Hindawi
Discrete Dynamics in Nature and Society
Volume 2018, Article ID 1529058, 8 pages
https://doi.org/10.1155/2018/1529058

Research Article

An Improved Genetic Algorithm Based Robust Approach for Stochastic Dynamic Facility Layout Problem

Yunfang Peng, Tian Zeng, Lingzhi Fan, Yajuan Han, and Beixin Xia

School of Management, Shanghai University, Shanghai 200444, China

Correspondence should be addressed to Beixin Xia; bxxia@shu.edu.cn

Received 31 August 2018; Accepted 25 November 2018; Published 5 December 2018

Guest Editor: Xinchang Wang

This paper deals with stochastic dynamic facility layout problem under demand uncertainty in terms of material flow between facilities. A robust approach suggests a robust layout in each period as the most frequent one falling within a prespecified percentage of the optimal solution for multiple scenarios. Mont Carlo simulation method is used to randomly generate different scenarios. A mathematical model is established to describe the dynamic facility layout problem with the consideration of transport device assignment. As a solution procedure for the proposed model, an improved adaptive genetic algorithm with population initialization strategy is developed to reduce the search space and improve the solving efficiency. Different sized instances are compared with Particle Swarm Optimization (PSO) algorithm to verify the effectiveness of the proposed genetic algorithm. The experiments calculating the cost deviation ratio under different fluctuation level show the good performance of the robust layout compared to the expected layout.

1. Introduction

Nowadays, the increasingly fierce market competition and the growing variable demands of customers have gradually made the production mode shift from high volume and repetitiveness to high mix and low volume. How to meet the variant production requirements is thought to be an important objective for new intelligent manufacturing system, which is an advanced manufacturing mode of next generation. Designing agile facilities makes much sense to satisfy these requirements [1]. Dynamic changes, such as fluctuations in product quantity, varieties in product mix, introduction of new products, and discontinuation of existing products, are frequently taking place in high-mix and low-volume production environment. As a result of these changes, the previous layout becomes less efficient, which makes material handling cost increased. The uncertainty of demand has brought great challenges to design a suitable facility layout. Therefore, it is of great theoretical and practical significance to study the facility layout problem under dynamic environment [2].

The research on facility layout problem (FLP) started in the 1950s. Koopmans and Beckmann [3] initially defined the facility layout problem as the assignment of facilities to discrete locations with the objective of minimizing the material handling cost. Static facility layout is obtained according to the deterministic material flow when the demand is constant. But when the demand changes frequently with time, static layout becomes no more suitable for various periods in the planning horizon. Dynamic facility layout problem (DFLP) divides the planning horizon into several discrete time periods, with different product demand. Designing flexible layout and designing robust layout are two approaches to deal with DFLP [4]. Flexible layout optimization is to design an optimal layout for each period in the multiperiod planning horizon so that the total material handling and rearrangement cost is minimized. The facility layout is changed according to the production demand in different periods. On the other hand, the robust approach is to design a fixed robust layout to minimize the total material handling cost over the entire time planning horizon [5]. Although the robust layout is not an optimal layout for a particular time period, its performance is good in each period. In a traditional DFLP, the demand in each period is determined by demand forecasting. But, in reality, the product demand in one period is difficult to forecast accurately. Therefore, it would be more valuable to

study DFLP, assuming the product demand is stochastic in each period. Considering uncertainty of the product demand in each period leads to stochastic dynamic facility layout problem (SDFLP) [6].

2. Literature Review

The approaches for the DFLP can be classified into four categories: exact methods, heuristics, metaheuristics, and hybrid approaches [1, 4]. Rosenblatt [7] who is the pioneer in DFLP presented a dynamic programming (DP) formulation. Based on this formulation, both optimal and heuristic procedures are developed. Lacksonen and Enscore [8] extended five algorithms to solve the DFLP, which is modeled as a modified Quadratic Assignment Problem (QAP) formulation. And the cutting plane algorithm was illustrated to be the best for all the test problems. To avoid the computational complexity of the DP and QAP formulations, Urban [9] proposed a new heuristic algorithm based on the steepest-decent pairwise-interchange procedure, which performs well in most of situations. Recently, more specifically metaheuristic and hybrid approaches, such as genetic algorithm (GA), tabu search (TS), and simulated annealing (SA), have been widely applied for DFLP. Kaku and Mazzola [10] defined a TS heuristic utilizing a dynamic tabu list for dynamic plant layout problem. Madhusudanan et al. [5] applied a robust layout to minimize the total material handling cost over all periods. They proposed a mathematical model for robust approach and designed a SA algorithm to solve the model. Tayal and Singh [11] presented mathematical formulation for multiobjective stochastic dynamic facility layout problem and solved it by SA and chaotic simulated annealing (CSA) metaheuristics. The experiment results observed that CSA performs better than SA. GA has been proven to be effective to generate suboptimal solutions for large-scale dynamic facility layout problems. Fazlelahi et al. [12] devised a customized permutation-based robust genetic algorithm in dynamic manufacturing environments, which is expected to be generating a unique robust layout for all the manufacturing periods.

Kulturel-Konak surveyed recent developments in designing robust and flexible facilities layout under uncertainty [1]. Webster and Tyberghein [13] measured the flexibility of a layout as the ability to react to disturbances caused by future change. They analyzed the annual material handling costs to measure the flexibility. Gupta [14] solved the FLP by Monte Carlo simulation to randomly generate the flow between all pairs of departments. Chan and Malmborg [15] also used Monte Carlo simulation to empirically search for robust solutions for dynamic line layout problem. Rosenblatt and Lee [16] solved the single period plant layout problem under stochastic demand by a robustness approach. They defined the robustness of a layout as the frequency it falls within a prespecified percentage of the optimal solution for various sets of scenarios. The robustness approach searches for a reliable layout for all scenarios but not the optimal layout for any given scenario. Besides the scenario-based robust optimization to deal with uncertainty, some other methods such as stochastic programming or fuzzy programming are widely used. Stochastic programming employs probabilistic models and describes the uncertainty by probability distributions. Moslemipour and Lee [6] considered the randomly changing product demands as independent normally distributed random variables with known probability density function. SA metaheuristic algorithm was utilized to solve the mathematical model. Fuzzy programming models uncertain parameters with fuzzy numbers and establishes constraints using fuzzy sets and membership functions. Considering the uncertainty of material flows, Cheng et al. [17] introduced fuzzy numbers to represent the material flows between department pairs. Then, GA was applied to solve this hard fuzzy combinatorial problem. Kaveh et al. [18] modeled the DFLP as fuzzy programming and solved the models by a hybrid intelligent algorithm including GA, simulated annealing, and fuzzy simulation.

Although previous studies have significantly improved FLP with uncertainty, most of articles assumed that the demand is in exact probability distribution or is defined by fuzzy numbers. In fact, the information about the uncertainty is sometimes lacking and its behavior is difficult to predict. Therefore, a scenario-based method is applied in this paper to describe the demand uncertainty. Designing flexible layout and designing robust layout are two approaches to cope with dynamic layout problem. In this article, these two approaches are combined. In each period, a robust layout inspired from Rosenblatt and Lee [16] for DFLP considering the assignment of transport devices under uncertain demands is present. The robust layout is the most frequent layout falling within a prespecified percentage of the optimal solution for different sets of scenarios generated by Mont Carlo simulation. To improve the search speed of finding the robust layout, an improved adaptive genetic algorithm with population initialization strategy is proposed to reduce the search space and improve the efficiency of solving the model.

The rest of this paper is organized as follows. In the next section, the dynamic facility layout problem that considers the assignment of transport devices is modelled. Then, a robust approach based on Monte Carlo simulation is proposed to deal with material flow uncertainty in Section 3. After that, an improved genetic algorithm is developed to solve the mathematical model. Some numerical results are compared and the advantages of the robust layout are illustrated in Section 5. Finally, the conclusions are given.

3. Dynamic Facility Layout Problem

DFLP considers material flow over multiple time periods. The material flow between facilities changes over time. But traditional dynamic layout optimization is studied under the condition that the demand in each period is constant; it cannot effectively solve the problem with demand fluctuation. Therefore, on the basis of proposing an improved adaptive genetic algorithm, Monte Carlo simulation method is used to describe the effect of demand fluctuation on the material flow. In our research, the assignment of transport devices (such as conveyor, AGV, and tow train) is an important decision because of different unit material handling cost for each transport device. Although we consider the fluctuation of material flow in each period, the problem becomes a

Table 1: Notations.

Indices:	
i,j	Indices for facilities where i, j=1,...,I, I= number of facilities in the layout
t	Index for periods where t=1,...,T, T=number of periods in the planning horizon
k,h	Index for locations of facility where k, h=1,...,L, L=number of locations in the layout
z	Indices for transport device types where z=1...,G, G=number of transport device types
Parameters:	
D_{kh}	The distance between location k and location h
A_{kh}	The cost of rearrangement from location k to location h
P_{khz}	The cost of transporting unit material from position k to position h using device z
Q_{tij}	The random variable of the total material flow from facility i to facility j in period t
Variables:	
x_{tik}	Binary variable:1, if facility i is placed at location k in period t; 0 otherwise
y_{tzij}	Binary variable:1, if using device z to transport material between facility i and facility j during period t; 0 otherwise

determined dynamic facility problem when one scenario is generated.

The mathematical model for DFLP is discussed as follows and Table 1 gives the notations used.

The objective (3) is to minimize the total material handling cost (see (1)) and the rearrangement cost (see (2)). Constraints (4) ensure that each facility should be placed in exactly one location in each period. Constraints (5) indicate that a location can place at most one facility in each period. Constraints (6) ensure that only one transport device can be used between two facilities in each period.

$$MHC = \sum_{t=1}^{T}\sum_{z=1}^{Z}\sum_{i=1}^{I}\sum_{j=1}^{I}\sum_{k=1}^{L}\sum_{h=1}^{L} x_{tik} \bullet x_{tjh} \bullet y_{tzij} \bullet Q_{tij} \bullet P_{khz} \bullet D_{kh} \tag{1}$$

$$AC = \sum_{t=1}^{T-1}\sum_{i=1}^{I}\sum_{k=1}^{L}\sum_{h=1}^{L} x_{tik} \bullet x_{(t+1)ih} \bullet A_{kh} \tag{2}$$

$$\min TC = MHC + AC \tag{3}$$

$$\sum_{k=1}^{L} x_{tik} = 1, \quad \forall t, i \tag{4}$$

$$\sum_{i=1}^{I} x_{tik} \le 1, \quad \forall t, k \tag{5}$$

$$\sum_{z=1}^{G} y_{tzij} = 1, \quad \forall t, i, j \tag{6}$$

4. Robust Approach Based on Mont Carlo Simulation

Considering demand uncertainty in terms of Q_{tij}, Monte Carlo simulation method is used to generate N different scenarios. In each period, we find a robust layout, which is the most frequent layout falling within a prespecified percentage of the optimal solution for different sets of production scenarios. Robust layout constraints (7) and (8) are defined to ensure that the gap of cost between robust layout and optimal layout in each scenario falls within a prespecified percentage.

$$RCC_n = \frac{TC_n - TC_n^{opt}}{TC_n^{opt}} \tag{7}$$

$$RCC_n \le M \tag{8}$$

In these constraints, n is the index for scenarios where n=1,..,N. RCC_n is the robust control coefficient of the nth scenario. TC_n^{opt} is optimal cost of the nth scenario, while TC_n denotes the total cost of the robust layout of the nth scenario. M is the prespecified percentage that is suggested to be less than 15% [19]. The detailed process to find the robust layout is as follows (Figure 1).

Step 1. Initially set n=1 and candidate solution set S_n=Φ,.

Step 2. Generate the expected scenario with the material flow Q_{tij} setting to the expected value.

Step 3. Obtain the optimal layout $Layout(n)$ and minimum cost TC_n^{opt} by genetic algorithm.

Step 4. Generate the initial population and improve the solution through evolution process until the solution satisfies the robust layout constraints. Add current layout into the candidate solution set S_n.

Step 5. If the number of elements in S_n is less than N_{max}, go back to the fourth step; otherwise, output set S_n and go to next step.

Step 6. If the number of simulations is less than N, n=n+1. Generate a new scenario corresponding to the random numbers Q_{tij} in a certain range and go back to Step 2. Else, go to the next step.

Step 7. Output the most frequent layout (robust layout) contained in S_n where n=1..N. If more than two layouts are satisfied, search for the robust layout $Layout'$ with minimal cost deviation from the optimal cost.

5. An Improved Adaptive Genetic Algorithm Design

Genetic algorithm (GA) is an excellent heuristic algorithm, which is a random optimization algorithm for the simulation of biological evolution in nature. Because of its strong practicability and robustness, it is widely used in the field of facility layout optimization, scheduling, and transportation. But genetic algorithm has the shortcomings of easy falling into local optimal and premature convergence. Therefore, in view of the above shortcomings, combined with the characteristics of the facility layout problems, an improved genetic algorithm is developed.

5.1. Population Initialization. Because the decision variables of the model are all 0-1 variables, binary coding is adopted. Each cell contains T chromosomes. The first $I \times L$ position in a chromosome defines the layout of the facility, and the latter $G \times I \times I$ position denotes the assignment of transport devices. In order to improve the efficiency of the algorithm and combine the constraint conditions of the model, a new population initialization strategy is proposed on the premise of ensuring diversity, which is as follows.

A sample of two facilities located in three alternative locations with three transport devices in two periods is shown in Figure 2. In the initialization of chromosome, the front-end part, due to $I<L$, generates a matrix A with $L\times L$ dimensions, which has only one element with value 1 in each row and each column, such as A=[0,1,0; 1,0,0; 0,01]. After that, we randomly select I rows to compose matrix B; B=[0,1,0; 0,0,1], thus generating the layout gene position *Chrom1x*=[0,1,0,1,0,0]. The back-end part needs to be initialized according to the transport devices in each period. If Q_{tij} is 0, then directly generate $1\times G$ dimension zero vector. If Q_{tij} is not 0, generate the $1\times G$ dimension row vector with only one element of 1 and the rest is 0. Then it is converted to $1\times(G\times I\times I)$ gene position *Chrom1y*=[0,0,0,1,0,0,0,1,0,0,0,0]. Finally, it is combined with *Chrom1x*, so an initialization chromosome *Chrom1*=[0,1,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0,0] is formed. According to the total number of time periods T, a cell is composed of T chromosomes.

5.2. Selection Operation. Selection operations are the process of eliminating individuals with low adaptation from an old population and selecting excellent individuals with high fitness. Compared to the common genetic algorithm, the selection process is divided into two steps, namely, cell selection and chromosome selection, due to the particularity of this problem. We use traditional roulette wheel selection method to select the cell and chromosome.

5.3. Mutation and Crossover Operations. Crossover is a random selection of individuals' genes and changes them to produce evolutionary operations for new individuals. Mutation operation is a crucial step in genetic algorithm, which ensures the diversity of population and avoids premature convergence. However, for different optimization problems, it is necessary perform tests repeatedly to determine cross probability P_c and mutation probability P_m, which is a very tedious task, and it is difficult to find the best value for each problem. So the idea of adaptive genetic algorithm is applied to determine the probability of cross and mutation.

$$P = \frac{\max(popfitness) - \min(popfitness)}{\max(popfitness)} \tag{9}$$

Because binary coding is adopted in this problem, the mutation and cross operations are combined together. During cross and mutation operations, it is necessary to fragment the chromosomes so as to find the corresponding facilities, locations, and types of transport devices. For the front-end part *Chrom1x*=[0,1,0; 0,0,1], as shown in Figure 3, assuming the mutation position is 2, the corresponding value 1 indicates that facility 1 is placed on location 2. Then value changes from 1 to 0, and the remaining two locations are randomly selected to place facility 1. If it is placed on location 1, there is no need for adjustment. If it is placed on position 3, we found that location 3 has already been placed by facility 2. Then, facility 2 is adjusted to the original variant location (location 2). If the mutation position is 3 corresponding to value 0, which means facility 1 is not placed on location 3, then change the value from 0 to 1, and meanwhile change the value 1 on position 2 to 0. After that change, we find that facilities 1 and 2 are both placed on location 3; then we randomly place facility 2 on the remaining locations. As a result, in condition 1, the values in positions 2 and 3 are crossed, and the values in positions 5 and 6 are exchanged. Meanwhile, in condition 2, the values in positions 1 and 2 are exchanged.

For the variation of the back-end part, as shown in Figure 4, if the mutation position is 2 corresponding to value 0, because this position represents no transport devices between facility 1 and facility 1, there is no need to do any operation. If the mutation position is 4 corresponding to value 0, we change value 0 to 1 and change the remaining position to 0. If the variation position is 5 corresponding to value 1, we change value 1 to 0. The residual position is randomly selected as 1. As a result, in conditions 2 and 3, the values on positions 4 and 5 are exchanged.

5.4. Termination Criteria. The algorithm is terminated when the generation reached the maximal number M, or the best value did not improve for $0.05\times M$ generations.

6. Computational Analysis

Because the transport device is not mentioned in most of articles, there are no standard benchmark instances. 20 determined instances with different size as article [20] are randomly generated. The size of instance is denoted by $T\_I\_L\_G$, where T is the number of periods, I is the number of facilities, L is the number of locations, and G is the number of transport device types. We compare the improved

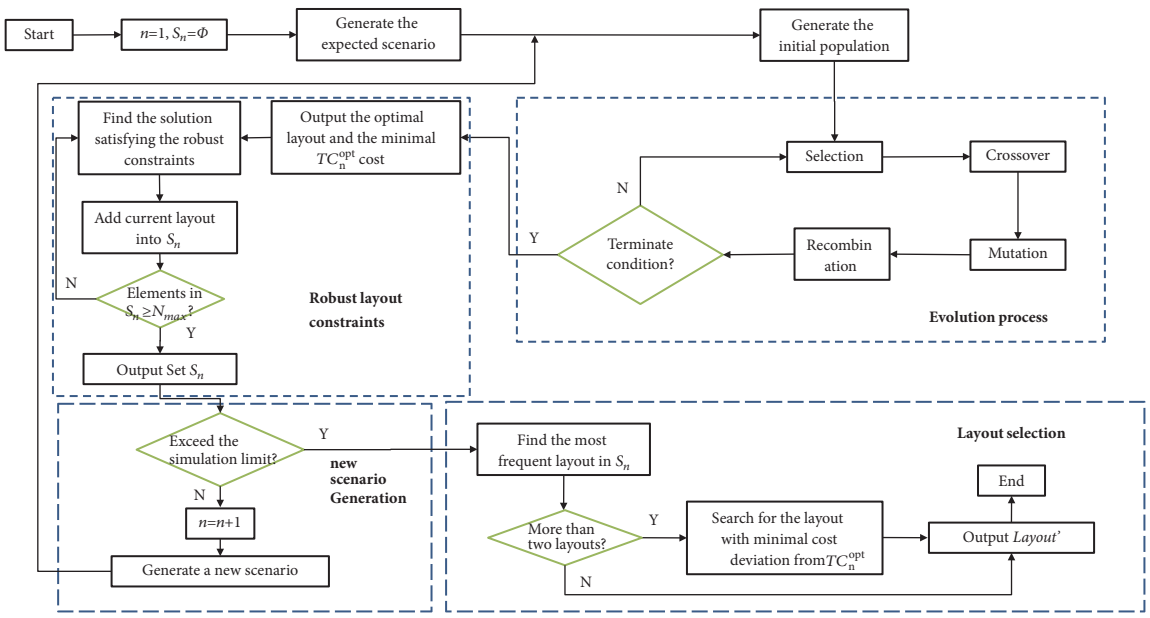

Figure 1: The robust layout generation process based on MCS.

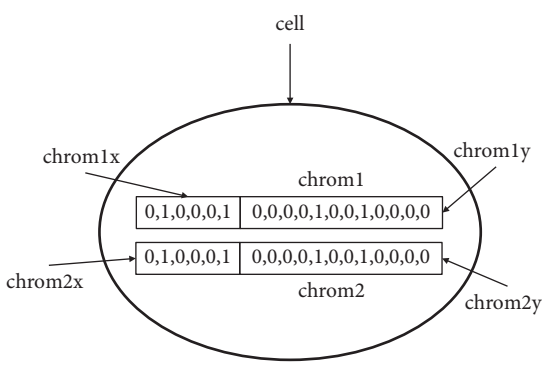

Figure 2: An example of a cell.

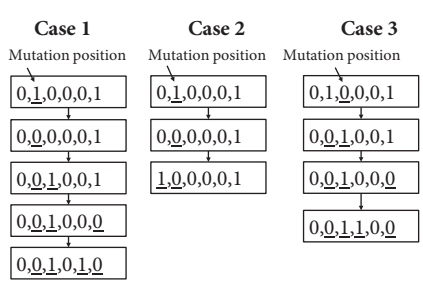

Figure 3: An example of mutation and cross operation of front-end part.

genetic algorithm with the Particle Swarm Optimization (PSO) algorithm proposed in [21]. The parameters are set as follows: the number of populations is 20 and the maximal number of generations is 100. In order to eliminate the influence of the random factors, the average values of the 20 simulation results are compared. A personal computer with

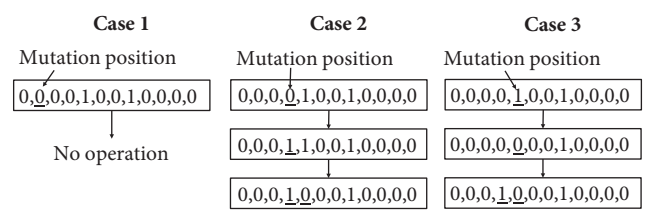

Figure 4: An example of mutation and cross operation of back-end part.

2.7GHz Intel Core i5, 8G RAM, is used to run the algorithms that are coded by matlab2014b. The results are shown in Table 2. It is found that the improved genetic algorithm is superior to the PSO in both solving speed and solution quality in almost all the instances except for instances 8_24_30_10 and 14_32_36_18. Especially for large-scaled instances, the improved genetic algorithm presented in this paper has more advantages.

In order to illustrate the effectiveness of the robust layout proposed in this paper, we compare it with the traditional expected layout (the optimal layout in each period with expected material flow). Cost departure ratio denoted by H is defined to describe the deviation ratio between the total cost of the evaluated layout and the optimal cost in each scenario. H_r represents the robust layout cost deviation ratio, while H_d represents expected layout cost deviation ratio.

$$H = \frac{\sum_{n=1}^{N} TC_n - \sum_{n=1}^{N} TC_n^{opt}}{\sum_{n=1}^{N} TC_n^{opt}} \tag{10}$$

For the stochastic variable material flow Q_{tij}, assume that the expected average value is a, and the fluctuation level

Table 2: Comparison of algorithms.

Instance scale	Improved genetic algorithm		PSO	
$T\_I\_L\_G$	Time(s)	Total cost	Time(s)	Total cost
4_8_12_4	40.4572	140	47.4022	140
4_10_16_6	58.7843	178	68.4300	178
6_14_20_6	94.8301	312	142.0803	314
8_20_26_8	180.2430	478	277.0293	480
8_24_30_10	210.4790	648	312.0019	642
10_24_30_8	260.4341	750	377.0341	750
10_30_36_14	380.4339	1078	500.0391	1090
14_32_36_18	1009.2215	1977	1228.0012	1975
14_38_42_22	1900.0034	2130	2277.1014	2134
20_48_52_32	7708.9988	4701	8792.1044	4710
28_60_68_48	15366.4222	7999	18722.0924	8073
35_70_80_60	23177.7788	9876	26718.8819	9915
40_80_90_70	32421.6644	13077	37877.0701	13203
45_90_100_80	40500.7721	18093	45722.3307	18210
50_100_110_90	56783.4451	24088	60421.7701	24191
55_105_115_95	69803.1111	34330	81445.7769	34700
60_110_120_100	77099.2299	40099	89001.7721	40280
60_115_125_105	84123.7798	46788	93100.1109	46989
65_120_130_110	93401.1194	52103	101012.1981	52708
65_125_135_115	99088.7711	58021	107662.0061	58810

is b. It means that the material flow is in the rage of $[(1-b)*a,(1+b)*a]$. We randomly generate N=200 scenarios, and the prespecified percentage M in robust constraint is 15%. The expected layout is obtained in the scenario when Q_{tij} is equal to a. The cost deviation ratios H_r and H_d are calculated under different fluctuation level in three instances. The results are shown in Table 3.

From Table 3, we can see that, with the increase of fluctuation level, the cost deviation ratio will increase for both robust layout and expected layout. We can conclude that the greater the fluctuation is, the bigger difference between robust/expected layout cost and minimum cost is. Moreover, for all these three different sized instances, the cost deviation ratio of robust layout is much less than expected layout. As the problem size increases, the cost deviation ratio of expected layout raises, while the cost deviation ratio of robust layout reduces. It is obviously concluded that the robust approach proposed in this paper is effective to deal with dynamic layout problem under demand uncertainty, especially for large sized problem.

7. Conclusions

In this paper, an improved genetic algorithm is designed to solve dynamic facility problem considering the assignment of transport devices. The developed algorithm has given near-optimal solutions to different sized case studies randomly generated. In addition, a robust layout is suggested with uncertain material flow in each period. The robust layout is the most frequent one falling within a prespecified percentage of the optimal solution for multiple scenarios generated by Mont Carlo simulation. Although robust layout may not be optimal in any scenario, the deviation from the best cost is small. Its performance is much better than the expected layout through sensitivity analysis.

More uncertain factors except for material flow can be investigated in our future research, such as the unit material handling cost and the flexible routing. Moreover, unequal-sized facilities can be considered and more efficient metaheuristic algorithm needs to be developed in the future.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grants no. 51405283, no. 71401098, and no. 71572104).

Table 3: Comparison of cost deviation ratio.

Fluctuation level	6_14_20_6		20_48_52_32		40_80_90_70	
	H_r	H_d	H_r	H_d	H_r	H_d
1.00%	4.11%	10.75%	2.71%	13.82%	1.72%	15.90%
1.45%	4.08%	10.88%	2.90%	14.50%	1.80%	16.98%
1.90%	4.22%	11.03%	3.11%	15.70%	1.83%	17.88%
2.35%	4.27%	11.77%	3.23%	16.10%	1.88%	19.09%
2.80%	4.29%	12.09%	3.29%	16.77%	1.91%	20.12%
3.25%	4.66%	12.80%	3.30%	17.23%	1.93%	21.98%
3.70%	4.88%	12.91%	3.45%	17.46%	1.94%	22.10%
4.15%	4.91%	13.70%	3.56%	17.56%	1.96%	22.70%
4.60%	4.97%	13.92%	3.71%	18.10%	1.99%	22.90%
5.05%	4.99%	14.10%	3.77%	18.22%	2.01%	23.11%
5.50%	5.01%	14.45%	3.89%	18.28%	2.09%	23.34%
5.95%	5.09%	14.67%	3.91%	18.45%	2.21%	23.56%
6.40%	5.03%	14.72%	3.94%	18.66%	2.32%	23.88%
6.85%	5.21%	14.92%	3.99%	18.88%	2.26%	23.99%
7.30%	5.27%	14.98%	4.01%	19.10%	2.45%	24.67%
7.75%	5.29%	15.21%	4.03%	19.23%	2.48%	24.99%
8.20%	5.33%	15.35%	4.07%	19.45%	2.29%	25.16%
8.65%	5.35%	15.62%	4.11%	19.57%	2.38%	25.37%
9.10%	5.66%	16.11%	4.13%	19.82%	2.44%	25.99%
9.55%	5.70%	16.32%	4.22%	20.03%	2.51%	26.81%
10.00%	5.72%	16.78%	4.25%	20.70%	2.68%	27.32%

References

[1] S. Kulturel-Konak, "Approaches to uncertainties in facility layout problems: Perspectives at the beginning of the 21st Century," *Journal of Intelligent Manufacturing*, vol. 18, no. 2, pp. 273–284, 2007.

[2] S. Emami and A. S. Nookabadi, "Managing a new multi-objective model for the dynamic facility layout problem," *The International Journal of Advanced Manufacturing Technology*, vol. 68, no. 9-12, pp. 2215–2228, 2013.

[3] T. C. Koopmans and M. Beckmann, "Assignment problems and the location of economic activities," *Econometrica*, vol. 25, pp. 53–76, 1957.

[4] G. Moslemipour, T. S. Lee, and D. Rilling, "A review of intelligent approaches for designing dynamic and robust layouts in flexible manufacturing systems," *International Journal of Advanced Manufacturing Technology*, vol. 60, pp. 11–27, 2012.

[5] V. Madhusudanan Pillai, I. B. Hunagund, and K. K. Krishnan, "Design of robust layout for dynamic plant layout problems," *Computers & Industrial Engineering*, vol. 61, no. 3, pp. 813–823, 2011.

[6] G. Moslemipour and T. S. Lee, "Intelligent design of a dynamic machine layout in uncertain environment of flexible manufacturing systems," *Journal of Intelligent Manufacturing*, vol. 23, no. 5, pp. 1849–1860, 2012.

[7] M. J. Rosenblatt, "Dynamics of Plant Layout," *Management Science*, vol. 32, no. 1, pp. 76–86, 1986.

[8] T. A. Lacksonen and E. E. Enscore, "Quadratic assignment algorithms for the dynamic layout problem," *International Journal of Production Research*, vol. 31, no. 3, pp. 503–517, 1993.

[9] T. L. Urban, "A Heuristic for the dynamic facility layout problem," *Institute of Industrial Engineers (IIE). IIE Transactions*, vol. 25, no. 4, pp. 57–63, 1993.

[10] B. K. Kaku and J. B. Mazzola, "A tabu-search heuristic for the dynamic plant layout problem," *INFORMS Journal on Computing*, vol. 9, no. 4, pp. 374–384, 1997.

[11] A. Tayal and S. P. Singh, "Formulating multi-objective stochastic dynamic facility layout problem for disaster relief," *Annals of Operations Research*, 2017.

[12] F. Zarea Fazlelahi, M. Pournader, M. Gharakhani, and S. J. Sadjadi, "A robust approach to design a single facility layout plan in dynamic manufacturing environments using a permutation-based genetic algorithm," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 230, no. 12, pp. 2264–2274, 2016.

[13] D. B. Webster and M. B. Tyberghein, "Measuring flexibility of job-shop layouts," *International Journal of Production Research*, vol. 18, no. 1, pp. 21–29, 1980.

[14] R. M. Gupta, "Flexibility in layouts: A simulation approach," *Material Flow*, vol. 3, pp. 243–250, 1986.

[15] W. K. Chan and C. J. Malmborg, "A Monte Carlo simulation based heuristic procedure for solving dynamic line layout problems for facilities using conventional material handling devices," *International Journal of Production Research*, vol. 48, no. 10, pp. 2937–2956, 2010.

[16] M. J. Rosenblatt and H. L. Lee, "A robustness approach to facilities design," *International Journal of Production Research*, vol. 25, no. 4, pp. 479–486, 1987.

[17] R. Cheng, M. Gen, and T. Tozawa, "Genetic search for facility layout design under interflows uncertainty," in *Proceedings of the 1995 IEEE International Conference on Evolutionary Computation*, pp. 400–405, December 1995.

[18] M. Kaveh, V. M. Dalfard, and S. Amiri, "A new intelligent algorithm for dynamic facility layout problem in state of fuzzy constraints," *Neural Computing and Applications*, vol. 24, no. 5, pp. 1179–1190, 2014.

[19] P. Kouvelis, A. A. Kurawarwala, and G. J. Gutiérrez, "Algorithms for robust single and multiple period layout planning for manufacturing systems," *European Journal of Operational Research*, vol. 63, no. 2, pp. 287–303, 1992.

[20] A. Kheirkhah, H. Navidi, and M. Messi Bidgoli, "Dynamic facility layout problem: a new bilevel formulation and some metaheuristic solution methods," *IEEE Transactions on Engineering Management*, vol. 62, no. 3, pp. 396–410, 2015.

[21] F. Jolai, R. Tavakkoli-Moghaddam, and M. Taghipour, "A multi-objective particle swarm optimisation algorithm for unequal sized dynamic facility layout problem with pickup/drop-off locations," *International Journal of Production Research*, vol. 50, no. 15, pp. 4279–4293, 2012.