Journal of Manufacturing Systems 53 (2019) 32–48

Contents lists available at ScienceDirect

Journal of Manufacturing Systems

journal homepage: www.elsevier.com/locate/jmansys

Multi-objective particle swarm optimization for multi-workshop facility layout problem

Chao Guan[a,b], Zeqiang Zhang[a,b,*], Silu Liu[a,b], Juhua Gong[a,b]

[a] School of Mechanical Engineering, Southwest Jiaotong University, Chengdu 610031, China
[b] Technology and Equipment of Rail Transit Operation and Maintenance Key Laboratory of Sichuan Province, Chengdu 610031, China

ARTICLE INFO

Keywords:
Multi-workshop facility layout problem
Multi-objective optimization
Particle swarm optimization
Two-stage approach
Discrete framework

ABSTRACT

The novel multi-workshop facility layout problem presented in this paper involves the placement of a group of departments into several workshops; it deals with the distribution of departments over workshops and their optimal exact coordinates without overlapping. In considering practical situations, the internal material handling flows and external transport flows are taken into account in the problem. In this study, the problem is first formulated as a mixed integer linear programming model with three objectives: minimization of overall material handling costs, minimization of number of workshops, and maximization of utilization ratio of workshop floor. Thereafter, the proposed multi-objective particle swarm optimization algorithm with an innovative discrete framework and incorporated with a two-stage approach is employed to search for feasible solutions locally and globally. Finally, several benchmark instances derived from literature that satisfy our case requirements are employed to evaluate the performance of the proposed method; highly preferable results are typically achieved.

1. Introduction

The facility layout problem (FLP) refers to the most profitable physical arrangement of a set of given facilities with known dimensions within the manufacturing system. In production organizations, the configuration of facility layouts is particularly relevant to the material-handling system, which involves the resources demanded for manufacture or delivery of service [1]. Nowadays, industrial companies ameliorate the facility layout in order to efficiently utilize resources, decrease unnecessary movements, improve internal logistics transportation, and consequently reduce costs [2]. Previous research found that approximately 20%–50% of operating costs are constituted by the material handling activities; moreover, 10%–20% of that cost can be abated by a reasonable optimization of equipment layout [3,4].

In terms of practical applications, the FLP can be divided into three broad categories: row facility layout problem (RFLP), unequal-area facility layout problem (UA–FLP), and multi-floor facility layout problem (MFFLP) [1,5,6]. The classic FLP continues to be studied extensively because of the remarkable reduction in investment and operating costs that results from the adoption of a novel and efficient layout configuration.

In this study, an innovative multi-workshop facility layout problem (MWFLP) is investigated; it takes into consideration that a group of facilities (called departments in this work) can be arranged into more than one workshop. Moreover, the MWFLP involves two types of physical flows: internal material handling flows and external work-in-process transport flows. Internal flows, which are generally executed via the material feeding system in the workshop, occurs inside each workshop. On the other hand, there are critical external flows among workshops that are dependent on extra transport vehicles (e.g., trucks and trains). Therefore, the unit flow cost of the two different logistics mentioned above is different; it can be embodied in the mathematical model presented in this paper.

Because the MWFLP is classified with non-deterministic polynomial time (NP)-hard problem, most studies have resolved these problems by employing iterative meta-heuristic approaches, such as hybrid genetic algorithm (HGA) [7–10], simulated annealing (SA) algorithm [11,12], tabu search (TS) method [13], ant colony optimization (ACO) [14,15], bacterial foraging optimization (BFO) [2], particle swarm optimization (PSO) [3,16,17], and clonal selection algorithm (CSA) [18]. In this article, a multi-objective particle swarm optimization (MOPSO) is presented on the basis of our multi-objective model, with the minimization of overall material handling cost, minimization of number of workshops, and maximization of utilization ratio of workshop floor. Additionally, because of the dual characteristics of combinatorial optimization and continuous optimization of the specific layout problem,

* Corresponding author at: School of Mechanical Engineering, Southwest Jiaotong University, Chengdu 610031, China.
E-mail addresses: 17175371524@163.com (C. Guan), zhangzq@home.swjtu.edu.cn (Z. Zhang), liusilu_66@163.com (S. Liu), gongjuhua@my.swjtu.edu.cn (J. Gong).

https://doi.org/10.1016/j.jmsy.2019.09.004
Received 23 May 2019; Received in revised form 7 September 2019; Accepted 9 September 2019
0278-6125/

a two-stage optimization method is designed and combined with the MOPSO. In the first stage, a novel encoding measure is devised to produce feasible MWFLP layout solutions; thus, all binary variables in the proposed model can be uniquely determined. For the second stage, a simplified linear model is implemented and solved by CPLEX optimization software. Furthermore, several benchmark instances derived from literature that satisfy our case requirements are employed to evaluate the performance of MOPSO; highly preferable results are typically achieved.

The main contributions of our work are listed as follows:

• A novel multi-workshop facility layout problem is proposed here, which involves the arrangement of departments in several available workshops, besides, the internal and external logistics costs are also considered according to practical production.

• A mixed-integer linear programming model is formulated for this problem, and some instances can be solved to optimality within reasonable times.

• The individual particle encoding and placement strategy for our MWFLP are designed, which are implemented to represent and search for all feasible solutions.

• On the basis of the combinatorial nature and intrinsic difficulty of the problem, a multi-objective particle swarm optimization algorithm with an embedded two-stage approach is proposed. Additionally, comparison experiments with other methods indicate that the proposed algorithm can achieve highly preferable results. Finally, some MWFLP cases are solved by our proposed algorithm.

The next section of this article presents a brief review of related literature; subsequently, a more detailed description of the MWFLP is introduced. Moreover, a proposed mathematical model is formulated, and a case study is introduced. Finally, the two-stage optimization method and structure of MOPSO are described in detail; the remaining section presents numerical experiments and certain comparisons.

2. Literature review

The FLP is among the most crucial classical problems in production management and industrial engineering that has considerably attracted the interest of scholars since the 1960s. Numerous variations of the FLP have been considered and extensively investigated in literature, and a wide variety of formulations and objectives has been proposed for each variation. Meanwhile, various approaches have been successively presented in order to achieve effective solutions.

2.1. Facility layout problem and evaluation criteria

Facility layouts are directly influenced by the specifications of manufacturing systems, and several types of physical configurations are designed based on practical characteristics, such as product range and production volumes. To date, in existing literature, most research works focus on layouts in the single region [2,3,5–7,9,13–16,19–23]. In the product layout, a set of rectangular machines and equipment are assigned to several rows depending on the procedure of operations required for the product; this is called row facility layout problem [6,19,21]. Additionally, different from the aforementioned problem, the unequal area facility layout problem considers the obtainment of an excellent arrangement of given indivisible departments with a two-dimensional rectangle and diverse areas; all departments are located in the same fixed region with no overlapping. The unequal area FLP has received considerable interest since it was first proposed in 1963 [24], it had several variants from then on. For example, an unequal area FLP with a fixed size and shape has been conceptualized and investigated [7,13]; geometric constraints for confirming the relative location of departments have also been proposed to prevent overlapping. Moreover, a space-filling curve was utilized by connecting each unequal area department to be continuously arranged without partitions [22]; this technique was applied to manage the unequal area FLP with irregular departments. Furthermore, the design of a cellular manufacturing system (CMS) with an equal area FLP, which was a special case of the unequal area FLP [23], was presented. On the other hand, the multi-floor facility layout problem, which is relatively similar to the MWFLP in this article, extended the unequal area FLP to different floors except for a single building [5,23]. Recently, a novel multi-floor FLP was developed for real-life scenarios where a number of departments have to be assigned to a multi-floor building with a fixed room configuration; rooms under the same department were required to be adjacent [25]. A further simplification of the multi-floor FLP was presented: the departments were restricted to the same shape and assigned to special locations in the building [26]. Apart from the above, an FLP in which the departments were placed in a group of facilities was studied; subsequently, because of its complexity, it was simplified into a quadratic assignment problem [27,28].

Classically, a layout design should be evaluated from different perspectives; in this regard, the optimization criteria in the FLP literature vary in form. The material handling cost minimization, which is the most critical indicator for evaluating the performance of a layout, is the most popular criterion in studies [2]. Additionally, minimizing the re-layout cost is another type of objective that is considered a necessity in dynamic layout design [2,16,23,27]. Other objectives that are studied in existing research also include minimizing the distance travelled [29,30], minimizing work in the process [31], maximizing satisfaction [32], maximizing adjacency/closeness [32,33], and maximizing profits [34,35]. In this work, two other functions are considered to measure the performance of the existing MFFLP layouts: the number of workshops required to build for production and the utilization ratio of workshop floor.

2.2. Staged approaches for facility layout problem

It has been proven that the FLP is an NP-hard problem in general [1,3,36]; hence, obtaining a global optimal solution within a reasonable time is difficult. Actually, numerous restricted FLPs, in which the fixed dimensions of the facilities are all equal and only the assignment of facilities to designated locations is considered, are known as combinatorial optimization problems, which can be abstracted into discrete quadratic programming formulations.

Furthermore, for some FLPs that merge the combinatorial nature and characteristics of continuous optimization, the traditional exact methods, such as branch-and-bound [37] and dynamic programming [38,39], are unavailable for finding optimal results or even feasible solutions because of their high computational cost. Therefore, several multi-stage frameworks are presented to handle the composite optimization FLPs. For example, Wang et al. [40] employed a methodology that combines the SA with mathematical programming to resolve the dynamic double-row layout problem (DDRLP). In the first stage, the sequence and locations of departments were designed; then, the use of SA was suggested to obtain the solution to the DDRLP. Actually, multi-stage approaches are frequently applied to manage the multi-floor FLP, as follows: in the first stage, the departments are generally distributed over the floors; in the second stage, reasonable relative positions are identified for the departments in each floor; in the last stage, no overlap is enforced, and the exact coordinates of departments are determined [41–43].

2.3. PSO and multi-objective processing methods

In this work, the multi-objective particle swarm optimization (MOPSO) is modified and employed to solve the multi-objective MWFLP. This is because the MOPSO is not only easy to implement where there are few parameters to adjust, but it has good convergence speed and is also one of the successfully established solution approaches to multi-objective optimization problems. Liu et al. [3] proposed an MPSO algorithm that combines the objective space division technique to optimize the multi-objective UAFLP. Asl and Wong [16] formulated multi-objective models for unequal area static and dynamic facility layout problems; therefore, a modified PSO was suggested to solve them. Onut et al. [17] conceived a PSO to handle the multiple-level warehouse layout design problem; the PSO was able to obtain near optimal results in a short time. Zhang et al. [44] developed a new barebone MOPSO algorithm to solve the environment/economic dispatch problems, this algorithm was able to expand the search capability without tuning up control parameters and a technique was introduced to handle unfeasible solutions. Zhang et al. [45] proposed a multi-swarm cooperative multi-objective particle swarm optimizer (MC-MOPSO), which consisted of multiple slave and master swarms to improve the performance of MOPSO. Mousavi et al. [46] examined a two-echelon distribution supply chain network by presenting a modified PSO to find the optimal location of manufacturers and retailers. Except for the mentioned above, many new variants of MOPSO are investigated in recent years and are widely applied in many fields, such as cost-based feature selection [47,48], robot path planning [49], equity portfolio management [50], etc. Accordingly, in view of the extensive applications mentioned above, it is appropriate to adopt the PSO based on the nature of the MWFLP.

Extensive research concerning the multiple optimization objectives of the FLP as a multi-objective optimization problem was implemented to handle this type of problem effectively. In the existing literature, the lexicographic ordering approaches are adopted to optimize each objective gradually [36,51–53]; however, it only optimizes some of the objectives and ignores others. On the other hand, the weighted sum approach is another common method employed [27,54–56]; it straightforwardly transforms the multi-objective problem into one that has a single objective by assigning weight values to each objective. However, the balance among objectives is difficult to achieve because the weight values are based on the subjective judgment of experts. Additionally, the Pareto solution set, which can acquire a series of balanced solutions that focuses on different objectives, is widely implemented and popular in decision-making [3,15,25,33,35,57]. Based on the foregoing method, in order to preserve the population diversity and convergence to the Pareto front, the crowding distance strategy proposed by Deb et al. [58] is applied in our algorithm. Compared with other representative methods in the existing literature such as clustering approach [59], the adaptive grid approach [60], the decomposition-based archiving approach [61], the ε-dominance-based approach [62], and so on, it has been widely employed and was easy to implement for multidimensional optimization problems to maintain the diversity of the external archives.

3. Problem description and exact formulation

3.1. Multi-workshop facility layout problem

The extension of the proposed FLP mentioned above involves the collection of fixed-shaped departments to multiple workshops, in which overlaps among departments are forbidden in the horizontal and vertical directions. Moreover, all the workshops have the same size and interactive position via which the resources flow among different workshops. Apart from the foregoing, based on the engineering of the manufacturing factory, the internal and external flows require different transportation expenses, and the external flows are extremely costly because trucks or forklifts are used. More detailed descriptions pertaining to the MWFLP are illustrated in Fig. 1; the figure indicates that 10 departments are allocated, and the interactive flows inside and outside the site are considered. Being similar with the previous research, the resource interaction point for each department is assumed to be situated in the centroid of the rectangle, and the lower left corner of each workshop is associated with a single location for entry and exit. Additionally, because of the horizontal and vertical walking paths of the automatic guided vehicle (AGV), the distance between two departments is the Manhattan distance. For convenience in describing the flow of communications among departments, Fig. 1 depicts two resource transportation modes. The internal flows between departments D-1 and D-2 are generated by means of material handling conveyances (such as the AGV), which are appropriate for conveying goods inside the workshop. On the other hand, transport conveyances, such as trucks and forklifts, are employed to carry products among workshops. The distance travelled by conveyances between departments D-1 and D-8 is measured from 0_g to 0_t; the remaining movements from D-1 to D-8 involve the internal flows generated by means of the AGV. It should be noted that in our MWFLP, it is necessary to optimize the precise location of each department in order to minimize the weighted sum of distances among departments. Moreover, the number of workshops built for departments is necessarily considered to be reduced, then the costs of establishment of all foundations are saved; besides, an adequate concurrent workshop floor utilization is suggested, which is used to minimize and balance the area of envelop rectangle among workshops, thus there will be more available space for storage, transportation and rest.

Moreover, as is shown in Fig. 1, the workshops placed along a straight line can be regarded as a special case of the single-row equidistant facility layout problem [19]; the major difference among them is that the number of workshops is not definite, and the allocation of given departments in each workshop must be considered. Moreover, note that the proposed problem has certain similarities with the multi-floor facility layout problem; each workshop can be regarded as the floor layout in the MFFLP, and the lower left corner can be treated as elevators. However, compared with relevant MFFLP research [6,25,41], the various transportation expenses of internal and external flows are taken into account in this paper, and the number of workshops or floors are not affirmatory beforehand. Thus, the MWFLP is a more popular and complex case of the FLP.

3.2. Mathematical formulation

Based on the characteristics of the proposed problem and considering the convenience of description, an independent coordinate system is defined for each workshop: the lower left corner is set as the origin and is used as reference to calculate the positions of centroids of departments; the X and Y axes coincide with the workshop length and width edges, respectively. Accordingly, the constraints (such as no department overlapping, each department can be arranged only in one workshop, and the department size fits into the workshop dimensions) can be easily constructed as presented in this section.

First, certain parameters and variable notations are defined; subsequently, the mixed integer linear programming (MILP) mathematical model of the MWFLP is formulated.

Parameters and indices

n:	number of departments
m:	maximum number of available workshops
$\boldsymbol{N}$:	set of given n departments ($\boldsymbol{N} = \{1, 2, ..., n\}$)
$\boldsymbol{M}$:	set of all numbers of given m available workshops ($\boldsymbol{M} = \{1, 2, ..., m\}$)
i, j, s:	indices for departments
k, g, t:	indices for available workshops
L_F:	length of workshop floor
W_F:	width of workshop floor
l_i:	length of ith department
w_i:	width of ith department
p_{ij}:	material flow value between departments i and j
f_{ij}:	frequency of material flow between departments i and j
$cInt$:	handling cost of unit material per unit distance inside workshops
$cEnt$:	transport cost of unit resources per unit distance among workshops

Decision variables

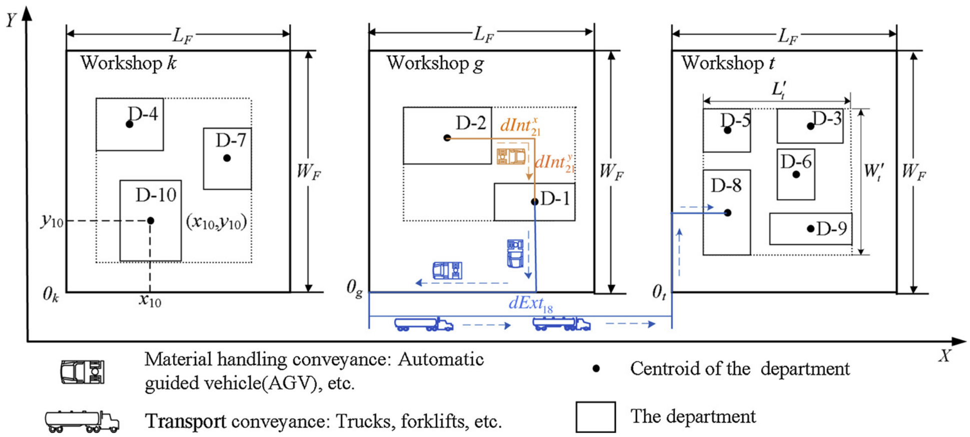


Fig. 1. Schematic of multi-workshop facility layout problem.

L'_k: length of envelope rectangle, which can obtain all departments that are assigned to kth workshop

W'_k: width of envelope rectangle, which can obtain all departments that are assigned to kth workshop

$dExt_{ij}$: Manhattan distance travelled by transport conveyances outside workshops between departments i and j

$dInt^x_{ij}$: distance between departments i and j projected in X direction that is covered by material handling conveyances

$dInt^y_{ij}$: distance between departments i and j projected in Y direction that is covered by material handling conveyances

x^k_i: X-coordinate of centroids of departments i, which are assigned to kth workshop

y^k_i: Y-coordinate of the centroid of departments i, which are assigned to kth workshop

α^k_{ij}, β^k_{ij}: sequence-pair variables that are used to determine the relative positions of departments; for any departments i and j, the following conditions hold:
if $\alpha^k_{ij}=1$ and $\beta^k_{ij}=1$, then i precedes j horizontally;
if $\alpha^k_{ij}=0$ and $\beta^k_{ij}=0$, then j precedes i horizontally;
if $\alpha^k_{ij}=1$ and $\beta^k_{ij}=0$, then i precedes j vertically;
if $\alpha^k_{ij}=0$ and $\beta^k_{ij}=1$, then j precedes i vertically.

$S_k=\begin{cases}1, \text{if workshop } k \text{ is employed}\\ 0, \text{otherwise}\end{cases}$

$Y^k_i=\begin{cases}1, \text{if department } i \text{ is assigned in workshop } k\\ 0, \text{otherwise}\end{cases}$

$q^k_{ij}=\begin{cases}1, \text{if departments } i \text{ and } j \text{ are placed in workshop } k \text{ together}\\ 0, \text{otherwise}\end{cases}$

In terms of the above notations, the MILP formulation of the MWFLP can be expressed as follows.

Objective functions

$$F_{MHC}=\min\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}p_{ij}\cdot f_{ij}\cdot cInt\cdot(dInt^x_{ij}+dInt^y_{ij})+\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}p_{ij}\cdot f_{ij}\cdot cExt\cdot dExt_{ij} \tag{1}$$

$$F_{NWS}=\min\sum_{k=1}^{m}S_k \tag{2}$$

$$F_{UR}=\min\sqrt{\sum_{k=1}^{m}\left(\frac{L'_k\cdot W'_k}{L_F\cdot W_F}\right)^2/n} \tag{3}$$

The three primary objectives we focus on are as formulated above; these duly reflect the multi-objective nature of the MWFLP. Objective function (1) is developed to measure the overall material handling costs, which is composed of internal and external transportation costs. Objective function (2) minimizes the number of available workshops required for departments. Objective function (3) is presented to optimize the the utilization ratio of the shop floor; it involves the use of the workshop floor occupied by departments. Given the foregoing, the interrelated constraints of the proposed MILP model can be formulated as follows:

Subject to

$$(m-1)\cdot L_F\cdot(2-Y^k_i-Y^g_j)+dExt_{ij}\geq(k-g)\cdot L_F,\ 1\leq i<j\leq n;\ \forall\ k,g\in\boldsymbol{M};\ k\neq g \tag{4}$$

$$(m-1)\cdot L_F\cdot(2-Y^k_i-Y^g_j)+dExt_{ij}\geq(g-k)\cdot L_F,\ 1\leq i<j\leq n;\ \forall\ k,g\in\boldsymbol{M};\ k\neq g \tag{5}$$

$$L_F\cdot(1-\sum_{t=1}^{m}q^t_{ij})+dInt^x_{ij}\geq x^k_i-x^k_j,\ 1\leq i<j\leq n;\ k\in\boldsymbol{M} \tag{6}$$

$$L_F\cdot(1-\sum_{t=1}^{m}q^t_{ij})+dInt^x_{ij}\geq x^k_j-x^k_i,\ 1\leq i<j\leq n;\ k\in\boldsymbol{M} \tag{7}$$

$$W_F\cdot(1-\sum_{t=1}^{m}q^t_{ij})+dInt^y_{ij}\geq y^k_i-y^k_j,\ 1\leq i<j\leq n;\ k\in\boldsymbol{M} \tag{8}$$

$$W_F\cdot(1-\sum_{t=1}^{m}q^t_{ij})+dInt^y_{ij}\geq y^k_j-y^k_i,\ 1\leq i<j\leq n;\ k\in\boldsymbol{M} \tag{9}$$

$$2\cdot L_F\cdot\sum_{t=1}^{m}q^t_{ij}+dInt^x_{ij}\geq x^k_i+x^g_j,\ 1\leq i<j\leq n;\ \forall\ k,g\in\boldsymbol{M};\ k\neq g \tag{10}$$

$$2\cdot W_F\cdot\sum_{t=1}^{m}q^t_{ij}+dInt^y_{ij}\geq y^k_i+y^g_j,\ 1\leq i<j\leq n;\ \forall\ k,g\in\boldsymbol{M};\ k\neq g \tag{11}$$

$$dInt^x_{ij},\ dInt^y_{ij}\geq 0,\ 1\leq i<j\leq n \tag{12}$$

$$dExt_{ij}\geq 0,\ 1\leq i<j\leq n \tag{13}$$

As mentioned above, constraints (4)–(13) determine the horizontal and vertical distances among various departments. As for constraints (4) and (5), the distance between two adjacent workshops is assumed to be L_F. Thus, the translation path from 0_k to 0_g is $|k-g|\cdot L_F$; it can be linearized as constraints (4) and (5). Constraints (6)–(9) define the distances in the horizontal and vertical directions among departments that are assigned to the same workshop. Finally, for the departments that are assigned to separate workshops, the distances among them can be measured by constraints (10) and (11).

$$\sum_{k=1}^{m}Y^k_i=1,\ \forall\ i\in\boldsymbol{N} \tag{14}$$

$$S_{k-1}\geq S_k,\ \forall\ k\in\{2,3,\ \ldots,m\} \tag{15}$$

$$\sum_{i=1}^{n} Y_i^k \geq S_k, \forall k \in \boldsymbol{M} \tag{16}$$

$$S_k \cdot n \geq \sum_{i=1}^{n} Y_i^k, \forall k \in \boldsymbol{M} \tag{17}$$

$$q_{ij}^k \leq \frac{1}{2} \cdot (Y_i^k + Y_j^k), 1 \leq i < j \leq n; \forall k \in \boldsymbol{M} \tag{18}$$

$$q_{ij}^k + 1 \geq Y_i^k + Y_j^k, 1 \leq i < j \leq n; \forall k \in \boldsymbol{M} \tag{19}$$

$$S_k \in \{0, 1\}, \forall k \in \boldsymbol{M} \tag{20}$$

$$Y_i^k \in \{0, 1\}, \forall i \in \boldsymbol{N}; \forall k \in \boldsymbol{M} \tag{21}$$

$$q_{ij}^k \in \{0, 1\}, 1 \leq i, j \leq n; i \neq j; \forall k \in \boldsymbol{M} \tag{22}$$

In constraints (14)–(22), the binary variables (Y_i^k, S_k and q_{ij}^k) are confirmed; these involve the distribution of a given number of departments. Constraints (14) guarantee that each department is only arranged to one workshop. Constraints (15)–(17) ensure that the binary values of S_k are attached to Y_i^k. Constraints (15) guarantee that the workshops are sequentially employed; this prevents having an empty workshop in the layout. Furthermore, the relationship between S_k and Y_i^k are expressed by constraints (16) and (17). Note that q_{ij}^k, which is defined to simplify the expressions of various distances, should satisfy constraints (18) and (19).

$$\frac{1}{2} \cdot l_i \cdot Y_i^k \leq x_i^k, \forall i \in \boldsymbol{N}; \forall k \in \boldsymbol{M} \tag{23}$$

$$x_i^k \leq (L_F - \frac{1}{2} \cdot l_i) \cdot Y_i^k, \forall i \in \boldsymbol{N}; \forall k \in \boldsymbol{M} \tag{24}$$

$$\frac{1}{2} \cdot w_i \cdot Y_i^k \leq y_i^k, \forall i \in \boldsymbol{N}; \forall k \in \boldsymbol{M} \tag{25}$$

$$y_i^k \leq (W_F - \frac{1}{2} \cdot w_i) \cdot Y_i^k, \forall i \in \boldsymbol{N}; \forall k \in \boldsymbol{M} \tag{26}$$

As indicated by constraints (23)–(26), it is necessary for the coordinates of the centroids of departments to satisfy these constraints. Based on the coordinate system we developed, the ranges of the abscissa and ordinate of department i, which is assigned to workshop k, are limited in order to guarantee that this department is located inside the workshop with no overlapping.

$$\alpha_{ij}^k + \alpha_{ji}^k = 1, 1 \leq i < j \leq n; \forall k \in \boldsymbol{M} \tag{27}$$

$$\beta_{ij}^k + \beta_{ji}^k = 1, 1 \leq i < j \leq n; \forall k \in \boldsymbol{M} \tag{28}$$

$$\alpha_{is}^k + \alpha_{sj}^k - \alpha_{ij}^k \leq 1, \forall i, j, s \in \boldsymbol{N}; \forall k \in \boldsymbol{M} \tag{29}$$

$$\beta_{is}^k + \beta_{sj}^k - \beta_{ij}^k \leq 1, \forall i, j, s \in \boldsymbol{N}; \forall k \in \boldsymbol{M} \tag{30}$$

$$\alpha_{ij}^k, \beta_{ij}^k \in \{0, 1\}, 1 \leq i, j \leq n; i \neq j; \forall k \in \boldsymbol{M} \tag{31}$$

Constraints (27)–(31) in this model are frequently applied to the UA–FLP for confirming the relative positions of departments and representing valid sequences. A pair of variables, α_{ij}^k and β_{ij}^k, encode the positional relationship. As mentioned above, constraints (27) and (28) ensure that each department appears exactly once when used, and constraints (29)–(30) are transitivity constraints for the two pairs of variables [1].

$$x_i^k + \frac{1}{2} \cdot l_i \cdot Y_i^k \leq x_j^k - \frac{1}{2} \cdot l_j \cdot Y_j^k + L_F \cdot (2 - \alpha_{ij}^k - \beta_{ij}^k), \forall i, j \in \boldsymbol{N}; i \neq j; \forall k \in \boldsymbol{M} \tag{32}$$

$$y_i^k + \frac{1}{2} \cdot w_i \cdot Y_i^k \leq y_j^k - \frac{1}{2} \cdot w_j \cdot Y_j^k + W_F \cdot (1 + \alpha_{ij}^k - \beta_{ij}^k), \forall i, j \in \boldsymbol{N}; i \neq j; \forall k \in \boldsymbol{M} \tag{33}$$

$$(1 - q_{ij}^k) \cdot L_F + L'_k \geq (x_i^k - x_j^k) + \frac{1}{2} \cdot (l_i + l_j), \forall i, j \in \boldsymbol{N}; \forall k \in \boldsymbol{M} \tag{34}$$

$$(1 - q_{ij}^k) \cdot W_F + W'_k \geq (y_i^k - y_j^k) + \frac{1}{2} \cdot (w_i + w_j), \forall i, j \in \boldsymbol{N}; \forall k \in \boldsymbol{M} \tag{35}$$

$$L'_k, W'_k \geq 0, \forall k \in \boldsymbol{M} \tag{36}$$

In view of variables α_{ij}^k and β_{ij}^k, constraints (32) and (33) are employed to prevent the overlapping of departments in the horizontal and vertical directions when these are allocated in the same workshop. As for constraints (32), if department i is on the left of department j, and the value of α_{ij}^k and β_{ij}^k together is 1, then the minimum interval between i and j should be $(l_i + l_j)/2$. Similarly, if department j precedes i vertically, then $\alpha_{ij}^k = 0$ and $\beta_{ij}^k = 1$; the distance between two centroids should not be less than $(w_i + w_j)/2$, as presented in constraints (33). The envelope rectangle of kth workshop can be determined by constraints (34) and (35).

In general, it is evident that the MILP model built for the MWFLP is also able to represent the UA-FLP with a fixed shape; it is only required to set the value of parameter m to 1. Furthermore, the model in this study can also be regarded as a special MFFLP, which assign the shape-fixed departments to several floors; the heights of adjacent floors are considered to be the same. Moreover, the handling costs consisting of internal and external flows are separately taken into account; this is more realistic and practical.

3.3. Case study

In this section, several trials are performed to evaluate the effectiveness of our MILP model; moreover, certain benchmark instances are applied and analyzed for the newly defined problem. However, the lack of test problems that satisfy our case requirements leads us to find benchmark instances that can be employed for the UA–FLP with some parameter modifications. Accordingly, seven benchmark instances are used for optimizing objectives (1)–(3), as listed in Table 1. In the first column in Table 1, the digits indicate the number of departments in a particular instance; the literature cited are listed in the second column. Detailed data including the number of available workshops, dimensions of each workshop floor, and cost of internal and external flows are also listed in Table 1. Moreover, the additional matrix of flow frequency is created for these instances and attached to the appendix.

As mentioned above, all trials are performed with IBM ILOG CPLEX V12.8 optimization software using the pre-processing values; and executed in Windows 10 environment on a desktop PC configured with Intel(R) Core(TM) i5-7400 CPU E6700 3.2-GHz and 4-GB RAM. Finally, accurate results are derived and summarized in Table 2. It is noteworthy that objective (3) in Section 3.2 involves a quadratic term in the formula; it is also nonconvex and cannot be managed via CPLEX. Accordingly, Table 2 only lists the optimal solutions to the two other objectives; the optimal layouts and coordinates of each department are

Table 1
Case study data.

Instance name	Reference	n	m	L_F	W_F	cInt	cEnt
Das04	Das [63]	4	4	20	30	2	5
Das06	Liu et al. [3]	6	6	25	25	4	10
Das08	Liu et al. [3]	8	8	40	30	7	9
SFLP08	Asl and Wong [16]	8	8	7	6	1	5
FLP09	Zhang et al. [64]	9	9	5	4	4	9
Das10	Liu et al. [3]	10	5	40	30	6	13
SFLP11	Asl and Wong [16]	11	5	8	8	10	21

Table 2
Computational results of seven benchmark instances obtained by CPLEX.

Instance name	MWFLP	Objective function values			Solutions from proposed MILP model		
		F_{MHC}	F_{NWS}	F_{UR}	Optimal assignment for workshop layout	Location coordinates $\{(\sum_{k=1}^{m} x_i^k, \sum_{k=1}^{m} y_i^k)\}$, $i = 1, 2, \ldots, n-1, n$	Running time (s)
Das04	(a)	9130.00	2	0.5069	{1, 2, 4}; {3}	{(10, 5), (10, 12.5), (10, 22.5)}; {(5, 5)}	0.11
	(b)	16 164.00	2	0.5973	{2, 4}; {1, 3}	{(15.5, 27.5), (10, 7.5)}; {(9, 5), (5, 25)}	0.05
Das06	(a)	42 944.00	2	0.3720	{1, 3, 4, 5, 6}; {2}	{(5, 4), (16, 13), (16, 4), (5, 18), (5, 11)}; {(10, 7.5)}	1.97
	(b)	80 782.00	2	0.5031	{4}; {1, 2, 3, 5, 6}	{(19, 21)}; {(5, 4), (15, 9.5), (16, 19), (14, 4), (20.5, 11)}	0.17
Das08	(a)	110 893.50	2	0.3322	{1, 2}; {3, 4, 5, 6, 7, 8}	{(7, 5), (7, 13.5)}; {(5, 6.5), (28, 7.5), (27, 21.5), (14, 7.5), (13.5, 14.5), (10.5, 23)}	109.66
	(b)	241 989.00	2	0.3830	{2, 3, 5, 6, 7, 8}; {1, 4}	{(5, 11.5), (35, 16), (19, 23.5), (12, 4), (35.5, 3), (23.5, 5.5)}; {(7, 20), (10, 7.5)}	0.89
SFLP08	(a)	6018.00	4	0.4842	{7}; {1, 3, 4, 6}; {2, 8}; {5}	{(2, 2)}; {(1, 4.5), (3, 5), (1.5, 1.5), (5, 2),}; {(5, 2.5), (1.5, 2)}; {(1, 2)}	87.11
	(b)	7415.00	3	0.5602	{1, 3, 4, 6}; {5, 7}; {2, 8}	{(6, 4.5), (1, 1), (5.5, 1.5), (2, 4)}; {(1, 4), (4, 4)}; {(5, 2.5), (1.5, 2)}	0.67
FLP09	(a)	11 863.80	5	0.5742	{8}; {4, 9}; {1, 2, 3, 5}; {6}; {7}	{(2.5, 1.5)}; {(1, 0.9), (3.5, 2)}; {(3.95, 2.4), (1.5, 2.5), (1, 0.5), (3.25, 0.75)}; {(1.5, 1.5)}; {(1.5, 1.4)}	2760.42
	(b)	20 908.00	5	0.6809	{1, 2, 3, 5}; {4, 6}; {9}; {8}; {7}	{(0.95, 2.1), (3.5, 3), (1, 3.5), (3.75, 1.25)}; {(1, 3.1), (3.5, 2.5)}; {(3.5, 2)}; {(2.5, 2.5)}; {(1.5, 1.4)}	12.47
Das10	(a)	289 462.00	3	0.3526	{2}; {1, 3, 4, 5, 6, 8, 9}; {7, 10}	{(6, 4)}; {(7.5, 25), (27.5, 6), (6.5, 15.5), (15.5, 9), (27.5, 21), (16, 3.5), (5.5, 5.5)}; {(10, 21), (9.5, 6.5)}	706.72
	(b)	518 600.00	2	0.4586	{2, 3, 5, 7, 10}; {1, 4, 6, 8, 9}	{(25, 4), (32.5, 24), (23.5, 11), (15, 22), (9.5, 7.5)}; {(7.5, 19), (4.5, 9.5), (27.5, 20), (4, 2.5), (14.5, 5.5)}	0.53
SFLP11	(a)	174 132.00	3	0.3792	{7, 11}; {1, 2, 3, 4, 5, 6, 8, 9}; {10}	{(2, 6.5), (2.5, 2.5)}; {(2, 6), (3.5, 1), (3.5, 3), (6.4, 2.5), (1.5, 1), (4.7, 2.5), (1.7, 3), (6, 6.4)}; {(2, 3.5)}	2996.55
	(b)	288 906.00	3	0.5039	{1, 2, 3, 4, 5, 6, 8}; {7, 9, 11}; {10}	{(4.6, 4), (0.5, 1), (7.5, 7), (1, 4.5), (5.1, 1), (7.3, 3.5), (5.7, 7)}; {(6, 6.5), (2, 6.4), (5.5, 2.5)}; {(2, 3.5)}	49.00

listed in columns 6 and 7, whereas column 8 lists the runtimes of instances. In Table 2, symbols (a) and (b) indicate the MWFLP criterion with the best material handling cost and the MWFLP criterion with the minimum number of workshops, respectively.

It can be observed in Table 2 that the departments are divided into several workshops, and the parameters that are set make it impossible to allocate all departments to only one workshop. On the other hand, the runtime rapidly increases with the scale of benchmark instances; this is one of the characteristics of combinatorial optimization problems. In the real-life situations, compromising plans are also required such that decision-makers can choose their preferred layout. Thus, in order to solve the MWFLP more effectively, the MOPSO is proposed and described with an embedded two-stage optimization frame.

4. Methodology for multi-workshop facility layout problem

Notice that certain sub-problems are required to handle our proposed problem. One key point is that it is necessary to determine the number of workshops available for layout before the departments are assigned to them; this renders the problem difficult to handle through exact methods because of the combinatorial explosion of allocations. Hence, the meta-heuristic MOPSO is proposed.

In this paper, a novel discrete framework for the evolutionary computation of each particle is presented; the emulating mechanism of each particle to the global optimal swarm (called Gbest) and to the best position in the past of every particle (called Pbest) is redefined for the multi-workshop facility layout problem. Additionally, a two-stage approach is incorporated to search for feasible solutions; the Pareto principle is applied to handle the multi-objective nature of the problem.

4.1. Individual particle encoding

According to the characteristics of the MWFLP, the simple sequences of department indices cannot be employed to represent intact and feasible layouts of the problem that, do not specify the information concerning department arrangement and exact locations of departments. On the other hand, different from the established bay structure developed in [65] and the flexible bay structure [66] that are frequently deployed to define the solutions to UA–FLP/MFFLP only when the aisles or floors are predefined and the shape unfixed, the constructive approach for our MWFLP should indicate all possible combinations regardless of the number of workshops employed. Meanwhile, reasonable positions for the departments that satisfy the formulations mentioned in Section 3.2 are supposed to be presented in encoding.

Therefore, as depicted in Fig. 2, the solutions to the problem can be encoded by vectors with four segments. The first n elements represent the identifier of departments to be laid out; these are used to obtain the placing sequence in Segment 1. Moreover, Segment 2 is used to obtain the number of departments assigned to each workshop. For example, if $\Gamma_k = 4$, then the departments from $\sum_{l=1}^{k-1} \Gamma_l\ +1$ to $\sum_{l=1}^{k} \Gamma_l$ in Segment 1 are to be allocated to the kth workshop, and the number of operational workshops in this layout will be $\sum_{k=1}^{m} S_k$. As for Segment 3, based on the allocation plans of Segment 2, the relative positions for every pair of departments are determined and shown in this segment. Finally, Segment 4 is used to obtain the two-dimensional coordinates of each department that are subject to the constraints in this study.

4.2. Placement strategy

For each individual particle, decoding the placing sequence can be accomplished by using the placement strategy described in this section. While attempting to place the departments into the workshops, the number of workshops used and the relative positions of pairs of departments assigned to the same workshop are noted; thus, Segments 2 and 3 can be acquired according to this strategy.

The placement strategy follows a sequential process, which arranges

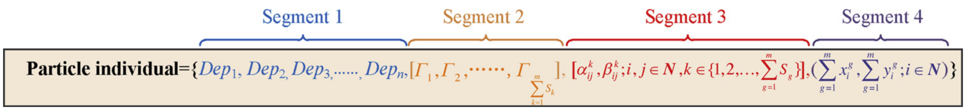
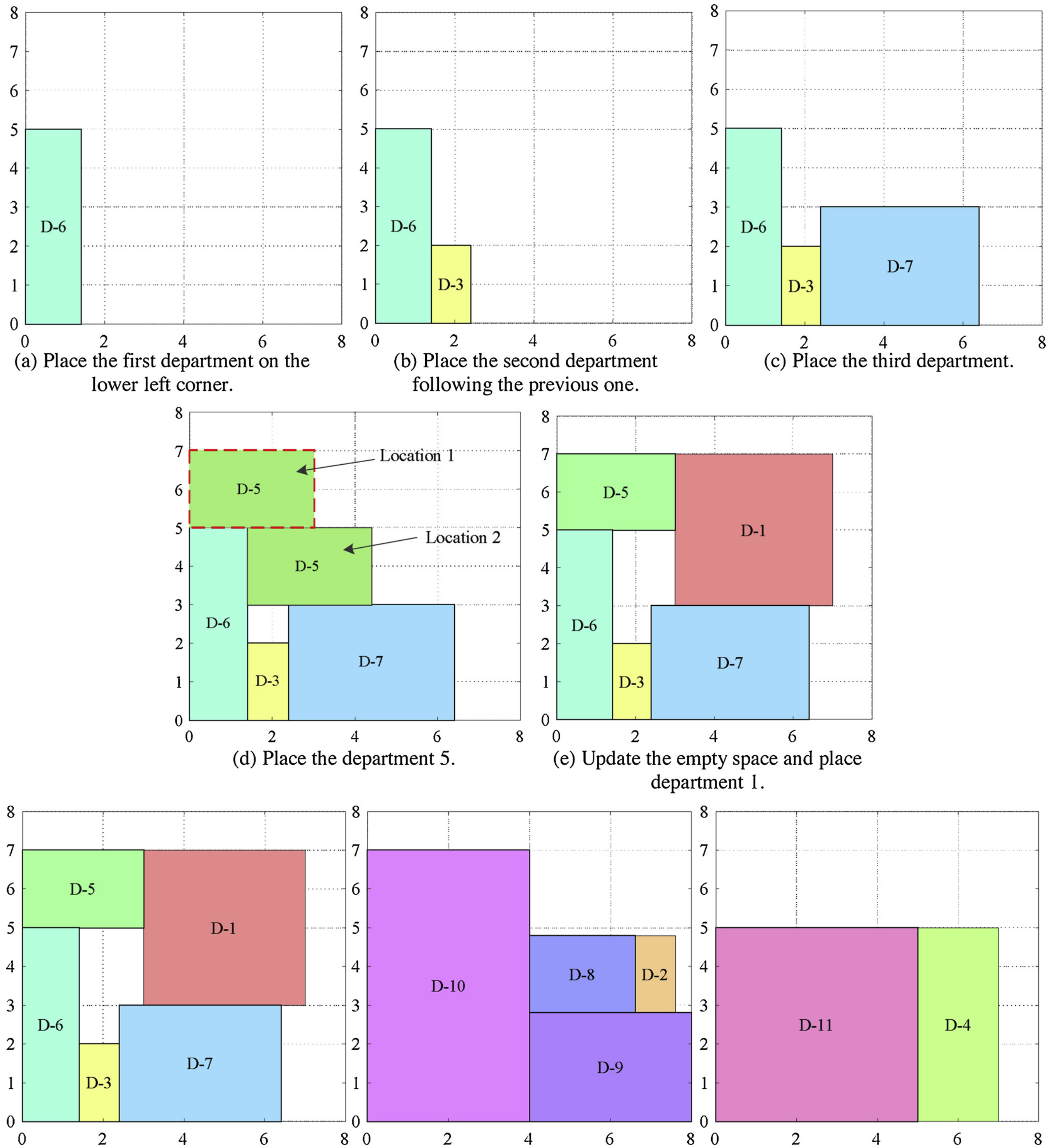


Fig. 2. Individual particle encoding.



(a) Place the first department on the lower left corner.

(b) Place the second department following the previous one.

(c) Place the third department.

(d) Place the department 5.

(e) Update the empty space and place department 1.

(f) Final result obtained using the placement strategy.

Fig. 3. Placement strategy example on decoding individual particles.

a single department at every step. The selected department is obtained sequentially according to the placing sequence in Segment 1. In the initial stage, the first workshop is employed, and the entire floor space is available; thereafter, the first single department is placed in the lower left corner of the workshop. To generate feasible solutions to the problem, the remaining empty spaces are updated after every placement of a single department. Accordingly, it is only necessary to find the available empty space where the department being placed fits; the next department is placed immediately along the *X*-axis. Moreover, once there is no adequate empty space available for the following department, a new workshop is to be used.

An example of the special placing process is shown in Fig. 3, where benchmark instance SFLP11 mentioned in Section 3.3 is used; the placing sequence is $\boldsymbol{Sol}=\{6, 3, 7, 5, 1, 10, 9, 8, 2, 11, 4\}$. As depicted in Fig. 3a, initially, the first department, 6, is placed on the lower left corner of the workshop. Therefore, the next departments, 3 and 7, are placed sequentially close to the previous one, and their relative positions to each other are determined concurrently. Department 5 can be placed above departments 3 and 7 or on top of this aforementioned location (locations 1 and 2 in Fig. 3d) because there is no available empty space for it if it is placed behind department 7. As for the placement of department 1, another workshop is required for it if department 5 is placed in location 2; this may make three objectives inferior because an extra workshop is used, and additional distances among the departments have to be travelled. Thus, department 5 is placed in location 1, and the first five departments are arranged in one workshop, as shown in Fig. 4e. Department 10 is placed in the corner of another workshop, and the rest are placed following the arrangement mentioned above.

The final result of the aforementioned example is illustrated in Fig. 3f. In this example, 11 departments are distributed to three workshops, and the individual particles can be represented as {6, 3, 7, 5, 1, 10, 9, 8, 2, 11, 4, [5,4,2],

$$\alpha_{ij}^1=\begin{bmatrix} Dep_i & 6 & 3 & 7 & 5 & 1 \\ 6 & 0 & 1 & 1 & 0 & 1 \\ 3 & 0 & 0 & 1 & 0 & 0 \\ 7 & 0 & 0 & 0 & 0 & 0 \\ 5 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}, \quad \beta_{ij}^1=\begin{bmatrix} Dep_i & 6 & 3 & 7 & 5 & 1 \\ 6 & 0 & 1 & 1 & 1 & 1 \\ 3 & 0 & 0 & 1 & 0 & 1 \\ 7 & 0 & 0 & 0 & 0 & 1 \\ 5 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}; \quad \alpha_{ij}^2=$$

$$\begin{bmatrix} Dep_i & 10 & 9 & 8 & 2 \\ 10 & 0 & 1 & 1 & 1 \\ 9 & 0 & 0 & 0 & 0 \\ 8 & 0 & 1 & 0 & 1 \\ 2 & 0 & 1 & 0 & 0 \end{bmatrix}, \quad \beta_{ij}^2=\begin{bmatrix} Dep_i & 10 & 9 & 8 & 2 \\ 10 & 0 & 1 & 1 & 1 \\ 9 & 0 & 0 & 1 & 1 \\ 8 & 0 & 0 & 0 & 1 \\ 2 & 0 & 0 & 0 & 0 \end{bmatrix}; \quad \alpha_{ij}^3=\begin{bmatrix} Dep_i & 11 & 4 \\ 11 & 0 & 1 \\ 4 & 0 & 0 \end{bmatrix},$$

$$\beta_{ij}^3=\begin{bmatrix} Dep_i & 11 & 4 \\ 11 & 0 & 1 \\ 4 & 0 & 0 \end{bmatrix}, \sum_g^3 x_i^g, \sum_g^3 y_i^g; i \in \boldsymbol{N}\}$$ according to the proposed placement strategy. The coordinate values in the last segment of encoding are determined using the approach described in the next section.

4.3. Two-stage approach

The stage approach concept is based on the attractor–repeller technique developed in the study of Anjos and Vannelli [67] for very-large-scale integration floorplanning, which was used to convert the nonconvex optimization problem into a convex version step by step. In this paper, an optimization-based approach is presented to efficiently find the unique optimal solution for each individual particle by it combining two stages. The idea of the two-stage approach is that, for every individual particle, the search for the relative locations of departments is implemented in the first stage; thereafter, their exact coordinates are determined in the second stage. In particular, decision variables S_k, Y_i^k, α_{ij}^k, and β_{ij}^k mentioned in the mathematical model can be obtained using the placement strategy introduced in Section 4.2. Consequently, the approximate positions of departments can be enforced, and the formulation pertaining to the optimal coordinates can be simplified to a linear programming (LP) model.

The resulting second-stage LP model is given by

objective functions

(1)–(2)

subject to:

(4)–(13), (23)–(26), and (32)–(36).

In general, the placing sequences decoded in the first stage are generated by the operations of MOPSO; thereafter, the above LP model is solved in CPLEX software (except for objective 3 because it is nonconvex). Hence, the optimal objective values of individual particles are acquired and subsequently compared.

4.4. Local and global searches

Note that the original particle swarm optimization is initially used to solve continuous optimization problems, which is supposed to be discretized for the MWFLP. In this section, the velocity and position of particles are redefined; their cognitive activities and social communications are also discretized according to the formulas in Liu et al. [3].

In our work, the process with which particles broadcast their neighborhood space and learn about the Gbest individual is abstracted as the local search of MOPSO; it is employed to search the solution

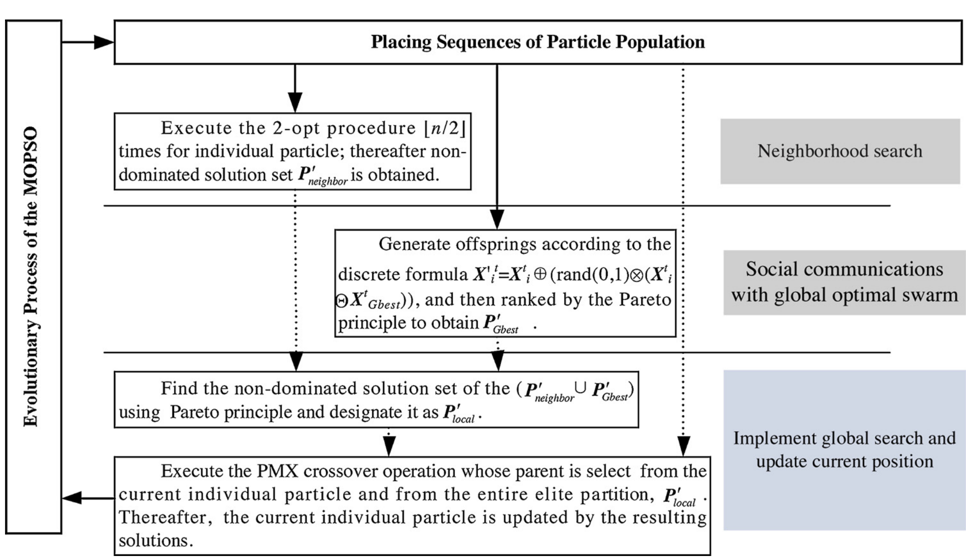


Fig. 4. Local and global search architecture.

space adequately and extensively. Moreover, the global search combines the parents of the current individual particle and elite individuals in the non-dominated solution set obtained from the local research. The architecture of the local and global procedures of MOPSO is illustrated in Fig. 4. The remainder of this section describes the details of the aforementioned procedures.

4.4.1. Neighborhood search

In our MOPSO, suppose that the position of the ith individual particle, $X_i^t = (Dep_{1i}^t, Dep_{2i}^t, ..., Dep_{ni}^t)$ represents a configuration of the MWFLP in the tth interaction. The neighborhood search space around the current individual particle is regarded here as a series of countable solutions with considerably similar sequences. Thus, the 2-opt procedure [6] is suggested to generate some offspring particles from the current individual. A valid 2-opt move consists of swapping the positions of two elements in the current sequence; this propels the search away from the current individual particle; however, it should not be extremely far. Moreover, it can be useful to execute several samplings in the neighborhood search space that strive to traverse all feasible solutions and find one that is preferable.

The specific executive process of the neighborhood search in the MOPSO is as follows.

```
2-opt_ NeighborSearch (Current individual, X_i^t; number of departments n)
SetP_neighbor is employed to record offspring particles;
    SetP'_neighbor represents the non-dominated solutions;
    SetF_neighbor indicates the objective function values of P_neighbor;
    SetF'_neighbor expresses the objective function values of P'_neighbor;
repeat
for i = 1 to i≤⌊n/2⌋
a, b:=random index between 1 and n;
while a=b do
a, b:=random index between 1 and n;
end while
swap positions of elements a and b in X_i^t, and an offspring is generated;
calculate the objective function values of the offspring by applying the placement
    strategy and two-stage approach;
record the novel solution in set P_neighbor, and add the objective function values to set
    F_neighbor;
end for
P_neighbor is filtered to obtain the non-dominated solution setP'_neighbor;
F'_neighbor:=f(P'_neighbor, placement strategy, two-stage approach);
return (P'_neighbor and F'_neighbor)
end
```

4.4.2. Social communications with Gbest

Apart from exploring the neighborhood solution space, it is necessary for all particles to learn from the global optimal swarm to accelerate the convergence process in the local search. Accordingly, the restructuring of the formula that describes the social communications with Gbest is presented in this section; it is expressed as Eq. (37). In this equation, X_{Gbest}^t denotes the best position that global particles in the population have obtained thus far; it is replaced by the elite individual selected at random from the non-dominated solution set. The specific discrete processes of this equation are introduced in the following section.

$$X'^t_i = X_i^t \oplus (\mathrm{rand}(0,1) \otimes (X_i^t \Theta X_{Gbest}^t)) \tag{37}$$

The special definition and discrete arithmetic of the procedure are described in detail as follows.

The velocity, I^i, in the MOPSO is defined as the pairs of distinct departments generated from solutions X_i^t and X_{Gbest}^t according to the subtraction term $(X_i^t \Theta X_{Gbest}^t)$ in Eq. (37). Suppose that $X_{Gbest}^t = (Dep_{1Gbest}^t, Dep_{2Gbest}^t, ..., Dep_{nGbest}^t)$, which is a valid swapping pair, is composed of different elements in the same dimension. If the ath element, Dep_{ai}^t, in sequence X_i^t is distinct from element Dep_{aGbest}^t in sequence X_{Gbest}^t, then the ath swapping pair, $I_a^i(X_i^t, X_{Gbest}^t)$, will be $(Dep_{ai}^t, Dep_{aGbest}^t)$; otherwise, $I_a^i(X_i^t, X_{Gbest}^t) = \varnothing$. After executing the arithmetic subtraction of particle i, the multiplication term (rand(0,1) $\otimes$ I^i) is used to choose the swapping pairs randomly $\lfloor \mathrm{rand}(0,1) \times |I^i| \rfloor$; rand(0,1) is a uniform random number between 0 and 1. Moreover, as for the arithmetic addition ($\oplus$) operation, the novel placing sequences can be acquired in each addition operation on particle i with every swapping pairs selected by arithmetic multiplication. Here, the swapping pair $(Dep_{ai}^t, Dep_{aGbest}^t)$ is considered as an example. The specific multiplication process exchanges the positions of Dep_{ai}^t and Dep_{aGbest}^t departments in the current placing sequence. It is worthwhile to explain that swapping pairs $(Dep_{ai}^t, Dep_{aGbest}^t)$ and $(Dep_{aGbest}^t, Dep_{ai}^t)$ lead to the same new particle; hence, duplicate checking work is essential to avoid replicating calculations after the subtraction operation. Finally, the non-dominated solution set, P'_{Gbest}, is obtained by filtering the new solutions using the Pareto principle.

In general, the current particle, X_i^t, is guided to converge with non-dominated solutions according to Eq. (37); this is used to improve the convergence speed and ameliorate the individual's quality.

4.4.3. Global search and update of current particle

After each particle searches its neighborhood space and learns about the elite individual, preferable solutions that are used to update the current position in the global search probably exist. Accordingly, offspring particles that dominate other particles from solution set $P'_{Gbest} \cup P'_{neighbor}$ are first found. Thereafter, the partial mapped crossover (PMX) [68] operation is applied to combine particle i and each non-dominated individual particle, X_{local}^t, from P'_{local}; the concrete process of the PMX procedure is shown in Fig. 5. A simple case is also illustrated in this figure: the continuous random substring is determined firstly in the sequence, which is called mapping section. Here, the substring composed of elements 3–5 in the two-parent solutions is selected; thereafter, the mapping section is exchanged to generate two new sequences. However, duplicated elements in the new sequences may probably exist; thus, the conflict detection operation has to be executed to guarantee the feasibility of solutions. For example, for the elements outside the two mapping sections, the second reiterated element, "5", of sequence in the third row can be replaced by element "1" according to the mapping list; this is similarly the case with elements "4" and element "6". Consequently, a feasible offspring solution, $X'^t_{i,2}$, is produced. As mentioned above, the profitable mapping sections from elite individuals are available for the PMX to use; it can evolve the current particle to approach the Pareto front.

Note that the similarity between offspring solutions and elite individuals is dependent upon the mapping section length. If the offspring solutions consist of larger fragments of elite individuals, it is probable that better features will be inherited. Thus, an adaptable method that can adjust the length of the mapping section is proposed for the PMX operation. The size of the mapping section is calculated by $\lfloor p_{pmx} \times n \rfloor$, where p_{pmx} indicates the probability obtained by the equation below. In Eq. (38), $p_{pmx}^{\max}$ and $p_{pmx}^{\min}$ express the upper and lower bounds of p_{pmx}, respectively, and P_{Pareto} represents the global non-dominated solution set; Z is the number of objectives considered in Section 3.2. Moreover, F_b^i indicates the bth objective function value of the ith particle in P'_{local}; $F_b^{\max}$ and $F_b^{\max}$ represent the maximum and minimum of the bth objective function value in P_{Pareto}, respectively. The term $1/|P'_{local}| \cdot \sqrt{\sum_{i=1}^{|P'_{local}|} d_i^2}$ in Eq. (38) describes the affinity between P'_{local} and P_{Pareto}; the smaller its value, the closer P'_{local} is to the global non-inferior solution set.

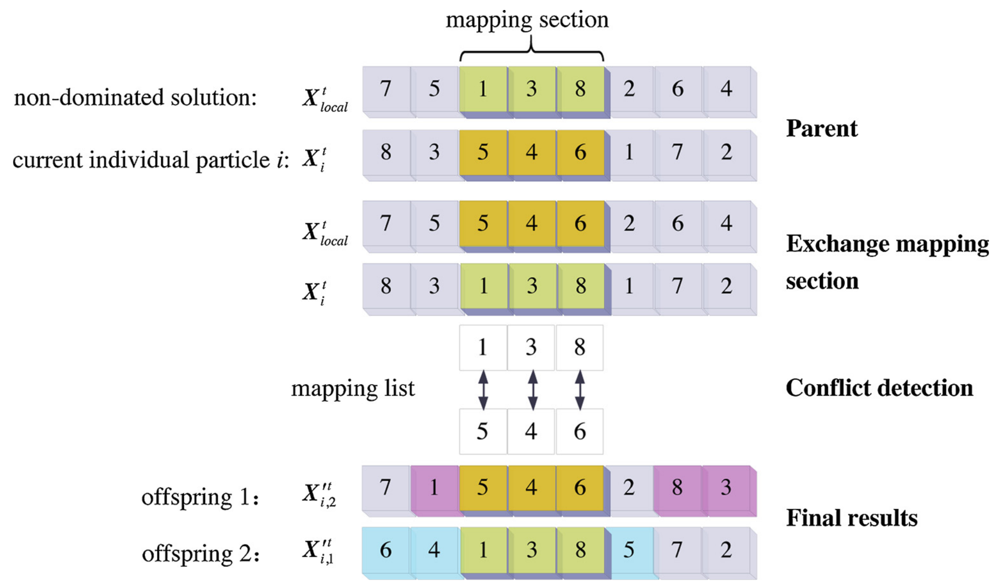


Fig. 5. Partial Mapped Crossover (PMX) operation.

$$p_{pmx} = p_{pmx}^{\min} + (p_{pmx}^{\max} - p_{pmx}^{\min})\cdot\left(1 - \frac{1}{|\boldsymbol{P}'_{local}|}\cdot\sqrt{\sum_{i=1}^{|\boldsymbol{P}'_{local}|} d_i^2}\right)$$

where

$$d_i = \min_{j\in\{1,2,\ldots,|\boldsymbol{P}_{Pareto}|\}}\left\{\sqrt{\sum_{b=1}^{Z}\left(\frac{F_b^i - F_b^j}{F_b^{\max} - F_b^{\min}}\right)^2}\right\} \tag{38}$$

Accordingly, a series of offspring solutions are generated by adopting the PMX operation; some solutions that dominate particle i are saved and used to update themselves.

4.5. Generation of new population and update of external archive

After each particle complete its local and global searches, there exists probably some preferable solutions that dominate itself. Thereafter, these preferable solutions are assembled with the primary particle population to update $\boldsymbol{P}_{Pareto}$ and produce the next generation population. In our MOPSO, *noP* particles are supposed to compose a population. To guarantee the population quality with each iteration, an individual in the Pareto front is first selected at random. If the population is not complete after all the Pareto optimal solutions have been selected, then the mutation procedure on randomly selected optimal particles is exploited to increase population diversity. The aforementioned procedure is applied with the selection of two elements from one solution and then inserting them into any other positions in the remaining sequence.

After each iteration, an external archive is employed to store the global optimal Pareto solutions, i.e., solutions that can achieve advantageous objectives are added to the external archive. Because the size of the external archive is limited, the crowding distance measure [58] is applied to control the number of solutions in the archive to be less than *Na*.

4.6. Description of MOPSO

The entire process of the MOPSO algorithm is described below. The parameter *maxIter* mentioned as below represents the maximum number of algorithm iterations.

Algorithm: multi-objective particle swarm optimization

Input problem information.
Set algorithm parameters.
Randomly generate *noP* feasible particles, and calculate the objective function values of all particles using the placement strategy and two-stage approach. Find the global optimal Pareto solutions, $\boldsymbol{P}_{Pareto}$, and let $t = 1$.
Let $i = 1$.
Execute neighborhood sampling $\lfloor n/2 \rfloor$ times for the ith particle, $\boldsymbol{X}_i^t$, and gain new configurations, $\boldsymbol{P}'_{neighbor}$.
Choose an optimal individual, $\boldsymbol{X}^t_{Gbest}$, from the current$\boldsymbol{P}_{Pareto}$; thereafter compute the velocity $\boldsymbol{I}^i(\boldsymbol{X}_i^t, \boldsymbol{X}^t_{Gbest})$ and novel positions, $\boldsymbol{P}'_{Gbest}$, according to Eq. (37).
Filter $\boldsymbol{P}'_{neighbor} \cup \boldsymbol{P}'_{Gbest}$ based on the Pareto principle to acquire non-dominated individual particles, $\boldsymbol{P}'_{local}$, obtained by the local search; the PMX operation is applied to $\boldsymbol{X}_i^t$ and each sequence in $\boldsymbol{P}'_{local}$.
From the offspring solutions obtained in Step 7, select and record solutions that dominate particle$\boldsymbol{X}_i^t$.
Let $i = i + 1$. If $i \le noP$, then return to Step 5; otherwise, proceed to the next step.
Update the external archive, $\boldsymbol{P}_{Pareto}$, of all particles.
Generate the new population for the next iteration.
Execute $t = t + 1$, and return to Step 4 until $t > maxIter$.
Terminate the MOPSO, and export the final external archive $\boldsymbol{P}_{Pareto}$ and $f(\boldsymbol{P}_{Pareto}$ placement strategy, two-stage approach).

5. Numerical experiments and analysis

To assess the performance of our proposed MOPSO, a series of computational experiments are executed on a computer with the same configuration as that mentioned in Section 3.3. The MOPSO algorithm is coded in MATLAB 2016b, and the linear programs in Section 4.3 are achieved with IBM ILOG CPLEX V12.8 optimization software. In Section 5.1, two different groups of instances are investigated, and comparisons with other algorithms are implemented. For each instance, the MOPSO is run 30 times independently; which is similar with other studies reported in literature. However, detailed sample data of repeated calculations obtained by other existing methods are unable to be found in their papers, which makes it impossible to use the paired-sample t test or other statistical methods. On the other hand, there are some processed statistical results reported in their works such as the best, worst and average objective function values/running time, which were all compared in our work.

Subsequently, several benchmark instances are applied to solve the MWFLP using the MOPSO; the applicability of the proposed method and the Pareto optimal solutions for each instance are demonstrated in Section 5.2.

Table 3
Comparison of results between proposed method and modified PSO for UA–FLP.

Problem instance	Method	F_{MHC} (min)			Running time(s)	
		Best	Worst	Average	Time for the best result	Time for the worst result
IM08	Modified PSO	193.7488	220.3437	208.74165	220.69	205.67
	Proposed MOPSO	**191.5000**	**191.5000**	**191.5000**	**71.62**	**78.22**
IM11	Modified PSO	1286.1069	1371.3264	1335.6385	888.32	919.74
	Proposed MOPSO	**1259.2000**	**1293.3000**	**1274.0600**	**134.29**	**125.91**
MHI20	Modified PSO	1206.6489	1315.2316	1264.2131	2352.12	2250.87
	Proposed MOPSO	**1205.5000**	**1250.5000**	**1234.0000**	**747.11**	**800.95**

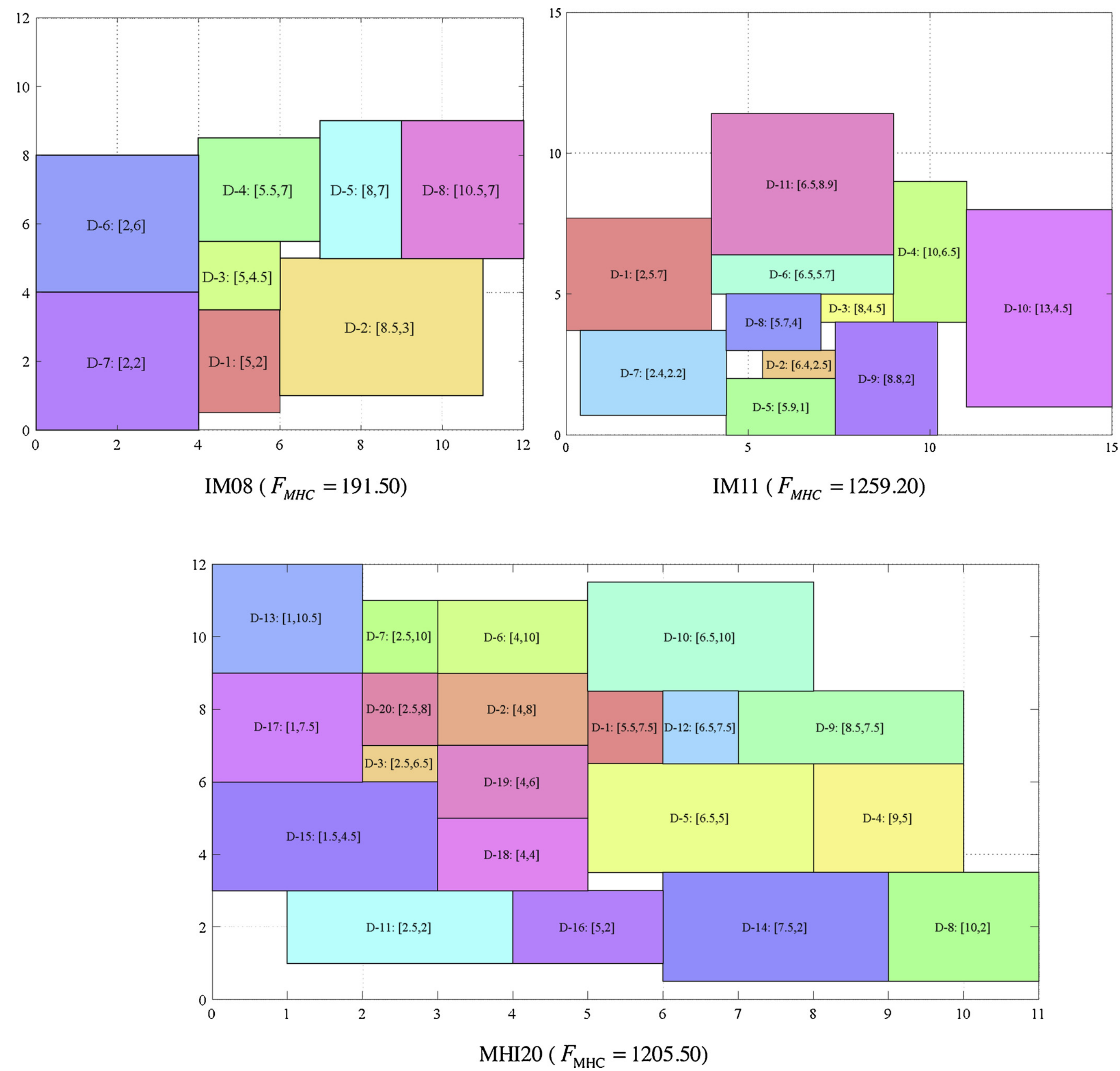


Fig. 6. Best solutions obtained by the proposed method for datasets IM08, IM11, and MHI20.

5.1. Comparison experiments for UA–FLP

In order to evaluate the behavior of our method objectively, the shape-fixed UA–FLP instances are first solved using MOPSO and compared with the results obtained by other algorithms. These UA–FLP problem instances are selected because there is no other problem instance in literature that is available for our MWFLP.

Table 4
Comparison of results between our proposed method and other methods for MHI20 benchmark.

Problem	Evaluation criteria	Our method-MOPSO	Modified PSO [16]	TOPOPT method [69]	FLOAT method [70]	HOT method [71]
MHI20	Best	**1205.5000**	1206.6489	1320.72	1264.94	1225.40
	Average	**1234.0000**	1264.2131	1395.64	1333.81	1287.29
	Dev (%)	—	+2.3899	+11.5818	+7.4831	+4.1397

Table 5
Comparison between previous approaches and our MOPSO for instances n= 15, 20, and 30.

Problem	Objective function value F_{MHC}/Running time (s)					
	MOPSO	HGA [8]	GA [7]	HGA [7]	TS [13]	FA [20]
n = 15	**2546.61/6.11**	6813/81.21	9120/28.2	6941.4/34.1	6615.81/17	5956.1/—
n = 20	**4946.70/16.98**	13 190/167.59	21 885/40.6	14 696/50.7	13 198.40/50	11 505/—
n = 30	**11 538.88**/86.66	35 358/284.01	50 492/**67.7**	32 386/83.9	33 721.20/95.4	29 115/—

Table 6
Best cost solutions for instances n= 15, 20, and 30.

Problem	Optimal layout	Center-coordinates of departments {(location coordinates along X-axis, location coordinates along Y-axis)}
n = 15	{3, 4, 12, 9, 11, 13, 7, 8, 5, 14, 15, 2, 1, 10, 6}	{(17.15, 11.00), (12.85, 9.85), (3.38, 4.23), (6.70, 2.60), (17.50, 6.10), (7.45, 14.10), (6.80, 5.60), (12.85, 6.00), (13.20, 2.00), (3.26, 12.10), (17.70, 2.00), (9.20, 2.00), (9.20, 5.10), (8.20, 9.10), (4.70, 9.10)}
n = 20	{11, 10, 3, 9, 8, 16, 4, 5, , 17, 20, 2, 1, 15, 7, 6, 18, 12, 14, 13}	{(2.15, 12.15), (12.25, 9.60), (12.63, 3.88), (7.50, 6.85), (17.45, 8.35), (16.15, 12.95), (7.60, 9.85), (2.58, 7.35), (17.45, 4.25), (7.86, 2.85), (3.43, 3.35), (8.70, 12.45), (12.90, 12.95), (8.70, 16.55), (5.50, 10.05), (5.50, 6.20), (5.50, 8.05), (5.80, 12.55), (9.40, 6.50), (9.30, 9.30)}
n = 30	{3, 26, 24, 18, 14, 10, 16, 12, 6, 17, 27, 23, 20, 7, 30, 25, 1, 4, 11, 22, 19, 13, 15, 28, 29, 8, 21, 2, 9, 5}	{(3.00, 16.00), (7.85, 19.95), (2.33, 9.33), (13.00, 15.00), (8.65, 23.90), (20.64, 13.00), (9.10, 12.40), (11.93, 19.00), (20.07, 20.50), (7.09, 9.40), (16.00, 16.50), (17.49, 11.10), (19.50, 17.00), (21.39, 8.00), (11.00, 16.00), (15.21, 10.55), (11.00, 12.00), (15.83, 8.00), (9.10, 14.65), (15.13, 12.95), (15.71, 20.65), (6.55, 16.00), (6.70, 12.90), (11.93, 8.40), (11.00, 14.00), (8.65, 6.55), (13.00, 12.00), (22.50, 17.00), (8.98, 17.00), (17.32, 13.60)}

Table 7
Results of n = 40 and n = 50 facility layout problems using the firefly algorithm.

Method	Trial	noP	$maxIter$	n = 40			n = 50		
				F_{MHC}(min)		Running time	F_{MHC}(min)		Running time
				Best	Average	Average	Best	Average	Average
FA[20]	1	100	500	488 611	499 546.3	198.0	2 056 984.0	2 084 621.9	300.0
	2	100	1000	486 505	498 544.2	396.0	2 036 950.0	2 076 700.2	648.0
	3	200	500	485 736	497 569.8	498.0	2 033 821.0	2 074 298.3	738.0
	4	200	1000	485 150	495 940.0	798.0	1 995 866.0	2 046 281.0	1494.0
MOPSO	1	5	30	**376 274**	**383 609.3**	551.2	**1 598 395.5**	**1 661 332.7**	1406.7

5.1.1. Computational results for problem instances IM08, IM11, and MHI20

First, three problem instances with different scales are solved using the MOPSO. These three instances, namely IM08, IM11, and MHI20, are from Asl and Wong [16]. The computational results of the MOPSO and modified PSO in literature are summarized in Table 3. In this table, the parameters for our method are set as follows: noP = 30, $maxIter$ = 100, Na = 15, $p_{pmx}^{\max}$ =0.6, and $p_{pmx}^{\min}$ =0.5. Because only a single objective, F_{MHC}, is considered in Asl and Wong [16], the best results of F_{MHC} obtained by MOPSO will be selected from the external archive and used for comparison, as listed in Table 3; the numbers written in bold denote the best values of these results.

As summarized in Table 3, the overall best objective values obtained by the MOPSO are uniformly better than those yielded by the modified PSO; thus, the same applied to the worst and average objective values. This is particularly the case with instance IM08; the optimal objective of 191.50 is achieved at every run. Furthermore, in terms of running time, the times spent for obtaining the best or worst results using our proposed method are considerably less than those of the modified PSO. This is because, for Modified PSO, the optimization for the exact coordinates of departments is integrated in the intelligent algorithms instead of solving the exact model via optimizers. For verification convenience, the optimal results of each instance are illustrated in Fig. 6.

Additionally, the comparison of MOPSO with other methods is listed in Table 4. It is evident that the MOPSO has improved the best objective function, F_{MHC}, compared with other methods; the improvement over other methods is measured by the relative deviation (Dev) criteria, which are listed in the last row of Table 4. The symbol "+" indicates that the average results obtained by the MOPSO are superior to those of the other four methods. Accordingly, the advantages of the proposed MOPSO for solving the instances above are verified.

5.1.2. Comparison of methods with five instances (from n = 15 to n = 50)

The other five available benchmark instances for the UA–FLP are also employed to check the effectiveness of our proposed MOPSO further. The detailed data of these instances are found in Scholz et al. [13]

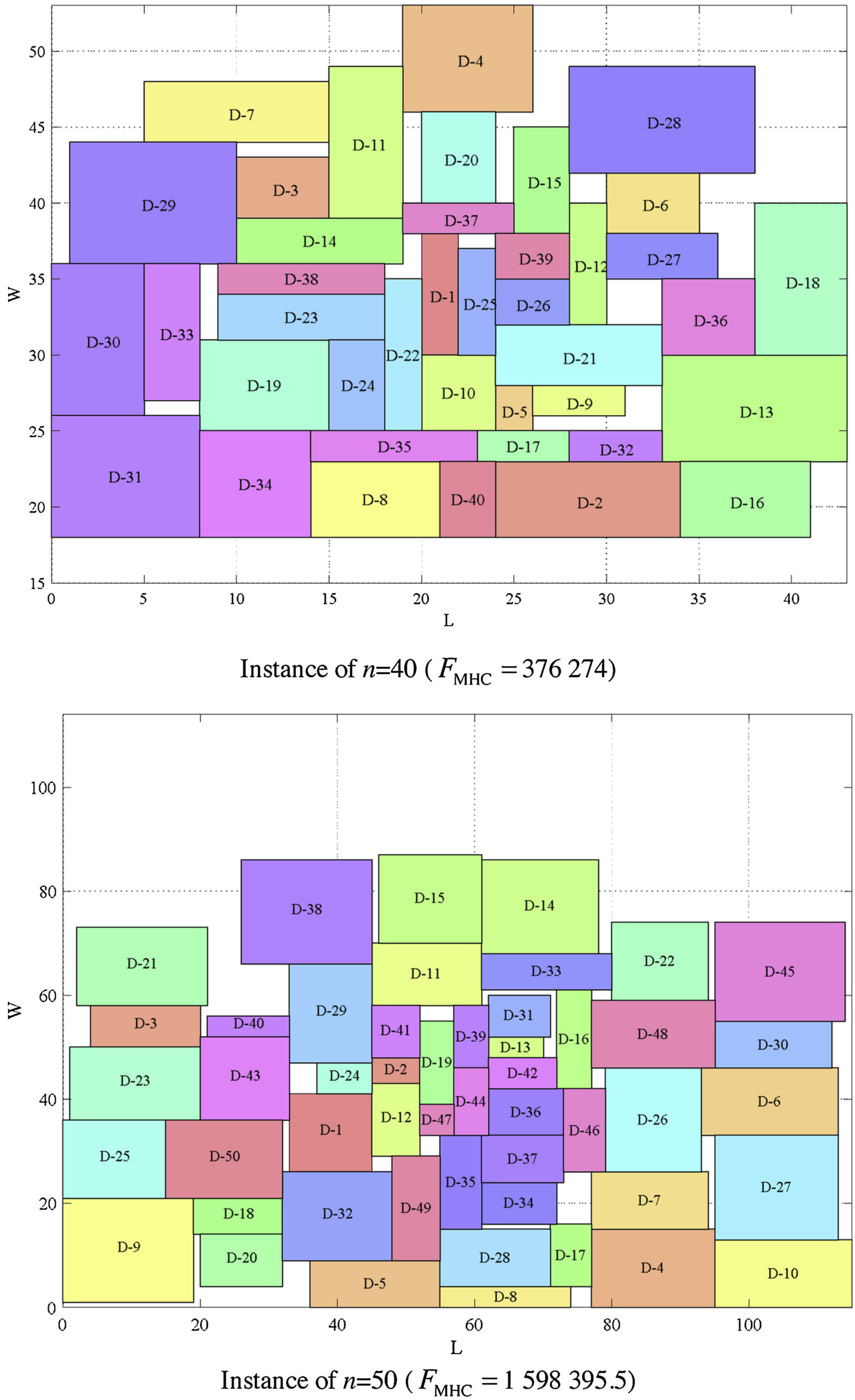


Fig. 7. Best layout obtained by our method for instances n = 40 and 50.

Table 8
Parameter values for MOPSO method.

Instance name	*noP*	*maxIter*	$p_{pmx}^{\max}$	$p_{pmx}^{\min}$	*Na*
Das06	10	10	0.8	0.3	15
SFLP11	30	100	0.8	0.3	15
SFLP20	30	200	0.8	0.3	15

and Ingole and Singh [20]; all instances have been tested for optimizing objective F_{MHC}.

Table 5 summarizes the best cost results obtained by the MOPSO and five other approaches from literature for typical instances $n = 15$, 20, and 30. The parameter combination for our MOPSO is set as $noP = 5$, $maxIter = 10$, $Na = 15$, $p_{pmx}^{\max} = 0.6$, and $p_{pmx}^{\min} = 0.2$. As can be observed in Table 5, the objective function values of MOPSO are significantly better than those of the HGA (hybrid genetic algorithms), GA (genetic algorithms), TS (tabu search), and FA (firefly algorithm); this could be attributed to the different encoding and decoding methods used for the shape-fixed UA–FLP. Given that the departments/facilities are arranged in the considered bay from the top left to the horizontal direction to avoid vertical overlapping among them in any other approaches, the remaining empty spaces among the bays are ignored for placement; this leads to redundant distances among the departments. On the other hand, the running times of the MOPSO (except for the instance $n = 30$) are less than those of the other methods in this study. As for instance $n = 30$, the GA method developed in Lee and Lee [7] only spent 67.7 s to solve the problem; however, the result is considerably inferior compared to that of MOPSO as well as those of the other approaches. The best cost solutions obtained by the MOPSO for these three instances are listed in Table 6.

The two larger extra benchmark instances, $n = 40$ and 50 (provided by Ingole and Singh [20]), are used here to test the performance of the MOPSO. The statistics of the computational results are summarized in Table 7. The table lists the best and average costs found by the MOPSO; the average value of running times and four other parameter combination trials of the FA are also listed. In this table, it is evident that the results obtained by the MOPSO are better than those obtained by the FA; this suggests that the performance of the proposed algorithm is preferable to other related algorithms. It should also be noted that the values of parameters *noP* and *maxIter* decided for the MOPSO are considerably less than those for the FA in spite of the longer running times of the MOPSO. This may be explained by the multiple complex operations embedded in the MOPSO that would consume a considerable amount of time with each iteration for every particle. Finally, the best solutions generated by the MOPSO for instances $n = 40$ and 50 are shown in Fig. 7.

5.2. Solutions to multi-workplace facility layout problem

As mentioned above, the performance of the MOPSO is demonstrated by several different instances that are available in literature. The proposed algorithm is significantly superior to the previous methods even though it is forced to be compared with single objective optimization algorithms. Thus, the MOPSO is implemented to solve the MWFLP here; for a few of the problem instances, it is attempted to achieve some excellent Pareto solutions. As presented in this section, the distances outside the workshops among departments which are assigned to different workshops are presumed to be the same, i.e., $dExt_{ij} = L_F$ when $\sum_{k=1}^{m} q_{ij}^{k}$ is not equal to one; here, a trivial simplification of the multi-workplace facility layout problem is made for the MOPSO.

The three problem instances that are adopted for the MWFLP are Das6, SFLP11, and SFLP20; the detailed data of instances Das6 and SFLP11 are described in Section 3.3, whereas instance SFLP20 with 20 departments is obtained from Mir and Imam [71]. For problem instance SFLP20, the default dimensions of each workshop floor are 10×10, and the unit moving costs inside and outside the workshops are set to 8 and 14, respectively; the additional dataset of frequency of material flows is provided in the appendix.

To achieve balance between solving quality and efficiency, numerous experimental tests are conducted; the preferable algorithm parameter combinations are listed in Table 8. For the instances considered, the initial attempt is to solve their objective, F_{MHC}, via CPLEX optimization software; however, this attempt failed to find optimal results except for instance Das06. Thereafter, the final suboptimal solutions, running time, and MIP gap tolerances are obtained, as listed in columns 2–4 of Table 9. Moreover, the best Pareto optimal solution set obtained by the MOPSO is extracted from the results of ten independent operations; these are also listed in Table 9.

As listed in Table 9, it would require an enormous computational cost to acquire the exact optimal solutions through accurate methods. In the table, instance Das06 is easily solved in 1.78 s, whereas SFLP11 and SFLP20 are barely calculated even after a long time. Thus, as the size of problem instances increases, it also becomes increasingly difficult to achieve effective solutions using accurate methods. For comparison convenience, the obtained cost results and relative tolerances of instances SFLP11 and SFLP20 are listed in the table after a long running time. On the other hand, as can be observed from the computational statistics of the MOPSO, a cluster of better solutions is obtained within reasonable times. As for instance Das06, the two best Pareto solutions could be gained at every turn in 10 repeated runs; the best cost objective value of 42944 obtained by the MOPSO is equal to the exact result but achieved within a shorter time. For instances SFLP11 and SFLP20, 15 and 6 Pareto optimal solutions are achieved, respectively. The best objective function values of F_{MHC} are all better than those

Table 9
Computational results of instances Das 6, SFLP11, and SFLP20 obtained by MOPSO.

Instance name	Solved by CPLEX software			Pareto optimal solution set obtained by MOPSO	
	F_{MHC}	Running time (s)	MIP Gap tolerances (%)	Objective function values: [F_{MHC}, F_{NWS}, F_{UR}]	Running time (s)
Das06	42 944	1.78	0.00	[42 944, 2, 0.372] [71 956, 2, 0.362]	0.62
SFLP11	185 234	854.45	99.98	[171 276, 3, 0.379] [243 064, 4, 0.298] [224 320, 5, 0.270] [222 584, 5, 0.286] [219 430, 3, 0.356] [188 060, 4, 0.357] [227 820, 4, 0.307] [239 978, 4, 0.299] [207 544, 4, 0.319] [201 074, 4, 0.331] [212 246, 3, 0.358] [207 024, 3, 0.371] [194 252, 4, 0.339] [204 942, 4, 0.329] [189 896, 4, 0.348]	306.02
SFLP20	495 140	8661.61	100	[123 052, 2, 0.224] [125 660, 2, 0.215] [129 848, 2, 0.204] [166 316, 2, 0.185] [299 404, 2, 0.184] [302 448, 2, 0.180]	1177.25

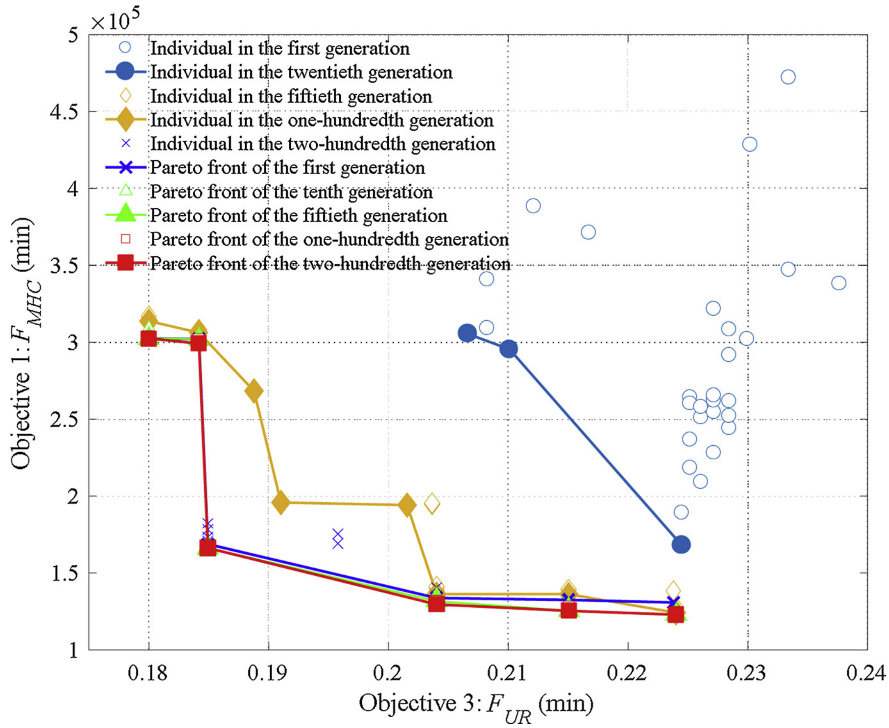


Fig. 8. Population convergence process of MOPSO method for instance SFLP20.

obtained by CPLEX; moreover, the running times of the MOPSO are considerably shorter. In summary, our proposed MOPSO is convincingly a more effective approach to solve a complex problem. Meanwhile, the various solutions with a different emphasis can be employed by organization leaders to aid in decision-making.

In order to demonstrate the performance of our MOPSO more clearly, Fig. 8 illustrates the specifics of the particle convergence process in the MOPSO method for instance SFLP20. In the figure, generations 1, 20, 50, 100, and 200 in the convergence process are selected and described; the scatter points in the figure represent individual particles, and the Pareto front of each generation is indicated by the curves created by Pareto optimal solutions (expressed by solid points). Only two objectives, F_{MHC} and F_{UR}, are shown in Fig. 8 because the objective values of F_{NWS} for most Pareto optimal solutions is 2. As can be observed in the figure, the particle population is concentrated near the contemporary Pareto front except for the initial generation; this illustrates that the MOPSO method has an excellent convergence. Furthermore, there are several Pareto optimal solutions distributed uniformly on the front; this demonstrates the good dispersion capability of the MOPSO method.

6. Conclusions and future research directions

This study presents a multi-workshop facility layout problem, which involves the physical organization of a group of departments inside several workshops. This problem has two aspects: arrangement of departments within the workshops and exact locations of departments without overlapping. In this study, a mixed integer linear programming model is developed for the problem relative to three objectives: the minimization of overall material handling costs including internal and external flow costs, minimization of the number of workshops, and maximization of the utilization ratio of the workshop floor. In order to test our proposed model, several benchmark instances from literature are employed and exact solutions are obtained using CPLEX.

Moreover, a discrete framework for particle swarm optimization is developed to discretize this method and handle the NP-hard problem. In the proposed MOPSO, an individual particle encoding approach and a placement strategy are proposed according to the characteristics of the MWFLP. Thereafter, the local search that is executed by individual particles through several samplings in the neighborhood search space is implemented by a 2-opt operation. As for the social communications among particles, the redefinition of velocity and discrete arithmetic according to the equation are developed to accelerate convergence speed. Furthermore, the two-stage approach for the MOPSO is presented to efficiently find the unique optimal solution for each individual particle. Finally, several available problem instances are applied; the contrastive results suggest the preferable performance of the proposed algorithm.

In future studies, the third objective F_{UR} should be relaxed or reformulated to make it tractable for exact methods. Afterwards, complex obstacle avoidance routes of the AGV/trucks that may influence flow costs will be considered in the manufacturing system. Additionally, more valuable objectives should be considered in line with the actual production situation. On the other hand, more efficient methods have to be developed to enhance the solution quality for the problem.

Acknowledgments

This research was partially supported by the National Natural Science Foundation of China [No.51205328, 51675450]; the Youth Foundation for Humanities, Social Sciences of Ministry of Education of China [No. 18YJC630255]; Sichuan Science and Technology Program [No. 2019YFG0285], and Doctoral Innovation Fund Program of Southwest Jiaotong University [No. G-CX201910].

Appendix A. Frequency of flow matrix for benchmark instances

Instance name	Frequency of material flows
Das04	$f_{ij} = 1, \forall\ i, j \in \{N^* \vert 1 \le i, j \le 4;\ i \ne j\}$
Das06	$f_{ij} = 1, \forall\ i, j \in \{N^* \vert 1 \le i, j \le 6;\ i \ne j\}$
Das08	$f_{ij} = 1, \forall\ i, j \in \{N^* \vert 1 \le i, j \le 8;\ i \ne j\}$
SFLP08	{0 6 8 7 2 1 5 4; 6 0 3 4 2 5 6 8; 8 3 0 3 2 8 5 1; 7 4 3 0 3 8 6 7; 2 2 2 3 0 3 8 4; 1 5 8 8 3 0 1 6; 5 6 5 6 8 1 0 6; 4 8 1 7 4 6 6 0}
Das10	$f_{ij} = 1, \forall\ i, j \in \{N^* \vert 1 \le i, j \le 10;\ i \ne j\}$
SFLP11	{0 10 8 10 5 9 0 7 4 5 1; 10 0 7 7 5 11 5 0 1 10 3; 8 7 0 2 1 8 3 1 7 6 2; 10 7 2 0 6 4 2 4 9 10 7; 5 5 1 6 0 6 2 6 5 7 9; 9 11 8 4 6 0 5 7 1 11 4; 0 5 3 2 2 5 0 4 3 3 9; 7 0 1 4 6 7 4 0 2 7 7; 4 1 7 9 5 1 3 2 0 3 0; 5 10 6 10 7 11 3 7 3 0 7; 1 3 2 7 9 4 9 7 0 7 0}
SFLP20	{0 5 9 11 14 15 16 11 12 15 3 19 3 20 18 9 15 11 15 10; 5 0 13 13 2 8 19 2 20 12 4 2 1 11 2 2 0 10 5 3; 9 13 0 11 11 14 7 2 11 19 1 2 19 14 15 20 1 18 15 16; 11 13 11 0 11 14 13 3 10 12 13 3 6 20 15 7 13 16 19 2; 14 2 11 11 0 9 9 14 16 0 6 3 6 6 11 6 12 15 17 6; 15 8 14 14 9 0 17 10 5 2 11 12 7 8 4 1 11 1 2 5; 16 19 7 13 9 17 0 4 10 17 14 11 9 9 12 6 15 1 7 11; 11 2 2 3 14 10 4 0 18 10 10 1 13 15 6 1 14 2 7 2; 12 20 11 10 16 5 10 18 0 17 11 19 1 16 3 10 16 16 14 8; 15 12 0 12 0 2 17 10 17 0 9 15 17 2 4 15 6 19 12 2; 3 4 1 13 6 11 14 10 11 9 0 15 11 4 18 13 14 14 16 2; 19 2 2 3 3 12 11 1 19 15 15 0 17 7 1 2 11 3 7 16; 3 1 19 6 6 7 9 13 1 17 11 17 0 1 5 2 8 14 4 6; 20 11 14 20 6 8 9 15 16 2 4 7 1 0 1 16 1 2 2 12; 18 2 15 15 11 4 12 6 3 4 18 1 5 1 0 18 16 2 15 19; 9 2 20 7 6 1 6 1 10 15 13 2 2 16 18 0 7 13 4 9; 15 0 1 13 12 11 15 14 16 6 14 11 8 1 16 7 0 7 8 14; 11 10 18 16 15 1 1 2 16 19 14 3 14 2 2 13 7 0 11 15; 15 5 15 19 17 2 7 7 14 12 16 7 4 2 15 4 8 11 0 9; 10 3 16 2 6 5 11 2 8 2 2 16 6 12 19 9 14 15 9 0}

References

[1] Anjos MF, Vieira MVC. Mathematical optimization approaches for facility layout problems: the state-of-the-art and future research directions. Eur J Oper Res 2017;261:1–16. https://doi.org/10.1016/j.ejor.2017.01.049.

[2] Turanoglu B, Akkaya G. A new hybrid heuristic algorithm based on bacterial foraging optimization for the dynamic facility layout problem. Expert Syst Appl 2018;98:93–104. https://doi.org/10.1016/j.eswa.2018.01.011.

[3] Liu J, Zhang H, He K, Jiang S. Multi-objective particle swarm optimization algorithm based on objective space division for the unequal-area facility layout problem. Expert Syst Appl 2018;102:179–92. https://doi.org/10.1016/j.eswa.2018.02.035.

[4] Zhang Z, Mao L, Guan C, Zhu L, Wang Y. An improved scatter search algorithm for the corridor allocation problem considering corridor width. Soft Comput 2019. https://doi.org/10.1007/s00500-019-03925-4.

[5] Ahmadi A, Pishvaee MS, Akbari Jokar MR. A survey on multi-floor facility layout problems. Comput Ind Eng 2017;107:158–70. https://doi.org/10.1016/j.cie.2017.03.015.

[6] Guan C, Zhang Z, Li Y. A flower pollination algorithm for the double-floor corridor allocation problem. Int J Prod Res 2019. https://doi.org/10.1080/00207543.2019.1566673.

[7] Lee YH, Lee MH. A shape-based block layout approach to facility layout problems using hybrid genetic algorithm. Comput Ind Eng 2002;42:237–48. https://doi.org/10.1016/s0360-8352(02)00018-9.

[8] Schnecke V, Vornberger O. Hybrid genetic algorithms for constrained placement problems. Int J Radiat Biol 1997;1:266–77. https://doi.org/10.1109/4235.687887.

[9] Paes FG, Pessoa AA, Vidal T. A hybrid genetic algorithm with decomposition phases for the Unequal Area Facility Layout Problem. Eur J Oper Res 2017;256:742–56. https://doi.org/10.1016/j.ejor.2016.07.022.

[10] Wang W, Koren Y. Scalability planning for reconfigurable manufacturing systems. Int J Ind Manuf Syst Eng 2012;31:83–91. https://doi.org/10.1016/j.jmsy.2011.11.001.

[11] Allahyari MZ, Azab A. Mathematical modeling and multi-start search simulated annealing for unequal-area facility layout problem. Expert Syst Appl 2018;91:46–62. https://doi.org/10.1016/j.eswa.2017.07.049.

[12] Defersha FM, Hodiya A. A mathematical model and a parallel multiple search path simulated annealing for an integrated distributed layout design and machine cell formation. Int J Ind Manuf Syst Eng 2017;43:195–212. https://doi.org/10.1016/j.jmsy.2017.04.001.

[13] Scholz D, Petrick A, Domschke W. STaTS: A Slicing Tree and Tabu Search based heuristic for the unequal area facility layout problem. Eur J Oper Res 2009;197:166–78. https://doi.org/10.1016/j.ejor.2008.06.028.

[14] Komarudin WKY. Applying ant system for solving unequal area facility layout problems. Eur J Oper Res 2010;202:730–46. https://doi.org/10.1016/j.ejor.2009.06.016.

[15] Liu J, Liu J. Applying multi-objective ant colony optimization algorithm for solving the unequal area facility layout problems. Appl Soft Comput 2019;74:167–89. https://doi.org/10.1016/j.asoc.2018.10.012.

[16] Asl AD, Wong KY. Solving unequal-area static and dynamic facility layout problems using modified particle swarm optimization. J Intell Manuf 2017;28:1317–36. https://doi.org/10.1007/s10845-015-1053-5.

[17] Onut S, Tuzkaya UR, Dogac B. A particle swarm optimization algorithm for the multiple-level warehouse layout design problem. Comput Ind Eng 2008;54:783–99. https://doi.org/10.1016/j.cie.2007.10.012.

[18] Ulutas BH, Islier AA. A clonal selection algorithm for dynamic facility layout problems. Int J Ind Manuf Syst Eng 2009;28:123–31. https://doi.org/10.1016/j.jmsy.2010.06.002.

[19] Anjos MF, Fischer A, Hungerlaender P. Improved exact approaches for row layout problems with departments of equal length. Eur J Oper Res 2018;270:514–29. https://doi.org/10.1016/j.ejor.2018.04.008.

[20] Ingole S, Singh D. Unequal-area, fixed-shape facility layout problems using the firefly algorithm. Eng. Optimiz 2017;49:1097–115. https://doi.org/10.1080/0305215x.2016.1235327.

[21] Ou-Yang C, Utanilma A. Hybrid estimation of distribution algorithm for solving single row facility layout problem. Comput Ind Eng 2013;66:95–103. https://doi.org/10.1016/j.cie.2013.05.018.

[22] Wang M-J, Hu MH, Ku M-Y. A solution to the unequal area facilities layout problem by genetic algorithm. Comput Ind 2005;56:207–20. https://doi.org/10.1016/j.compind.2004.06.003.

[23] Kia R, Khaksar-Haghani F, Javadian N, Tavakkoli-Moghaddam R. Solving a multi-floor layout design model of a dynamic cellular manufacturing system by an efficient genetic algorithm. Int J Ind Manuf Syst Eng 2014;33:218–32. https://doi.org/10.1016/j.jmsy.2013.12.005.

[24] Armour GC, Buffa ES. A heuristic algorithm and simulation approach to relative location of facilities. Manage Sci 1963;9:294–309. https://doi.org/10.1287/mnsc.9.2.294.

[25] Che A, Zhang Y, Feng J. Bi-objective optimization for multi-floor facility layout problem with fixed inner configuration and room adjacency constraints. Comput Ind Eng 2017;105:265–76. https://doi.org/10.1016/j.cie.2016.12.018.

[26] Hahn P, Smith JM, Zhu Y-R. The multi-story space assignment problem. Ann Oper Res 2010;179:77–103. https://doi.org/10.1007/s10479-008-0474-3.

[27] Azevedo MM, Crispim JA, de Sousa JP. A dynamic multi-objective approach for the reconfigurable multi-facility layout problem. Int J Ind Manuf Syst Eng 2017;42:140–52. https://doi.org/10.1016/j.jmsy.2016.12.008.

[28] Shiripour S, Mahdavi I, Amiri-Aref M, Mohammadnia-Otaghsara M, Mahdavi-Amiri N. Multi-facility location problems in the presence of a probabilistic line barrier: a mixed integer quadratic programming model. Int J Prod Res 2012;50:3988–4008. https://doi.org/10.1080/00207543.2011.579639.

[29] Kosucuoglu D, Bilge U. Material handling considerations in the FMS loading problem with full routing flexibility. Int J Prod Res 2012;50:6530–52. https://doi.org/10.1080/00207543.2011.653837.

[30] D'Antonio G, Saja A, Ascheri A, Mascolo J, Chiabert P. An integrated mathematical model for the optimization of hybrid product-process layouts. Int J Ind Manuf Syst Eng 2018;46:179–92. https://doi.org/10.1016/j.jmsy.2017.12.003.

[31] Hasan MA, Sarkis J, Shankar R. Agility and production flow layouts: an analytical decision analysis. Comput Ind Eng 2012;62:898–907. https://doi.org/10.1016/j.cie.2011.12.011.

[32] Bozorgi N, Abedzadeh M, Zeinali M. Tabu search heuristic for efficiency of dynamic facility layout problem. Int J Adv Manuf Technol 2015;77:689–703. https://doi.org/10.1007/s00170-014-6460-9.

[33] Aiello G, La Scalia G, Enea M. A multi objective genetic algorithm for the facility layout problem based upon slicing structure encoding. Expert Syst Appl 2012;39:10352–8. https://doi.org/10.1016/j.eswa.2012.01.125.

[34] Tari FG, Neghabi H. A new linear adjacency approach for facility layout problem with unequal area departments. Int J Ind Manuf Syst Eng 2015;37:93–103. https://doi.org/10.1016/j.jmsy.2015.09.003.

[35] Emami S, Nookabadi AS. Managing a new multi-objective model for the dynamic facility layout problem. Int J Adv Manuf Technol 2013;68:2215–28. https://doi.org/10.1007/s00170-013-4820-5.

[36] Hathhorn J, Sisikoglu E, Sir MY. A multi-objective mixed-integer programming model for a multi-floor facility layout. Int J Prod Res 2013;51:4223–39. https://doi.org/10.1080/00207543.2012.753486.

[37] Lakehal S, Aitzai A. Branch and bound for facility layout problem using minimum weighted clique problem in complete k-partite graph. 2017 International Conference on Robotics and Artificial Intelligence, ICRAI 2017 2017:20–5. https://doi.org/10.1145/3175603.3175610. Association for Computing Machinery. Shanghai, China; 2017.

[38] Dunker T, Radons G, Westkamper E. Combining evolutionary computation and

dynamic programming for solving a dynamic facility layout problem. Eur J Oper Res 2005;165:55–69. https://doi.org/10.1016/j.ejor.2003.01.002.
[39] Udomsakdigool A, Bangsaranthip S. Combining ant colony optimization and dynamic programming for solving a dynamic facility layout problem. World Acad Sci Eng Technol 2010;40:528–32. https://doi.org/10.5281/zenodo.1333288.
[40] Wang S, Zuo X, Liu X, Zhao X, Li J. Solving dynamic double row layout problem via combining simulated annealing and mathematical programming. Appl Soft Comput 2015;37:303–10. https://doi.org/10.1016/j.asoc.2015.08.023.
[41] Ahmadi A, Akbari Jokar MR. An efficient multiple-stage mathematical programming method for advanced single and multi-floor facility layout problems. Appl Math Model 2016;40:5605–20. https://doi.org/10.1016/j.apm.2016.01.014.
[42] Bernardi S, Anjos MF. A two-stage mathematical-programming method for the multi-floor facility layout problem. J Oper Res Soc 2013;64:352–64. https://doi.org/10.1057/jors.2012.49.
[43] El Kady A, Sami SA, Eldeib AM. A two stage heuristics for improvement of existing multi floor healthcare facility layout. 9th International Conference on Bioinformatics and Biomedical Technology, ICBBT 2017, May 14, 2017 - May 16 2017:97–101. https://doi.org/10.1145/3093293.3093308.
[44] Zhang Y, Gong D-W, Ding Z. A bare-bones multi-objective particle swarm optimization algorithm for environmental/economic dispatch. Inf Sci (Ny) 2012;192:213–27. https://doi.org/10.1016/j.ins.2011.06.004.
[45] Zhang Y, D-w G, Ding Z. Handling multi-objective optimization problems with a multi-swarm cooperative particle swarm optimizer. Expert Syst Appl 2011;38:13933–41. https://doi.org/10.1016/j.eswa.2011.04.200.
[46] Mousavi SM, Bahreininejad A, Musa SN, Yusof F. A modified particle swarm optimization for solving the integrated location and inventory control problems in a two-echelon supply chain network. J Intell Manuf 2017;28:191–206. https://doi.org/10.1007/s10845-014-0970-z.
[47] Zhang Y, D-w G, Cheng J. Multi-objective particle swarm optimization approach for cost-based feature selection in classification. IEEE ACM T COMPUT BI 2017;14:64–75. https://doi.org/10.1109/tcbb.2015.2476796.
[48] Zhang Y, D-w G, Zhang W. Feature selection of unreliable data using an improved multi-objective PSO algorithm. Neurocomputing 2016;171:1281–90. https://doi.org/10.1016/j.neucom.2015.07.057.
[49] Zhang Y, D-w G, Zhang J. Robot path planning in uncertain environment using multi-objective particle swarm optimization. Neurocomputing 2013;103:172–85. https://doi.org/10.1016/j.neucom.2012.09.019.
[50] Kaucic M. Equity portfolio management with cardinality constraints and risk parity control using multi-objective particle swarm optimization. Comput Oper Res 2019;109:300–16. https://doi.org/10.1016/j.cor.2019.05.014.
[51] Romero C. Extended lexicographic goal programming: a unifying approach. Omega-Int J Manage Sci 2001;29:63–71. https://doi.org/10.1016/s0305-0483(00)00026-8.
[52] Baril C, Yacout S, Clement B. Design for Six Sigma through collaborative multi-objective optimization. Comput Ind Eng 2011;60:43–55. https://doi.org/10.1016/j.cie.2010.09.015.
[53] Cortés BM, García JCE, Hernández FR. Multi-objective flow-shop scheduling with parallel machines. Int J Prod Res 2012;50:2796–808. https://doi.org/10.1080/00207543.2011.593006.
[54] Samanta S, Philip D, Chakraborty S. Bi-objective dependent location quadratic assignment problem: formulation and solution using a modified artificial bee colony algorithm. Comput Ind Eng 2018;121:8–26. https://doi.org/10.1016/j.cie.2018.05.018.
[55] K-s Z, Z-h H, W-j L, Song W. Bilevel Adaptive Weighted Sum Method for Multidisciplinary Multi-Objective Optimization. Aiaa J 2008;46:2611–22. https://doi.org/10.2514/1.36853.
[56] Zahlan J, Asfour S. A multi-objective approach for determining optimal air compressor location in a manufacturing facility. Int J Ind Manuf Syst Eng 2015;35:176–90. https://doi.org/10.1016/j.jmsy.2015.01.003.
[57] Zhang Z, Wang K, Zhu L, Wang Y. A Pareto improved artificial fish swarm algorithm for solving a multi-objective fuzzy disassembly line balancing problem. Expert Syst Appl 2017;86:165–76. https://doi.org/10.1016/j.eswa.2017.05.053.
[58] Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multiobjective genetic algorithm: NSGA-II. Int J Radiat Biol 2002;6:182–97. https://doi.org/10.1109/4235.996017.
[59] Zitzler E, Thiele L. Multiobjective evolutionary algorithms: A comparative case study and the Strength Pareto approach. Int J Radiat Biol 1999;3:257–71. https://doi.org/10.1109/4235.797969.
[60] Coello CAC, Pulido GT, Lechuga MS. Handling multiple objectives with particle swarm optimization. Int J Radiat Biol 2004;8:256–79. https://doi.org/10.1109/tevc.2004.826067.
[61] Zhang Y, Gong DW, Sun JY, Qu BY. A decomposition-based archiving approach for multi-objective evolutionary optimization. Inf Sci (Ny) 2018;430:397–413. https://doi.org/10.1016/j.ins.2017.11.052.
[62] Menchaca-Mendez A, Coello Coello CA. Ieee. GDE-MOEA: a new MOEA based on the generational distance indicator and E-dominance. Ed. 2015.
[63] Das SK. A facility layout method for flexible manufacturing systems. Int J Prod Res 1993;31:279–97. https://doi.org/10.1080/00207549308956725.
[64] Zhang Y, Lu C, Zhang H, Fang Z. Workshop layout optimization based OB differential cellular multi-objective genetic algorithm. Comput Integr Manuf Syst 2013;19:727–34. https://doi.org/10.13196/j.cims.2013.04.57.zhangy.019.
[65] Lee KY, Roh MI, Jeong HS. An improved genetic algorithm for multi-floor facility layout problems having inner structure walls and passages. Comput Oper Res 2005;32:879–99. https://doi.org/10.1016/j.cor.2003.09.004.
[66] Ulutas BH, Kulturel-Konak S. An artificial immune system based algorithm to solve unequal area facility layout problem. Expert Syst Appl 2012;39:5384–95. https://doi.org/10.1016/j.eswa.2011.11.046.
[67] Anjos MF, Vannelli A. An Attractor-Repeller approach to floorplanning. Math Methods Oper Res (Heidelb) 2002;56:3–27. https://doi.org/10.1007/s001860200197.
[68] Ting C-K, Su C-H, Lee C-N. Multi-parent extension of partially mapped crossover for combinatorial optimization problems. Expert Syst Appl 2010;37:1879–86. https://doi.org/10.1016/j.eswa.2009.07.082.
[69] Imam MH, Mir M. Nonlinear programming approach to automated topology optimization. Comput Aided Des 1989;21:107–15. https://doi.org/10.1016/0010-4485(89)90146-2.
[70] Imam MH, Mir M. Automated layout of facilities of unequal areas. Comput Ind Eng 1993;24:355–66. https://doi.org/10.1016/0360-8352(93)90032-s.
[71] Mir M, Imam MH. A hybrid optimization approach for layout design of unequal-area facilities. Comput Ind Eng 2001;39:49–63. https://doi.org/10.1016/s0360-8352(00)00065-6.