Also, to keep the total crossover rate, Tcr, at level 0.79, $Cr2$ and $Cr3$ are obtained through solving the following equation in terms of:

$$y : TCr = 1 - (1 - 0.4y)(1 - 0.6y) = 0.79,$$

where $Cr2 = 0.3y$ and $Cr3 = 0.4y$.

5. Evaluating GA performance

To examine the effectiveness of the proposed GA, 14 numerical examples with different scales have been provided from the literature (problem 14 is an industrial case). For all the problems, the unit intra

Table 6. ANOVA: Effects of parameters on solution quality and computational time.

		Solution quality					Computational time			
Stage	**Source**	**DF**[a]	**SS**[b]	**MS**[c]	***F*-value**	***P*-value**	**SS**	**MS**	***F*-value**	***P*-value**
1	Gmr	3	4.0190	1.3400	2621.25	**0.000**	209.940	69.979	556.92	0.000
	Hmr	3	0.0430	0.0140	28.12	**0.000**	1.042	0.347	2.76	0.042
	$Gmr \times Hmr$	9	0.0320	0.0040	6.98	**0.000**	4.633	0.515	4.10	0.000
	Error	304	0.1550	0.0010			38.198	0.126		
	Total	319								
2	$Cr1$	3	0.0004	0.0001	0.28	0.837	0.263	0.088	1.11	0.345
	$Cr2$	3	0.0015	0.0005	1.08	0.357	1.055	0.352	4.45	0.004
	$Cr3$	3	0.0024	0.0008	1.71	0.164	0.131	0.044	0.55	0.647
	$Cr1 \times Cr2$	9	0.0034	0.0004	0.81	0.610	1.973	0.219	2.77	0.003
	$Cr1 \times Cr3$	9	0.0058	0.0006	1.37	0.197	0.960	0.107	1.35	0.207
	$Cr2 \times Cr3$	9	0.0045	0.0005	1.05	0.398	0.226	0.025	0.32	0.969
	$Cr1 \times Cr2 \times Cr3$	27	0.0089	0.0003	0.70	0.873	1.628	0.060	0.76	0.803
	Error	1216	0.5741	0.0005			96.093	0.079		
	Total	1279								
3	Ps	3	0.0498	0.0166	21.58	**0.000**	68.254	22.751	1574.55	0.000
	Mg	3	0.0843	0.0281	36.55	**0.000**	39.993	13.331	922.60	0.000
	$Ps \times Mg$	9	0.0075	0.0008	1.09	0.371	8.652	0.961	66.53	0.000
	Error	304	0.2337	0.0008			4.393	0.014		
	Total	319								

[a]DF: Degree of Freedom; [b]SS: Sum of Square; [c]MS: Mean Square

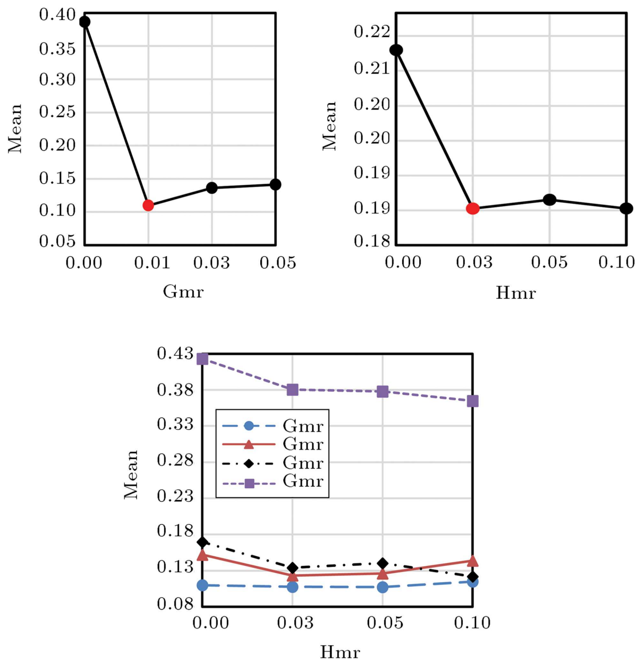


Figure 5. Main effects and interaction plot of the significant factors in the solution quality (first stage of parameter setting).

Table 7. Summary of results after parameter setting.

		Level	
Stage	**Parameter**	**For multiple routings problems**	**For single routing problems**
1	Gmr	0.01	0.01
	Hmr	0.03	0
2	$Cr1$	0.3	0
	$Cr2$	0.4	0.47
	$Cr3$	0.5	0.6
3	Ps	200	200
	Mg	300	300

and inter-cell material handling costs are assumed as 1 and 1.5, respectively; also the aisle distance between the cells is assumed as 1. Further incomplete information such as part demands, operation sequences, processing times, etc., are added to the original data to complete all necessary information for the problems. Reference to the problems, as well as the

Table 8. Characteristic of numerical examples selected for evaluating GA*.

Problem no.	Reference	$Ma \times P \times \sum_i AR_i$	Scale	Routing alternatives	Description and added information
1	[34]	$7 \times 10 \times 23$	Small	Multiple	—
2	[35]	$8 \times 13 \times 26$	Small	Multiple	$t_{ijk} = 0.1, \forall i, j, k$ and $AT_k = 60, \forall k$
3	[36]	$9 \times 8 \times 20$	Small	Multiple	$AT_k = 2000, \forall k$
4	[37]	$10 \times 7 \times 4$	Small	Multiple	—
5	[38]	$12 \times 19 \times 19$	Small	Single	It is assumed that parts visit machines in increasing order of machine indices $d_i = 1, \forall i$
6	[33]	$14 \times 24 \times 183$	Medium	Multiple	—
7	[6]	$15 \times 15 \times 27$	Medium	Multiple	—
8	[39]	$16 \times 30 \times 30$	Medium	Single	It is assumed that parts visit machines in increasing order of machine indices $d_i = 1, \forall i$
9	[9]	$20 \times 20 \times 20$	Medium	Single	$d_i = 1, \forall i$
10	[40]	$17 \times 16 \times 402$	Large	Multiple	For this problem the alternative routings of parts have been determined based on the final solution in the reference paper
11	[14]	$20 \times 20 \times 51$	Large	Multiple	$d_i = 1, \forall i$, $t_{ijk} = 0.1$, $\forall i, j, k$ and $AT_k = 0.5$, $\forall k$. It is assumed that parts visit machines in increasing order of machine indices
12	[41]	$25 \times 40 \times 40$	Large	Single	$d_i = 1, \forall i$
13	[35]	$30 \times 40 \times 89$	Large	Multiple	$t_{ijk} = 0.1, \forall i, j, k$ and $AT_k = 100$, $\forall k$
14	[14]	$37 \times 30 \times 30$	Large	Single	$t_{ijk} = 0.1$, $\forall i, j, k$ and $AT_k = 100, \forall k$

*: For all the problems $c_i^{\text{Intra}} = 1$, $c_i^{\text{Inter}} = 1.5$, $\forall i$ and $ld_{ll'} = 0.5$, $\forall l, l'$.

added information to each problem, has been reported in Table 8.

5.1. Small-sized problems

The proposed small-sized problem in Table 8 is investigated for various configurations, i.e. cell dimensions (no. columns × no. rows) and two types of machine capacities (infinite and limited machine capacities). Each case of the small-sized problems is solved 10 times by the GA, with the same parameters reported in Table 7, and the better solution out of them is considered to be the best solution. Also, for each case, the presented Lower Bound (LB) in Section 2.2 is solved by the CPLEX 10 solver (available in the GAMS Rev 145 modelling language) on the same PC described in Section 4. To evaluate the quality of the results, the gap percent between the solutions of the GA and LB is obtained as follows: $\frac{\text{obj}^{\text{GA}} - \text{obj}^{\text{LB}}}{\text{obj}^{\text{GA}}} \times 100$. Since the LB model has been formulated based on the QAP with further variables and constraints, some cases may not be solved optimally over a reasonable computational time (because the QAP is NP-hard). Hence, for these problems, the solver is interrupted after 2 hours (7200 second). Table 9 summarizes the computational results for the small-sized problems.

From Table 9, we figure out that for the small-sized problems, the proposed GA can obtain efficient solutions in a short amount of computational time and with an insignificant gap (the average gap percent for these problems is equal to 4.37%). The gap percent for case 2.3 is equal to zero, and, since its LB model was solved optimally, it can be concluded that this case has been solved optimally by the GA. Also, the results show that the number of cells, the layout type of machines within the cells and the type of machine capacities may have a significant impact on the amount of material handling costs. For instance, in problem 2, when two cells are considered (case 2.1),

Table 9. Summary of computational results for small-sized problems.

Problem no.	Case no.	Cell dimensions	Machine capacities	Genetic algorithm: Average run time (s)	Genetic algorithm: MHC$^{\text{Intra}}$[a]	Genetic algorithm: MHC$^{\text{Inter}}$[b]	Genetic algorithm: TMHC[c]	Lower bound: Run time (s)	Lower bound: Obj.	Gap%
1	1.1	$2\times(4\times1)$	Proposed	0.45	344	101.25	445.25	27.04	441.50	0.84
	1.2	$2\times(2\times2)$	Proposed	0.47	347	115.5	462.5	27.04	441.50	4.54
	1.3	$3\times(3\times1)$	Proposed	0.43	330	145.5	475.5	39.3	2462.75	2.68
2	2.1	$2\times(4\times1)$	Proposed	0.52	2250	1342.5	3592.5	787.33	3447.50	4.04
	2.2	$2\times(2\times2)$	Proposed	0.47	2250	1342.5	3592.5	787.33	3447.50	4.04
	2.3	$3\times(3\times1)$	Proposed	0.48	2845	146.25	2991.25	2156.70	2991.25	0.00
	2.4	$3\times(3\times1)$	Infinite	0.52	2590	371.25	2961.25	> 7200	2961.25	0.00
3	3.1	$2\times(5\times1)$	Proposed	0.53	2080	1353.75	3433.75	5913.22	3023.75	11.94
	3.2	$2\times(3\times2)$	Proposed	0.59	2015	1113.75	3128.75	> 7200	2953.75	5.59
	3.3	$2\times(3\times2)$	Infinite	0.51	2115	236.25	2351.25	> 7200	2351.25	0.00
	3.4	$3\times(3\times1)$	Proposed	0.62	1495	2283.75	3778.75	4725.97	3513.75	7.01
4	4.1	$2\times(5\times1)$	Proposed	0.54	36650	30562.5	67212. 5	> 7200	63125	6.08
	4.2	$2\times(3\times2)$	Proposed	0.59	39200	20850	60050	> 7200	58475	2.62
	4.3	$3\times(4\times1)$	Proposed	0.60	31700	35962.5	67662.5	> 7200	63900	5.56
	4.4	$3\times(2\times2)$	Proposed	0.53	32600	35662.5	68262.5	> 7200	65500	4.05
5	5.1	$2\times(4\times2)$	—	0.86	57	29.25	86.25	> 7200	83.75	2.90
	5.2	$3\times(4\times1)$	—	0.89	40	72	112	> 7200	97	13.39
	5.3	$3\times(2\times2)$	—	0.67	41	63	104	> 7200	95	8.65

[a]MHC$^{\text{Intra}}$: Intra-cell material handling cost; [b]MHC$^{\text{Inter}}$: Inter-cell material handling cost; [c]TMHC=MHC$^{\text{Intra}}$ + MHC$^{\text{Inter}}$.

the total material handling cost is obtained as 3592.5. However, by increasing the number of cells to three, case 2.3, it is reduced by 16.74% to 2991.25. Another example is problem 3. In this problem, if the linear layout within two cells is considered (case 3.1), the total material handling cost is obtained as 3433.75, however, when the double-row layout is considered within two cells (case 3.2), it is reduced by 8.17% to 3128.75.

5.2. Medium-sized problems

Similar to the small-sized problems, the proposed medium-sized problems in Table 8 are investigated for various configurations. Each case of the proposed medium-sized problems is solved 20 times by the GA and the better result out of them is considered the best solution. For problems 7-9, the quality of the solutions (i.e., gap percent) is evaluated by solving the corresponding LB model. For problem 6, due to the large number of binary variables as a result of the large number of alternative routings, the LB model is not solved, and only the results are presented. As mentioned previously, the LB model may not be solved optimally over a reasonable computational time, therefore, for these problems, the solver is interrupted after 3 hours (10800 seconds). The computational results for these problems have been reported in Table 10.

The computational results indicate that the proposed GA can solve the medium-sized problems in a short amount of computational time. For problems 7-9, the average gap percent is equal to 6.27%. As the LB model was not solved for problem 6, we cannot evaluate the cases of this problem; however, the results will be compared to the literature in Section 5.4. The zero values in column MHC$^{\text{Inter}}$ of Table 10 imply that for cases 6.1, 6.2 and 6.3, the resulting machine cells are completely independent.

5.3. Large-sized problems

The proposed large-sized problems in Table 8 are investigated for various configurations. Due to the large number of binary variables, the LB model is not solved for these problems and only the results are presented. However, in the next section, the results of GA are compared with those published in the

Table 10. Summary of computational results for medium-sized problems.

Problem no.	Case no.	Cell dimensions	Machine capacities	Genetic algorithm: Average run time (s)	MHC^{Intra}	MHC^{Inter}	TMHC	Lower bound: Run time (s)	Obj.	Gap%
6	6.1	$2 \times (3 \times 3)$	Proposed	1.75	850	0	850	—	—	—
	6.2	$3 \times (5 \times 1)$	Proposed	1.92	930	0	930	—	—	—
	6.3	$3 \times (3 \times 2)$	Proposed	2.03	890	0	890	—	—	—
	6.4	$4 \times (4 \times 1)$	Proposed	2.33	880	22.5	902.5	—	—	—
	6.5	$4 \times (2 \times 2)$	Proposed	2.24	820	112.5	932.5	—	—	—
7	7.1	$2 \times (4 \times 2)$	Proposed	1.91	175	214.5	389.5	> 10800	386	0.90
	7.2	$3 \times (5 \times 1)$	Proposed	2.29	132	343.5	475.5	> 10800	472	0.74
	7.3	$3 \times (3 \times 2)$	Proposed	2.26	127	307.5	434.5	> 10800	427.75	1.55
	7.4	$4 \times (4 \times 1)$	Proposed	2.78	101	389.25	490.25	> 10800	438.75	10.50
	7.5	$4 \times (2 \times 2)$	Proposed	2.48	93	406.5	499.5	> 10800	438.75	12.16
8	8.1	$3 \times (3 \times 2)$	—	1.89	75	96	171	> 10800	152.5	10.82
	8.2	$2 \times (2 \times 2)$, $1 \times (4 \times 2)$	—	1.76	78	90	168	> 10800	164	2.38
	8.3	$2 \times (2 \times 2)$, $1 \times (3 \times 2)$, $1 \times (2 \times 1)$	—	1.96	65	117.75	182.75	> 10800	177.5	2.87
	8.4	$4 \times (2 \times 2)$	—	2.24	58	133.5	191.5	> 10800	176.25	7.96
	8.5	$4 \times (4 \times 1)$	—	2.06	58	132.75	190.75	> 10800	176.25	7.60
9	9.1	$3 \times (4 \times 2)$	—	3.06	62	48.75	110.75	> 10800	106.5	3.84
	9.2	$4 \times (3 \times 2)$	—	3.09	53	63	116	> 10800	111	4.31
	9.3	$4 \times (5 \times 1)$	—	2.93	55	73.5	128.5	> 10800	111.5	13.28
	9.4	$5 \times (2 \times 2)$	—	3.25	44	84.75	128.75	> 10800	117.25	8.93

literature. For each case, the GA is run 30 times and the better solution out of them is considered the best solution. Table 11 shows the computational results.

The results indicate that the GA is able to solve the large-sized problems in a short amount of computational time. The results also show that when the number of cells increases, the average computational time increases to some extent. In cases 11.2, 13.2 and 13.4, the resulting machine cells are completely independent, which may be preferred to the other solutions.

5.4. Comparing the integrated approach against sequential approach

To illustrate the advantages of the proposed integrated approach against the sequential design approach (the approach in which the machine cells are determined first, then the layout of machines and cells), the solutions of the GA are compared with those published in the literature. In all the problems, except for problem 6, similar approaches were used by the authors to design the manufacturing cells. To be able to compare the results fairly, the inter and intra-cell layouts should be obtained according to the CF given in the literature. In this way, Forghani et al. [13] presented a layout design tool, identical to the layout approach presented in this paper, which can be used to fill the lack of layout information for the CF results reported in the literature. For each problem, after obtaining the optimum or near optimum layout, the closest case is selected for comparison. Table 12 shows the comparison of the results between the integrated and sequential design approaches. In this table, the improvement percent in the total material handling cost has been obtained as follows:

$$\frac{\text{TMHC}^{\text{Literature-TMHC}^{\text{GA}}}}{\text{TMHC}^{\text{Literature}}} \times 100,$$

Table 11. Summary of computational results for large-sized problems.

Problem no.	Case no.	Cell dimensions	Machine capacities	Genetic algorithm: Average run time (s)	Genetic algorithm: MHC^{Intra}	Genetic algorithm: MHC^{Inter}	Genetic algorithm: TMHC
10	10.1	2 × (5 × 2)	Proposed	2.87	1600	195	1795
	10.2	3 × (3 × 2)	Proposed	3.12	1555	337.5	1892.5
	10.3	3 × (5 × 1)	Proposed	4.06	1545	847.5	2392.5
	10.4	5 × (4 × 1)	Proposed	4.04	1435	1005	2440
	10.5	5 × (2 × 2)	Proposed	4.26	1135	1170	2305
11	11.1	2 × (5 × 2)	Proposed	2.29	53	2.25	55.25
	11.2	3 × (4 × 2)	Proposed	2.65	55	0	55
	11.3	4 × (3 × 2)	Proposed	2.67	52	4.5	56.5
	11.4	5 × (5 × 1)	Proposed	2.67	55	2.25	57.25
	11.5	5 × (4 × 1)	Proposed	2.23	53	6.75	59.75
	11.6	5 × (2 × 2)	Proposed	2.08	52	6.75	58.75
12	12.1	3 × (3 × 3)	—	5.11	94	108.75	202.75
	12.2	4 × (4 × 2)	—	7.35	92	114.75	206.75
	12.3	5 × (3 × 2)	—	7.86	85	135	220
	12.4	7 × (2 × 2)	—	8.16	63	168	231
	12.5	1 × (5 × 1), 2 × (3 × 1), 2 × (2 × 2), 3 × (2 × 1)	—	8.53	60	185.25	245.25
13	13.1	4 × (3 × 3)	Proposed	12.51	14314	1838.25	16152
	13.2	4 × (3 × 3)	Infinite	10.81	14437	0	14437
	13.3	2 × (4 × 2), 3 × (3 × 2)	Proposed	11.89	14539	1466.25	16005.25
	13.4	2 × (4 × 2), 3 × (3 × 2)	Infinite	10.65	14369	0	14369
	13.5	3×(2 × 2), 2 × (3 × 2), 1 × (4 × 2)	Proposed	12.37	13979	1938.75	15917.75
	13.6	3×(2 × 2), 2 × (3 × 2), 1 × (4 × 2)	Infinite	11.13	14489	483.75	14972.75
14	14.1	1 × (3 × 4), 1 × (3 × 3), 2 × (4 × 2)	—	14.12	76	73.5	149.5
	14.2	5 × (4 × 2)	—	16.01	70	94.5	164.5
	14.3	7 × (3 × 2)	—	19.28	61	109.5	170.5

where $TMHC^{Literature}$ is the total material handling cost obtained by the layout design model for the solution reported in the literature, and $TMHC^{GA}$ is the solution of GA associated with the selected case for comparison.

The layout of problems 1-8, 10 and 11 were solved optimally. Also, for problems 9 and 12-14, a near optimal or possibly optimal layout was obtained by the heuristic method presented in [13]. The comparison results show that in all the problems, except for problem 13, our solutions are better than those reported in the literature. Nevertheless, for problem 13, the GA gives a solution with less number of machines in comparison with the solution reported in [35]. Generally, the average improvement percent in the total material handling cost is 16.06%, with the largest cost reduction of over 42%. It implies that the simultaneous cell design strategy generally yields better results than the sequential strategies, since all the decisions about the CF, inter and intra-cell layout and routing selection of parts are made simultaneously.

6. Conclusions

In this study, an integrated approach was presented to simultaneously solve the CF and layout problems considering various production factors, such as part demands, operation sequences and times, alternative

Table 12. Comparison between the results of GA and the solutions reported in the literature.

	Sequential approach			Integrated approach		
Problem no.	**Source of CF**	**Run time (s)**	**TMHC$^{\text{Literature}}$**	**Selected case**	**TMHC$^{\text{GA}}$**	**Imp. (%)**
1	[34]	1.08	704.5	1.3	475.5	32.51
2	[35]	0.32	2991.25	2.4	2961.25	1.00
3	[36]	2.85	3030	3.3	2351.25	22.40
4	[37]	18.90	110312.5	4.3	67662.5	38.66
		1.910	117687.5	4.4	68262.5	42.00
5	[38]	9.14	99.5	5.1	86.25	13.32
		5.859	138.75	5.2	112	19.28
		19.02	139.25	5.3	104	25.31
6	[33]	813.41	1602.5	6.2	930	41.97
7	[6]	3020.52	470.25	7.3	434.5	7.60
8	[39]	5477.58	175.25	8.2	168	4.14
		12245.1	191.75	8.3	182.75	4.69
9	[9]	200.62*	119.5	9.1	110.75	7.32
		183.67*	120	9.2	116	3.33
10	[40]	4080.83	2810	9.1	1795	36.12
11	[14]	593.41	57.25	10.4	57.25	0.00
12	[41]	302.07*	265.25	11.5	245.25	7.54
13	[35]	200.11*	14412.75	12.6	14972.75	-3.89
14	[14]	293.86*	152.25	13.1	149.5	1.81

$*$: The layout of this problem was obtained by using the heuristic method presented in [13].

process routings, machine capacities, cell dimensions, etc. Due to the complexity of the problem, a GA was developed to efficiently solve it. Several numerical examples in different sizes were selected from the literature and the proposed GA was used to solve them. For small and medium-sized problems, the results of the GA were evaluated by a lower bound model and the results showed that the GA could solve the problems within a reasonable computational time and with an insignificant gap percent equal to 4.37% and 6.27%, respectively. Also, the solutions of GA were compared to those reported in the literature and the results indicated that the average improvement percent in the total material handling cost is equal to 16.06%. Finally, according to the results, it was concluded that when the decisions about routing of parts, CF and its layout are simultaneously made, the total material handling costs may decrease considerably.

Acknowledgments

The authors would like to thank the anonymous referee for his/her constructive comments and suggestions which, significantly improved the quality and scientific level of the paper.

References

1. Kusiak, A., *Intelligent Manufacturing Systems*, Prentice-Hall International Inc., Englewood Cllifs, NJ (1990).

2. Chiang, C.P. and Lee, S.D. "Joint determination of machine cells and linear intercell layout", *Computers and Operations Research*, **31**, pp. 1603-1619 (2004).

3. Arıkan, F. and Güngör, Z. "Modeling of a manufacturing cell design problem with fuzzy multi-objective parametric programming", *Mathematical and Computer Modelling*, **50**, pp. 407-420 (2009).

4. Wemmerlöv, U. and Hyer, N.L. "Procedures for the part family/machine group identification problem in cellular manufacture", *Journal of Operations Management*, **6**, pp. 125-147 (1986).

5. Akturk, M.S. "A note on the within-cell layout problem based on operation sequences", *Production planning and control*, **7**, pp. 99-103 (1996).

6. Ramabhatta, V. and Nagi, R. "An integrated formulation of manufacturing cell formation with capacity planning and multiple routings", *Annals of Operations Research*, **77**, pp. 79-95 (1998).

7. Tavakkoli-Moghaddam, R., Javadian, N., Javadi, B. and Safaei, N. "Design of a facility layout problem in cellular manufacturing systems with stochastic demands", *Applied Mathematics and Computation*, **184**, pp. 721-728 (2007).

8. Kia, R., Paydar, M.M., Jondabeh, M.A., Javadian, N. and Nejatbakhsh, Y. "A fuzzy linear programming approach to layout design of dynamic cellular manufacturing systems with route selection and cell reconfiguration", *International Journal of Management Science and Engineering Management*, **6**, pp. 219-230 (2011).

9. Adil, G.K. and Rajamani, D. "The trade-off between intracell and intercell moves in group technology cell formation", *Journal of Manufacturing Systems*, **19**, pp. 305-317 (2000).

10. Aktürk, M.S. and Turkcan, A. "Cellular manufacturing system design using a holonistic approach", *International Journal of Production Research*, **38**, pp. 2327-2347 (2000).

11. Saghafian, S. and Akbari Jokar, M.R. "Integrative cell formation and layout design in cellular manufacturing systems", *Journal of Industrial and Systems Engineering*, **3**, pp. 97-115 (2009).

12. Hicks, C. "A genetic algorithm tool for optimising cellular or functional layouts in the capital goods industry", *International Journal of Production Economics*, **104**, pp. 598-614 (2006).

13. Forghani, K., Arshadi-khamseh, A. and Mohammadi, M. "Integrated quadratic assignment and continuous facility layout problem", *International Journal of Industrial Engineering Computations*, **3**, pp. 787-806 (2012).

14. Chan, F.T.S., Lau, K.W., Chan, P.L.Y. and Choy, K.L. "Two-stage approach for machine-part grouping and cell layout problems", *Robotics and Computer-Integrated Manufacturing*, **22**, pp. 217-238 (2006).

15. Wu, X., Chu, C.H., Wang, Y. and Yan, W. "A genetic algorithm for cellular manufacturing design and layout", *European Journal of Operational Research*, **181**, pp. 156-167 (2007).

16. Jolai, F., Tavakkoli-Moghaddam, R., Golmohammadi, A. and Javadi, B. "An electromagnetism-like algorithm for cell formation and layout problem", *Expert Systems with Applications*, **39**, pp. 2172-2182 (2012).

17. Yalaoui, N., Mahdi, H., Amodeo, L. and Yalaoui, F. "A new approach for workshop design", *Journal of Intelligent Manufacturing*, **22**, pp. 933-951 (2011).

18. Car, Z. and Mikac, T. "Evolutionary approach for solving cell-formation problem in cell manufacturing", *Advanced Engineering Informatics*, **20**, pp. 227-232 (2006).

19. Heragu, S., *Facilities Design*, PWS Publishing Company, Boston, MA (1997).

20. Mungwattana, A. "Design of cellular manufacturing systems for dynamic and uncertain production requirements with presence of routing flexibility", Ph.D. Thesis, Faculty of the Virginia Polytechnic Institute and State University, Blackburg, VA (2000).

21. Sahni, S. and Gonzalez, T. "P-complete approximation problems", *Journal of the Association of Computing Machinery*, **23**, pp. 555-565 (1976).

22. Garey, M.R. and Johnson, D.S. "Computers and intractability: A guide to the theory of NP-completeness", *Freeman*, San Francisco, CA (1979).

23. Singh, N. "Design of cellular manufacturing systems: An invited review", *European Journal of Operational Research*, **69**, pp. 284-291 (1993).

24. Solimanpur, M. and Jafari, A. "Optimal solution for the two-dimensional facility layout problem using a branch-and-bound algorithm", *Computers and Industrial Engineering*, **55**, pp. 606-619 (2008).

25. Koopmans, T.C. and Beckmann, M.J. "Assignment problems and the location of economic activities", *Econometrica*, **25**, pp. 53-76 (1957).

26. Holland, J.H., *Adaptation in Natural and Artificial Systems, 2nd Edn.*, The University of Michigan Press, Ann Arbor, MI (1975).

27. Gen, M. and Cheng, R., *Genetic Algorithms and Engineering Design*, Wiley Interscience Publication, MA (1997).

28. Goldberg, D.E., *Genetic Algorithms: In Search, Optimization and Machine Learning*, Addison-Wesley, MA (1989).

29. Man, K.F., Tang, K.S. and Kwong, S., *Genetic Algorithms: Concepts and Design*, Springer, London (1999).

30. Filho, E.V.G. and Tiberti, A.J. "A group genetic algorithm for the machine cell formation problem", *International Journal of Production Economics*, **102**, pp. 1-21 (2006).

31. Mutingi, M. and Onwubolu, G.C. "Integrated cellular manufacturing system design and layout using group genetic algorithms", In *Manufacturing System*, F.A. Aziz, Ed., InTech-Open pp. 205-222 (2012).

32. Ghosh, T., Dan, P.K., Sengupta, S. and Chattopadhyay, M. "Genetic rule based techniques in cellular manufacturing (1992-2010): A systematic survey", *International Journal of Engineering, Science and Technology*, **2**, pp. 198-215 (2010).

33. Wang, X., *Reliability Consideration in the Design of Cellular Manufacturing Systems using Genetic Algorithm*, Ph.D. Thesis, Windsor, Ontario, Canada (2009).

34. Kao, Y. and Lin, C.H. "A PSO-based approach to

cell formation problems with alternative process routings", *International Journal of Production Research*, **50**, pp. 4075-4089 (2012).

35. Kazerooni, M., Luong, H.S. and Abhary, K. "A genetic algorithm based cell design considering alternative routing", *Computer Integrated Manufacturing Systems*, **10**, pp. 93-107 (1997).

36. Chung, S.H., Wu, T.H. and Chang, C.C. "An efficient tabu search algorithm to the cell formation problem with alternative routings and machine reliability considerations", *Computers and Industrial Engineering*, **60**, pp. 7-15 (2011).

37. Garbie, I.H., Parsaei, H.R. and Leep, H.R. "Machine cell formation based on a new similarity coefficient", *Journal of Industrial and Systems Engineering*, **1**, pp. 318-344 (2008).

38. Balakrishnan, J. and Jog, P.D. "Manufacturing cell formation using similarity coefficients and a parallel genetic TSP algorithm: Formulation and comparison", *Mathematical and Computer Modelling*, **21**, pp. 61-73 (1995).

39. Chang, P.T. and Lee, E.S. "Multisolution method for cell formation exploring practical alternatives in group technology manufacturing", *Computers and Mathematics with Applications*, **40**, pp. 1285-1296 (2000).

40. Yin, Y. and Yasuda, K. "Manufacturing cells' design in consideration of various production factors", *International Journal of Production Research*, **40**, pp. 885-906 (2002).

41. Dixit, A.R. and Mishra, P.K. "Heuristic based approach of cell formation considering operation sequence", *Proceedings of the World Congress on Engineering*, London (2007).

Biographies

Kamran Forghani obtained his BS degree in Industrial Engineering, in 2009, from the Islamic Azad University of Bonab, East Azarbaijan, Iran, and his MS degree from Kharazmi University, Tehran, Iran, in 2013, in the same subject. His research interests include design and planning of cellular manufacturing systems, facility layout, production planning, heuristics and metaheuristics.

Mohammad Mohammadi received his BS degree in Industrial Engineering from Iran University of Science and Technology, Tehran, Iran, in 2000, and MS and PhD degrees in Industrial Engineering from Amirkabir University of Technology, Tehran, Iran, in 2002 and 2009, respectively. He is currently Assistant Professor in the Department of Industrial Engineering at Kharazmi University, Tehran, Iran. His research and teaching interests include sequencing and scheduling, production planning, time series, metaheuristics and supply chains.