A Heuristic Algorithm for Multi-Objective Unequal-Area Facility Layout Problems (Postprint)

Authors: Liu Jingfa, Liu Jun

Date: 2018-06-19T00:00:00+00:00

Abstract

The multi-objective unequal-area facility layout problem (UA-FLP) involves the placement of unequal-area facilities within a workshop to optimize multiple objectives while satisfying certain constraints. A multi-objective optimization model for production workshops is established with the objectives of minimizing material handling costs and maximizing non-logistics relationship strength, and a heuristic algorithm is proposed to solve it. The algorithm employs a heuristic layout update strategy for configuration updates, and addresses interference constraints between facilities by combining a local search mechanism based on the adaptive step size gradient method with a heuristic facility deformation strategy. To obtain the Pareto optimal solution set for the problem, a Pareto-optimization-based local search and a global optimization method based on niche technology are proposed. The algorithm' s performance is tested through two typical examples, and experimental results demonstrate that the proposed heuristic algorithm is an effective method for solving the multi-objective UA-FLP.

Full Text

Preamble

Heuristic Algorithm for Unequal Area Facility Layout Problem with Multiple Objectives

Liu Jingfaa,b, Liu Juna,b

aJiangsu Engineering Center of Network Monitoring, bSchool of Computer & Software, Nanjing University of Information Science & Technology, Nanjing 210044, China

Abstract: The multi-objective unequal area facility layout problem (UA-FLP) involves placing departments with different areas within a facility to optimize multiple objectives while satisfying certain constraints. This paper establishes a multi-objective optimization model for production workshops based on minimizing material handling costs and maximizing non-logistics relationship strength, and proposes a heuristic algorithm to solve it. The algorithm employs a heuristic layout updating strategy to update configurations, and combines a local search mechanism based on an adaptive step size gradient method with a heuristic facility deformation strategy to handle interference constraints between facilities. To obtain the Pareto optimal solution set for the problem, we propose a local search method based on Pareto optimization and a global optimization method based on niche technology. The algorithm' s performance is tested through two typical examples, and experimental results demonstrate that the proposed heuristic algorithm is an effective method for solving multi-objective UA-FLP.

Keywords: facility layout problem; heuristic algorithm; multi-objective optimization; Pareto optimization; niche technology

0 Introduction

The facility layout problem (FLP) involves placing several facilities to be arranged within a given workshop to optimize one or more objective functions while satisfying certain constraints. A reasonable facility layout can help enterprises improve production efficiency, reduce production costs, and shorten product cycles. Therefore, the workshop facility layout problem has significant research importance in manufacturing.

In recent years, scholars have conducted extensive research on facility layout problems, but most literature discusses equal-area facility layout problems [1], which do not reflect actual workshop layout conditions in real production applications. For the unequal area facility layout problem (UA-FLP), some scholars have also proposed solution algorithms. Komarudin et al. [2] proposed an ant colony system algorithm based on slicing tree representation, combined with several local search methods to improve the algorithm' s search performance. Gonçalves et al. [3] established a linear programming model and proposed a biased random-key genetic algorithm. Liu et al. [4] proposed a sequence-pair representation-based model and solved it by combining a genetic algorithm with mixed-integer programming methods. Vitayasak et al. [5] proposed a backtracking search algorithm based on genetic algorithms for solving UA-FLP, where the backtracking search algorithm is a population-based iterative evolutionary algorithm. Paes et al. [6] proposed a genetic algorithm combined with decomposition strategies, which was tested on classic benchmark instances and obtained high-quality layout solutions. Asl et al. [7] proposed an improved particle swarm optimization algorithm that applied two local search methods and a facility exchange method to improve layout solution quality and prevent local optima. Qiu et al. [8] proposed an improved genetic algorithm and optimized the layout design scheme for a screw manufacturing workshop, with experimental results proving the effectiveness of the proposed algorithm. Liao et al. [9] proposed a genetic algorithm using sequential encoding to optimize workshop layout design to reduce logistics handling costs.

Although the above methods have achieved good progress in solving UA-FLP, the optimization objectives have mainly focused on single-objective optimization centered on material handling costs, neglecting the achievement of other objectives such as space utilization and non-logistics relationship strength. For multi-objective UA-FLP, the linear weighted sum method is generally used to transform multiple objectives into a single objective for solution. However, in the linear weighting method, the weight coefficients are difficult to set reasonably and have certain limitations [10]. Therefore, Pareto optimization methods have become a new approach for multi-objective solution. Xu et al. [11] proposed an adaptive genetic algorithm for UA-FLP, which overcame the shortcomings of premature convergence and falling into local optima through nonlinear processing of its crossover and mutation operators. Ripon et al. [12] proposed an algorithm based on evolutionary methods with adaptive variable neighborhood search to minimize material handling costs and maximize non-logistics relationship strength, with experimental results showing good performance. Kumar et al. [13] proposed a method based on integer linear programming and variable neighborhood for solving multi-objective UA-FLP. Recently, Hunagund et al. [14] proposed a simulated annealing algorithm based on a flexible bay structure for dynamic UA-FLP. Liu et al. [15] proposed a multi-objective particle swarm optimization algorithm combined with objective space division methods to optimize material handling costs, total adjacency values, and workshop utilization rates.

This paper studies multi-objective UA-FLP and proposes a heuristic algorithm (HA) by applying local search based on Pareto optimization and global optimization search based on niche technology, combined with multiple heuristic strategies. Through calculations on two representative examples from the literature, experimental results demonstrate that the proposed heuristic algorithm is an effective algorithm for solving multi-objective UA-FLP.

1.1 Problem Description

Assume all facilities and the workshop have rectangular shapes with variable length and width, but fixed area. The sum of facility areas equals or is less than the workshop area. The multi-objective UA-FLP can be described as: given the length and width of a rectangular workshop, the area of all facilities to be arranged, the material flow between facilities, the unit material handling cost, and the non-logistics relationship strength between facilities, the goal is to provide a layout scheme that minimizes material handling costs and maximizes non-logistics relationship strength while satisfying the following constraints: (a) all facilities must be placed within the workshop, facilities cannot overlap, and there must be no overlap between facilities and workshop boundaries; (b) all facilities must satisfy minimum side length or maximum aspect ratio constraints.

As shown in [Figure 1: see original paper], assume the origin of the Cartesian coordinate system is located at the lower left corner of rectangular workshop F, where L and W are the length and width of rectangular workshop F, and li and wi represent the length and width of facility Di. Facilities can be placed in either horizontal or vertical orientation. If the long side of a facility is parallel to the x-axis, it is horizontally placed; if the long side is parallel to the y-axis, it is vertically placed. A layout can be represented as X = (x1, y1, l1, w1, x2, y2, l2, w2, ···, xN, yN, lN, wN), where (xi, yi) represents the center coordinates of facility Di, and N is the number of facilities in the workshop.

1.2 Mathematical Model

a) Objective Functions

Objective function (1) represents the minimization of material handling cost (MH Cost), where Cij is the unit material handling cost per unit distance between facilities Di and Dj; fij is the material handling frequency between facilities Di and Dj; and dij is the material handling distance between facilities Di and Dj, i.e., the Euclidean distance between facility center points. Objective function (2) represents the maximization of non-logistics relationship strength (CR Score), where rij is the non-logistics closeness value between facilities (as shown in). If facilities share a common boundary, the closeness value is calculated based on the closeness rating between facilities; otherwise, the closeness value is 0. A larger non-logistics relationship value between facilities indicates a closer non-logistics relationship.

b) Constraints

Where Ii and Ij represent the interior spaces of facilities Di and Dj, and and s represent the maximum aspect ratio value and minimum side length value of facility Di, respectively. Constraint (3) indicates that facilities Di and Dj cannot overlap; constraints (4)-(7) indicate that facilities must be within the workshop interior and cannot overlap with workshop boundaries; constraint (8) indicates that all facilities must satisfy maximum aspect ratio constraints; and constraint (9) indicates that all facilities must satisfy minimum side length constraints.

shows the classification of facility relationship closeness.

2 Algorithm Design

For multi-objective UA-FLP, to obtain a set of Pareto optimal solutions, this paper proposes a heuristic algorithm (HA). First, a set of n initial layouts (configurations) is randomly generated. For each layout, a heuristic layout updating strategy is proposed to update the current layout. Then, a layout legalization mechanism combining an adaptive step size gradient method and heuristic facility deformation strategy is applied to make each layout satisfy constraints. Next, two search methods—local search based on Pareto optimization and global search based on niche technology—are proposed to find optimal layouts. Finally, the specific steps of the proposed heuristic algorithm are presented.

2.1 Heuristic Layout Updating Strategy

In the proposed HA, layouts need to be updated in each iteration. Based on the characteristics of workshop layouts, this paper proposes a heuristic layout updating strategy for facility selection and placement.

If the current layout is illegal, the facility with the maximum relative compression elastic potential energy CPi is selected for re-placement, where CPi is calculated using formula (11), and Ai is the area of facility Di.

If the current layout is legal, the facility with the maximum relative material handling cost MCi is selected for re-placement.

To increase layout diversity, the facility placement method is as follows: the selected facility' s center is randomly placed in the blank area of the workshop 2N times (if there is no blank area in the workshop, the selected facility' s center is randomly placed in the workshop 2N times), and each placement includes both horizontal and vertical orientations, resulting in a total of 4N new configurations.

2.2 Layout Legalization

After generating 4N new configurations through the heuristic layout updating strategy, these configurations may not satisfy the constraints, and there may be overlaps between facilities or between facilities and workshop boundaries. This paper performs layout legalization operations on illegal configurations by alternately executing the local search mechanism and heuristic facility deformation strategy to make all configurations satisfy constraints.

Drawing on the quasi-physical concept [16], all facility interiors and the exterior of the workshop are imagined as smooth elastic entities. According to the principles of elasticity mechanics, when interference exists between two different facilities or between a facility and workshop boundaries, the compression elastic potential energy between them is proportional to the square of the embedding depth [17]. Therefore, the total elastic potential energy of the entire system is:

Where: is the elastic coefficient, set to = 1 in this paper; E(X) represents the total elastic potential energy of the layout; lij represents the embedding depth between facilities Di and Dj; and liF represents the embedding depth between facility Di and workshop boundaries. If facilities Di and Dj overlap, the embedding depth is defined as the shortest distance required to move facilities Di and Dj to a non-overlapping state in either the x-axis or y-axis direction [17]. If facility Di intersects with workshop boundaries, there are three cases: (a) facility Di intersects with workshop boundaries in the x-axis direction (as shown in Figure 2: see original paper); (b) facility Di intersects with workshop boundaries in the y-axis direction (as shown in Figure 2: see original paper); (c) facility Di intersects with workshop boundaries in both x-axis and y-axis directions simultaneously (as shown in Figure 2: see original paper). The embedding depths are liF = |bipF|, |cipF|, and |dipF|, respectively, where |AB| represents the Euclidean distance between vertices A and B.

2.2.1 Local Search Mechanism After obtaining a set of new layouts through the heuristic layout updating strategy, to find layouts with lower elastic potential energy near the current layout, this paper introduces a local search mechanism based on the adaptive step size gradient method (GM) [18], which uses elastic forces to push apart overlapping parts between facilities or between facilities and workshop boundaries to obtain legal layouts.

GM, also known as the steepest descent method, uses the negative gradient direction as the search direction where the function value decreases fastest. In the GM iteration process, for a configuration X1, the gradient in the x-axis and y-axis directions is first calculated to obtain a new configuration X2 = X1 - h × (E(X1)), where h is the iteration step size. If the energy E(X2) of the newly generated configuration X2 is higher than the energy E(X1) of X1, it indicates that the step size h is too large, and h is reduced by multiplying it by a coefficient of 0.8; otherwise, the step size remains unchanged. To shorten the running time of GM, an "early escape" strategy is applied to GM: if the energy of the newly generated configuration X2 is very close to that of configuration X1 (|E(X2) -E(X1)| < 10-4) and the energy E(X2) of the new configuration X2 is greater than 1, then configuration X2 is considered not to be a "promising" configuration, and GM terminates early to avoid excessive search time caused by too small step sizes. If the energy E(X2) of the newly generated configuration X2 is less than 1, GM continues to execute until the configuration' s energy satisfies E(X2) < , or the iteration step size h is smaller than the minimum iteration step size hmin, where is the minimum elastic potential energy value that a legal configuration must achieve. In fact, h, hmin, and are all empirical values obtained through repeated experiments. When h [0.1, 10], hmin [10-5, 10-3], and [10-25, 10-15], the calculation results of the heuristic algorithm in this paper show no significant difference, and all can obtain a legal configuration. Therefore, this paper sets h = 1, hmin = 10-4, and = 10-20.

The detailed steps of the adaptive step size gradient method GM are as follows:

a) Set parameters h = 1, hmin = 10-4, = 10-20;

b) Calculate the gradient of the energy function E(X1) of the current configuration X1 in the x-axis and y-axis directions to obtain the new configuration X2 = X1 - h × (E(X1));

c) If E(X2) 1, then:
   (a) If E(X2) > E(X1), then set h = h × 0.8;
   (b) X1 = X2, X2 = X1 - h × (E(X1));
   (c) If |E(X2) - E(X1)| > 10-4, go to step (a);

d) If E(X2) < 1, then:
   (a) If E(X2) > E(X1), then set h = h × 0.8;
   (b) X1 = X2, X2 = X1 - h × (E(X1));
   (c) If E(X2) and h hmin, go to step (a);

e) Output X2.

2.2.2 Heuristic Facility Deformation Strategy During the execution of GM, a "stuck" phenomenon may occur, where although facility overlaps exist in the configuration, the forces on the overlapping facilities are balanced in two different directions, making it impossible to separate the overlapping facilities by executing GM to legalize the layout. Since the area of facilities is fixed, appropriately changing the length and width of rectangular facilities can yield a legal layout while satisfying constraints (8) and (9). Therefore, after executing GM, a heuristic facility deformation strategy is proposed to legalize the layout.

The heuristic facility deformation strategy proceeds as follows: First, all facilities that overlap with workshop boundaries are selected and processed in ascending order of embedding depth between the facility and workshop boundaries. The overlapping parts are moved to blank areas around the facility while ensuring that the deformed facility remains rectangular and satisfies constraints (8) and (9). Second, for cases where facilities overlap with each other, facilities are similarly selected in ascending order of embedding depth and processed through analogous deformation operations, transferring overlapping parts to blank areas around the facilities. The criterion for determining whether a facility is suitable for deformation operation is whether the total elastic potential energy of the layout system decreases after executing the facility deformation. If the elastic potential energy decreases, the facility deformation is performed; otherwise, it is not executed. If there is insufficient blank area around a facility to accommodate the overlapping parts, part of the overlap is first moved to a blank area, and the remaining overlapping parts along with other overlapping parts are moved to other blank areas around the facility. If overlaps between facilities or between facilities and workshop boundaries still exist after executing the heuristic facility deformation strategy, or if there are gaps around facilities, GM is executed again to make facilities more compact and to obtain a legal layout satisfying constraints.

A typical example is provided to illustrate the facility deformation process (as shown in [Figure 3: see original paper]). For improved readability, blank areas in the workshop are indicated by shading. Assume Figure 3: see original paper shows the layout state after executing GM, where D5 overlaps with D6 and the boundary of workshop F, and D2 overlaps with D1 and D3. According to the principles of the heuristic facility deformation strategy, facilities overlapping with workshop boundaries are selected first for deformation operations. The overlapping part of D5 with the workshop is transferred to the rectangular area above D5 that contains blank space (Figure 3: see original paper). Then, facilities that overlap with each other are selected and processed in ascending order of embedding depth. First, the overlapping parts of D2 and D3 are transferred to the rectangular area to the left of D2 that contains blank space (Figure 3: see original paper). Next, the overlapping parts of D5 and D6 are transferred to the rectangular area below D5 that contains blank space (Figure 3: see original paper). Finally, the overlapping parts of D2 and D1 are moved to the rectangular area to the right of D2 that contains blank space (Figure 3: see original paper). During this process, the overall elastic potential energy of the layout continuously decreases. However, after executing all sequential deformation operations, D5 and D7 still overlap. GM is executed again, ultimately yielding a legal layout as shown in Figure 3: see original paper.

2.3 Pareto-Based Local Search

In the HA algorithm for multi-objective UA-FLP, n initial configurations are first randomly generated. Then, for each configuration Xi (i {1, 2, ···, n}), a trial configuration library consisting of 4N layouts is obtained by executing the heuristic layout updating strategy and layout legalization operations, denoted as TC. In UA-FLP, a solution corresponds to a layout or configuration. The quality function value of a solution and the search direction of the current solution are determined by comparing the Pareto dominance relationship [10] between the current solution Xi and solutions Xj (j = 1, 2, ···, 4N) in its corresponding trial configuration library TC. If a solution Xj in the trial configuration library is infeasible, it indicates that Xj does not help the search direction of the current solution Xi. If Xj is feasible and dominates Xi, it indicates that Xj is beneficial for the current solution Xi to iterate toward the Pareto front, and thus the quality function value of the solution is also larger. At iteration step t, the quality function j(t) of the current solution Xi relative to Xj in TC is defined as follows:

where: j {1, 2, ···, 4N}. Parameters 1, 2, 3, 4 are four levels of the quality function with 1 < 2 < 3 < 4. Considering that the optimization direction is related not only to solution quality but also to the distance between solutions, the larger the quality function value of a solution in the experimental configuration library and the closer it is to the current solution, the greater the probability of being selected as the optimization direction for the current solution. Therefore, for each experimental configuration Xj in TC, the selection probability is defined as:

where: j(t) is the quality function value of Xj at algorithm iteration step t, j(t) = 1/fdij, where fdij is the distance between the current solution Xi and solution Xj in the trial configuration library, calculated as follows:

where: j {1, 2, ···, 4N}, and m is the number of objective functions.

2.4 Global Search Based on Niche Technology

Relying solely on Pareto-based local search methods may cause the algorithm to fall into local minima and make it difficult to maintain solution diversity. To improve the algorithm' s global search capability, this paper proposes a global search method based on niche technology. An external archive is established to store all possible non-dominated solutions found by the current algorithm, denoted as BP. When the algorithm uses the global search based on niche technology for optimization, the most sparsely distributed non-dominated solution Xl in set BP is selected as the search direction for the current solution Xi.

Assume there are p non-dominated solutions X1, X2, ···, Xp in the current BP. For the current solution Xi, the non-dominated solution Xl with the smallest niche count in set BP is selected as the optimization direction for Xi. The niche count niche(l) is calculated as follows:

where: S(fdlj) is the value of the sharing function, calculated as follows:

where: share is the niche radius, and fdlj is the distance between two solutions Xl and Xj in BP, with the calculation formula referred to in equation (15).

2.5 Description of Heuristic Algorithm Steps

The specific iterative steps of HA are as follows:

a) Randomly generate n initial layouts. Set P0 = 0.6, t = 1, and the maximum number of iterations T.

b) For each layout, execute GM and the heuristic facility deformation strategy for legalization operations to obtain a set of layouts that are as legal as possible, called the configuration library, denoted as C.

c) Identify all non-dominated solutions in C, denoted as BP.

d) Set i = 1.

e) Randomly generate a random number P in [0, 1]. When P P0, apply global search based on niche technology to the current configuration Xi in C, selecting the sparsest non-dominated solution from BP, denoted as Xl; otherwise, go to step f).

f) For the current configuration Xi in C, apply the heuristic layout updating strategy to generate 4N new configurations. Execute GM and the heuristic facility deformation strategy on each new configuration to obtain 4N configurations that are as legal as possible, called the trial configuration library, denoted as TC. Apply the Pareto-based local search strategy to select an optimal configuration from TC, denoted as Xl.

g) Update the configuration library C by setting Xi = Xl.

h) Update BP. If the selected feasible solution Xl is non-dominated with respect to BP, add configuration Xi to BP and delete solutions in BP that are dominated by Xl; otherwise, BP remains unchanged.

i) Set i = i + 1. If i n, go to step e); otherwise, go to step j).

j) Set t = t + 1. If t < T, go to step c); otherwise, output BP and terminate the algorithm.

3 Simulation Experiments and Results Analysis

Two commonly used examples of different scales from the literature are employed to test the proposed HA algorithm. Example O8, first proposed by Meller et al. [19], involves placing 8 facilities in a workshop of size 11.31 × 13. In this paper, the number of initial configurations n is set to 200, the number of iterations T is set to 100, and the four levels of the quality function are set to 1 = 0.01, 2 = 0.1, 3 = 2, 4 = 5. Example SC30, proposed by Liu and Meller [4], involves placing 30 facilities in a workshop of size 50 × 50. In this paper, the number of initial configurations n is set to 1000, the number of iterations T is set to 900, and the four levels of the quality function are set to 1 = 0.01, 2 = 0.2, 3 = 8, 4 = 32. For each example, the unit material handling cost per unit distance is set to 1. Detailed parameter settings regarding material flow, facility areas, closeness ratings, maximum aspect ratio constraints, and minimum side length constraints can be found in the literature [2].

3.2 Experimental Results and Analysis

The HA algorithm was implemented in Java and run on a PC with an Intel Core 2 Duo 2.94 GHz CPU and 2.0 GB RAM. For each example, the HA algorithm was independently run 30 times. The algorithm' s performance was evaluated based on the best values (Best) and average values (Avg) of the objectives, and compared with the adaptive variable neighborhood search algorithm (VNS) proposed by Ripon et al. [12], the evolutionary algorithm with local search (EA with LS) and the evolutionary algorithm without local search (EA without LS) proposed by Ripon et al. [20]. The experimental results are listed in .

According to the results in , HA is superior to or equal to the other three algorithms in terms of both best and average values for material handling cost (MH cost) and non-logistics relationship strength (CR Score). For example O8, compared with the best-performing VNS method among the three comparison algorithms, HA improves the result for objective MH Cost by (202.72 - 202.29)/202.71 = 0.21%, while the CR Score value is the same. For example SC30, the HA algorithm improves the result for objective MH cost by (3707 - 3553.53)/3707 = 4.14% compared with the VNS algorithm, and improves the CR score by (392 - 380)/392 = 3.06%. [Figure 4: see original paper] and [Figure

5: see original paper] use triangular points to represent the Pareto optimal solution sets for examples O8 and SC30 obtained by the HA algorithm, respectively. The figures show that the Pareto solutions obtained by the HA algorithm are uniformly distributed with good diversity and form a clear Pareto front.

To further analyze the algorithm's performance, this paper presents the variation of the best values (best) and average values (avg) of the objectives for the two examples as the number of iterations increases. Figure 6: see original paper and Figure 6: see original paper show the iteration graphs for the two objectives MH Cost and CR Score of example O8, respectively. Figure 7: see original paper and Figure 7: see original paper show the iteration graphs for the two objectives MH cost and CR Score of example SC30, respectively. The objective values in the figures are taken from 30 runs of the HA algorithm. The figures show that the objectives of both examples fluctuate significantly in the early stages of algorithm iteration, but as the number of iterations increases, both Best and Avg tend to stabilize, demonstrating that the algorithm has good convergence speed and stability.

4 Conclusion

This paper studies multi-objective UA-FLP and proposes an efficient constraint handling mechanism for solving non-overlapping constraints between facilities by combining GM and the heuristic facility deformation strategy. This constraint handling method is an effective approach for solving non-overlapping problems in (two-dimensional) workshop facility layout problems and can also be extended to three-dimensional rectangular layout problems. However, from the facility deformation process, it is evident that this method is mainly applicable to rectangular facility layouts. Direct and simple extension to circular, triangular, and irregularly shaped facilities is difficult, which will be the direction of future research. Additionally, by combining Pareto-based local search and niche-based global search, the limitations of using a single search method are avoided. This not only prevents the algorithm from falling into local optima and enhances its global search capability but also improves the diversity of Pareto optimal solutions obtained by the algorithm. The performance of the algorithm is tested and analyzed through two typical examples and compared with excellent algorithms in the current literature. Experimental results show that the proposed HA algorithm is an effective method for solving multi-objective UA-FLP, with obvious advantages in solution convergence and diversity, and can provide effective layout solutions for enterprises. At the same time, the proposed HA algorithm can also be extended to solve other multi-objective optimization problems.

References

[1] Matai R, Singh S P, Mittal M L. Modified simulated annealing based approach for multi-objective facility layout problem [J]. International Journal of Production Research, 2013, 51 (14): 4273-4288.

[2] Komarudin, Wong K Y. Applying ant system for solving unequal area facility layout problems with Borda-Kendall-based method [J]. International Journal of Business and Systems Research, 2017, 11 (1-2): 62-81.

[3] Gonçalves J F, Resende M G C. A biased random-key genetic algorithm for the unequal area facility layout problem [J]. European Journal of Operational Research, 2015, 246 (1): 86-107.

[4] Liu Q, Meller R D. A sequence-pair representation and MIP-model-based heuristic for the facility layout problem with rectangular departments [J]. IIE Transactions, 2007, 39 (4): 377-394.

[5] Vitayasak S, Pongcharoen P, Hicks C. A tool for solving stochastic dynamic facility layout problems with stochastic demand using either a genetic algorithm or modified backtracking search algorithm [J]. International Journal of Production Economics, 2017, 190 (8): 146-157.

[6] Paes F G, Pessoa A A, Vidal T. A hybrid genetic algorithm with decomposition phases for the unequal area facility layout problem [J]. European Journal of Operational Research, 2017, 256 (3): 742-756.

[7] Asl A D, Wong K Y. Solving unequal-area static and dynamic facility layout problems using modified particle swarm optimization [J]. Journal of Intelligent Manufacturing, 2017, 28 (6): 1317-1336.

[8] Qiu Shenghai, Chen Shuding, Wang Yunxia, et al. Application of genetic algorithm in optimization of facility layout workshop [J]. Machine Design and Manufacturing Engineering, 2017, 46 (2): 80-83.

[9] Liao Yuanquan, Liu Qiong, Chen Junming. Optimization and simulation research of workshop facility layout based on genetic algorithm [J]. Mechanical Research and Application, 2017, 30 (4): 119-122.

[10] Wang Yong, Cai Zixing, Zhou Yuren, et al. Constrained optimization evolutionary algorithms [J]. Journal of Software, 2009, 20 (1): 11-29.

[11] Xu Liyun, Yang Shouyin, Li Aiping, et al. Multi-objective optimization for facility layout and its simulation analysis [J]. Machine Design and Research, 2011, 27 (6): 55-59.

[12] Ripon K S N, Glette K, Khan K N, et al. Adaptive variable neighborhood search for solving multi-objective facility layout problems with unequal area facilities [J]. Swarm and Evolutionary Computation, 2013, 8 (1): 1-12.

[13] Kumar R, Singh S P. Optimal selection of multi-criteria unequal area facility layout problem: integer linear program [J]. European Journal of Operational Research, 2010, 202 (3): 730-746.

[14] Hunagund I B, Pillai V M, Kempaiah U N. A simulated annealing algorithm for unequal area dynamic facility layout problems with flexible bay structure [J]. International Journal of Industrial Engineering Computations, 2018, 9 (3): 307-330.

[15] Liu Jingfa, Zhang Huiyun, He Kun, et al. Multi-objective particle swarm optimization algorithm based on objective space division for unequal-area facility layout problem [J]. Expert Systems with Applications, 2018, 102 (5): 179-192.

[16] Liu Jingfa, Xue Shengjun, Liu Zhaoxia, et al. An improved energy landscape paving algorithm for the problem of packing circles into a larger containing circle [J]. Computers and Industrial Engineering, 2009, 57 (3): 797-806.

[17] Liu Jingfa, Li Jian, Lyu Zhipeng, et al. A quasi-human strategy-based improved basin filling algorithm for the orthogonal rectangular packing problem with mass balance constraint [J]. Computers and Industrial Engineering, 2017, 107 (5): 196-210.

[18] Liu Jingfa, Li Gang. Basin filling algorithm for the circular packing problem with equilibrium behavioral constraints [J]. Science China: Information Sciences, 2010, 40 (3): 423-432.

[19] Meller R D, Narayanan V, Vance P H. Optimal facility layout design [J]. Operations Research Letters, 1998, 23 (3-5): 117-127.

[20] Ripon K S N, Khan K N, Glette K, et al. Using pareto-optimality for solving multi-objective unequal area facility layout problem [C]// Proc of the 13th Annual Conference on Genetic and Evolutionary Computation. New York: ACM Press, 2011: 681-688.

Note: Figure translations are in progress. See original paper for figures.