

APPLICATION OF MULTI-OBJECTIVE MEMETIC ALGORITHM IN MULTI-OBJECTIVE FLEXIBLE JOB-SHOP SCHEDULING PROBLEM

Zhenwen HU

Huanggang Normal University Department of Computer Science, Huanggang 438000, China
E-mail: jsjhz@hgnu.edu.cn

ABSTRACT: *The flexible job-shop scheduling problem (FJSP) is a hard optimization problem that mirrors the actual situation of production. Currently, many industrial production tasks have more than one objective. To maximize the profit margin, the enterprise must achieve the multiple objectives of each task simultaneously. After analyzing the problems in the FJSP, this paper applies the multi-objective memetic algorithm (MA) in an actual FJSP, and optimizes the solution process of the FJSP with minimal processing span and total cost. The proposed algorithm was compared with the traditional multi-objective optimization algorithm through 15 groups of tests. The comparison shows that our algorithm outperformed the traditional algorithm. This research provides a feasible way to solve multi-objective FJSP.*

KEYWORDS: *Memetic algorithm (MA), multi-objective optimization, flexible job-shop scheduling problem (FJSP), processing span, total cost*

1 INTRODUCTION

The progress in manufacturing puts forward higher requirements on job-shop scheduling: satisfy customer demand for more complex products, while ensuring the profit-making of the job-shop (Remenyi and Sherwood-Smith, 1998). Job-shop scheduling refers to the regulation of the production process (Levinsohn and Petrin, 2003). The job-shop scheduling problem (JSP) is a core issue of the production management in the manufacturing industry (Chang and Peter, 2007). Faced with this problem, production resources should be allocated rationally to ensure smooth and efficient production, and timely measures should be taken in unforeseen circumstances to maintain production safety and optimize the cost allocation. After all, the scheduling plan directly bears on the production benefit of the enterprise (Dimelis, 2002). Since the dawn of industrial mass production, the JSP has been a research hotspot in the field of combinatorial optimization (Fattahi et al., 2007; Giovanni and Pezzella, 2010).

In modern industry, the targets of production scheduling are often flow-shops or job-shops under strict constraints. Currently, many industrial production tasks have more than one objective. To maximize the profit margin, the enterprise must achieve the multiple objectives of each task simultaneously (Li and Huo, 2009; Pintrich, 2000). Besides the strict constraints and multiple objectives, the job-shop scheduling capacity face of the enterprise many other challenges, namely, the

short supply of raw materials, the sudden failure of machines, the cancellation of orders or the insertion of rush orders. Therefore, the JSP is a combinatorial optimization problem with multiple constraints and multiple variables (Deb et al., 2005).

With the increase of production scale and operations, it is increasingly difficult to solve the JSP effectively. The traditional production theory is no longer suitable for the complex production modes (Duncan, 1980). Instead, the flexible job-shop scheduling problem (FJSP) has emerged with the development of modern manufacturing (Brussel et al., 1998; Yan, 2003). To shorten the processing span and enhance enterprise profit, the FJSP strategy should make full use of machines, especially idle ones, in production tasks, without increasing capital input (Kacem et al., 2002; Moslehi and Mahnam, 2011; Pezzella et al., 2008).

In this paper, the traditional multi-objective optimization algorithm is optimized by the memetic algorithm (MA). Then, the optimized algorithm was applied to solve an FJSP. The results prove the feasibility and superiority of our algorithm.

2 FJSP MODELLING

2.1 Problem description

The FJSP asks for minimizing the processing span and total cost of processing N jobs on M machines under a series of constraints, in which each job requires different operations. Most FJSPs have multiple objectives, due to the growing

complexity and variability of modern manufacturing tasks.

There are two ways to solve a multi-objective FJSP, namely, dimensionality reduction and integrated solution. The former simplifies the problem by decomposing it into machine selection and operation sequencing, while the latter improves the production efficiency by adjusting the operation sequence and selecting the machines at the same time.

As shown in Figure 1, the feasible solutions of the FJSP can be illustrated with a disjunctive graph $G=(N, X, Y)$, where N are the nodes (operations), X are the directed edges (operation sequence for the same job; solid arrows), and Y are the disjunctive arcs (operation sequence for the same machine; dotted arrows).

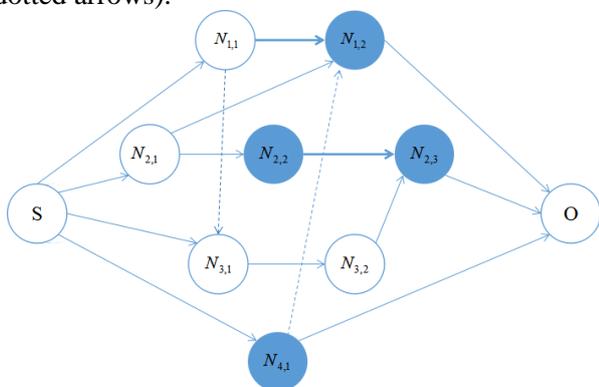


Fig. 1 Disjunctive graph on the feasible solutions of the FJSP

In the disjunctive graph, the longest path between the start and the end of production stands for the processing span. It is an important indicator of the FJSP efficiency, and a key optimization target.

2.2 Theoretical basis of multi-objective evolutionary algorithm

To maximize the benefit in actual production, the job-shop scheduling plan should be optimized for multiple objectives. These objectives may have the relations of mutual promotion or mutual inhibition. However complex the relations are, the ultimate goal of the JSP is to optimize the processing span, total cost and efficiency with various algorithms. Instead of a unique solution, the multi-objective optimization problem has a set of Pareto optimal solutions. The multi-objective optimization problem can be described as:

Objective function:

$$\text{Min } y = f(x) = [f_1(x), f_2(x), \dots, f_n(x)],$$

$$n = 1, 2, \dots, N$$

m inequality constraints:

$$(1)$$

$$g_i(x) \leq 0, i = 1, 2, \dots, m \quad (2)$$

k equality constraints:

$$h_j(x) = 0, j = 1, 2, \dots, k \quad (3)$$

$$x = [x_1, x_2, \dots, x_d, \dots, x_D],$$

$$x_{dmin} \leq x_d \leq x_{dmax}, d = 1, 2, \dots, D \quad (4)$$

where x are D -dimensional decision variables; X is the decision space formed by the decision variables; y are the objective vectors; Y is the objective space formed by the objective vectors. The feasible domain of solutions depends on $g_i(x)$ and $h_j(x)$.

2.3 Multi-objective optimization model

The multi-objective evolutionary algorithm comes to solve multi-objective tasks. It is essentially a genetic algorithm (GA) that converges to optimal solution of a population, by searching and computing multiple individuals in the population. Therefore, this algorithm has been widely applied in multi-objective optimization problems.

To minimize the processing span, the shortest path and time can be derived from the disjunctive graph on the production process. However, no algorithm should optimize towards a single objective in multi-objective optimization. Therefore, another objective, the minimal total cost, must be considered. The total cost of the job-shop equals the sum of the costs of all the nodes in the production process:

$$Q = \sum_1^n N_{i,j}, i \in \{1, 2, L, n\}, j \in \{1, 2, L, n\} \quad (5)$$

After the multiple objectives have been confirmed, the next step is to configure the local search operator of the multi-objective function. The Pareto solution of each search should be assigned a suitable weight, such that high-quality solutions could be found according to the corresponding weight. The objective function of local search can be expressed as:

$$g(x) = m b \quad (6)$$

where m is a Pareto solution; b is the weight assigned to that solution. Thus, the two objectives can be combined into an integrated objective:

$$T = C_{j_i} * b_1 + Q_i * b_2 \quad (7)$$

where C_{j_i} is a set of different paths; Q_i is the cost of path i . Any path in C_{j_i} is less optimal than the shortest path C_j . Therefore, the above integrated objective method is further optimized in the next subsection. The optimized method compares the processing spans and costs of different paths, outputting the optimal results based on the MA.

2.4 MA-based model optimization

The workflow of our multi-objective MA in the FJSP is illustrated in Figure 2 below.

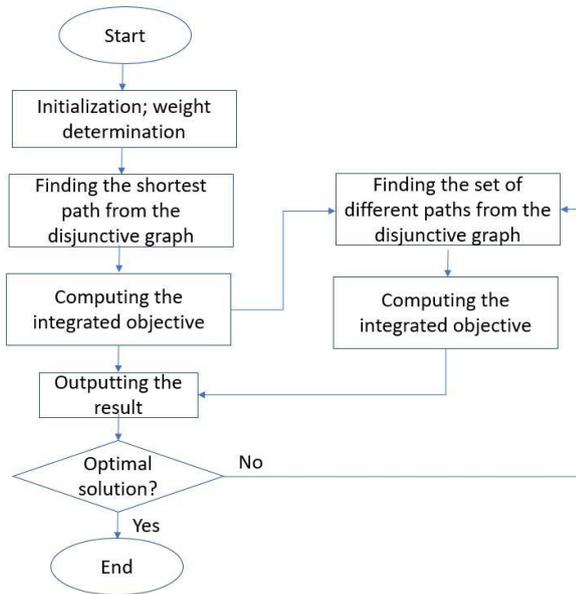


Fig. 2 The workflow of our multi-objective MA in the FJSP

3 EXPERIMENTS AND RESULTS ANALYSIS

3.1 General recommendations

To verify its feasibility and effectiveness, the multi-objective optimization algorithm improved by the MA (our algorithm) was compared with the traditional multi-objective optimization algorithm (the traditional algorithm) through experiments.

The test parameters were configured as follows: the population size, 150; the number of elite solutions, 5; the crossover probability, 0.85; the mutation probability, 0.05; the number of iterations, 50; the intensity of local searches, i.e. the number of evaluations, 500.

A total of 15 tests were carried out, and divided evenly into 5 groups. The test data are listed in Table 1. Both our algorithm and the traditional algorithm were applied to solve the FJSP. To reflect the diversity and convergence, the performance of the two algorithms were evaluated by hypervolume and the number of non-dominated solutions (solution number).

Table 1. Parameter setting

Test	Number of operations	Number of jobs	Number of machines
Test 1 (S01, S02, S03)	15	8	5
Test 2 (S04, S05, S06)	27	8	8
Test 3 (S07, S08, S09)	30	10	10
Test 4 (S10, S11, S12)	48	15	10
Test 5 (S13, S14, S15)	65	15	15

Figure 3 compares the hypervolumes and the solution numbers of the two algorithms in the 15 tests. The comparison shows that our algorithm clearly outperformed the traditional algorithm. This means the MA can effectively enhance the ability of the traditional algorithm to find better solutions to the FJSP.

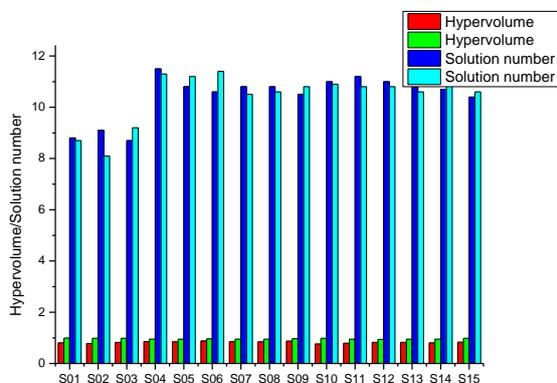


Fig. 3 The hypervolumes and the solution numbers of the two algorithms

The search intensity has a major impact on the performance of our algorithm. If the search intensity is too small, our algorithm cannot fully display its excellence. If the search intensity is too large, the search space will be so narrow as to suppress the search efficiency.

Considering the different scales of the test groups, the most representative dataset of each group was selected to test the performance of our algorithm at different search intensities, aiming to identify the suitable range of search intensity. For small groups S01, S04 and S07, the search intensities were set to 50, 100, 200, 400, 600 and 800, respectively; For large groups S10 and S13, the search intensities were set to 400, 600, 800, 1,000, 1,200 and 1,400, respectively. The test results are displayed in Figures 4 and 5 below.

As shown in Figure 4, with the growth in search intensity, the mean hypervolume of S01 increased first and then decreased, peaking at the search intensity of 600. However, the mean hypervolume of this group varied little, as the search intensity increased from 100 to 600. Therefore, the 100~600 is the suitable range for the search intensity of this

group. For groups S04 and S07, the mean hypervolume exhibited similar trends and both peaked at the search intensity of 600. Hence, the suitable search intensity for these groups falls between 400 and 800.

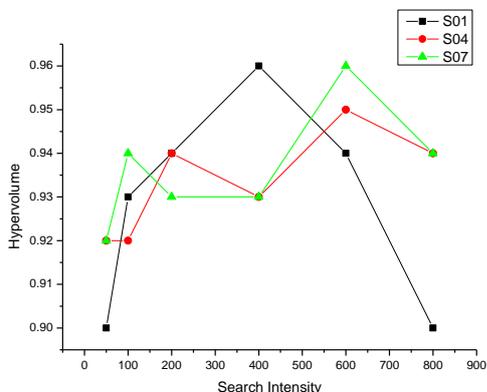


Fig. 4 Test results on small groups

As shown in Figure 5, with the growth in search intensity, the mean values of S10 and S13 both increased first and then decreased, and peaked at the search intensity of 1,200.

The above results show that the suitable search intensity is positively correlated with the scale of search. If the search scale is large, an optimal search intensity can be identified; if the search scale is small, it is only possible to obtain an interval of optimal search intensity.

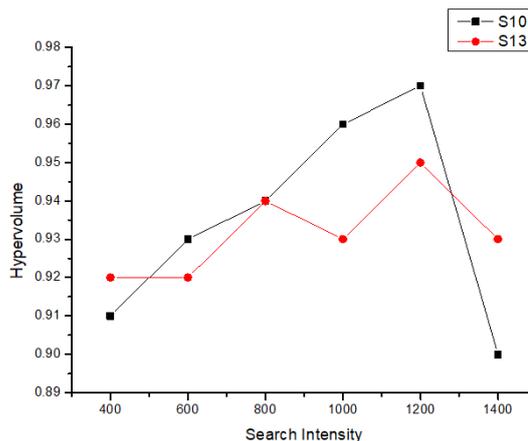


Fig. 5 Test results on large groups

To further prove the superiority of our algorithm, the smallest group S01 and the largest group S13 were selected as FJSPs, and solved by our algorithm and the traditional algorithm. The processing spans and total costs obtained by the two algorithms are displayed in Figure 6 below, where Traditional refers to the traditional algorithm and Memetic refers to our algorithm.

As shown in Figure 6, our algorithm had a clear advantage over the traditional algorithm in both groups, and the advantage increased with the complexity of the multi-objective FJSP.

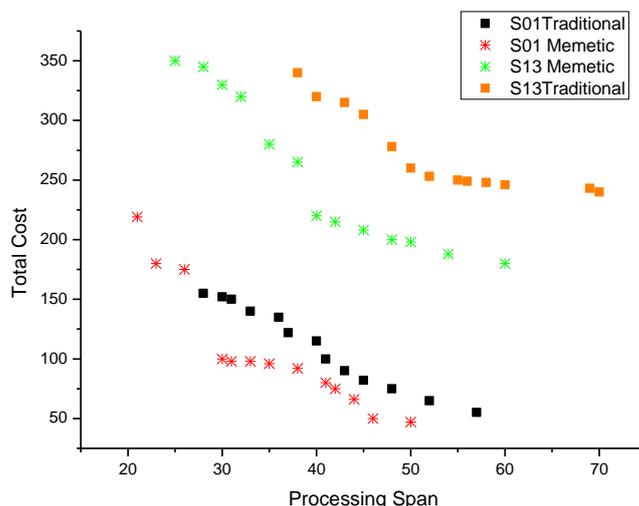


Fig. 6 The processing spans and total costs obtained by the two algorithms

4 CONCLUSIONS

This paper designs an optimization model for the multi-objective FJSP based on the MA, and optimizes the multi-objective optimization algorithm for the FJSP with minimal processing

span and total cost. The main contributions of this research are as follows:

- (1) The features and difficulties of multi-objective FJSP were analyzed in details. The minimal processing span was solved by the disjunctive graph. The multi-objective optimization

algorithm was improved by the MA.

(2) Taking minimal processing span and total cost as the objectives, the author established an optimization model for the JFSP based on the MA, and designed the solution process. The experimental results show that our algorithm greatly outperformed the traditional algorithm and the advantage increased with the population size.

5 REFERENCES

- ▶ Brussel, H. V., Wyns, J., Valckenaers, P., Bongaerts, L., Peeters, P. (1998). Reference architecture for holonic manufacturing systems: *prosa*. *Computers in Industry*, 37(3), 255-274.
- ▶ Chang, T. H., Peter, J. K. (2007). Misallocation and manufacturing tfp in china and india. *MPRA Paper*, 124(4), 1403-1448.
- ▶ Deb, K., Mohan, M., Mishra, S. (2005). Evaluating the ϵ -dominance based multi-objective evolutionary algorithm for a quick computation of pareto-optimal solutions. *Evolutionary Computation*, 13(4), 501-525.
- ▶ Dimelis, S. (2002). Foreign ownership and production efficiency: a quantile regression analysis. *Oxford Economic Papers*, 54(3), 449-469.
- ▶ Duncan, G. M. (1980). Formulation and statistical analysis of the mixed, continuous/discrete dependent variable model in classical production theory. *Econometrica*, 48(4), 839-852.
- ▶ Fattahi, P., Mehrabad, M. S., Jolai, F. (2007). Mathematical modeling and heuristic approaches to flexible job shop scheduling problems. *Journal of Intelligent Manufacturing*, 18(3), 331-342.
- ▶ Giovanni, L. D., Pezzella, F. (2010). An improved genetic algorithm for the distributed and flexible job-shop scheduling problem. *European Journal of Operational Research*, 200(2), 395-408.
- ▶ Kacem, I., Hammadi, S., Borne, P. (2002). Approach by localization and multiobjective evolutionary optimization for flexible job-shop scheduling problems. *IEEE Transactions on Systems Man and Cybernetics Part C (Applications and Reviews)*, 32(1), 1-13.
- ▶ Levinsohn, J., Petrin, A. (2003). Estimating production functions using inputs to control for unobservables. *Review of Economic Studies*, 70(2), 317-341.
- ▶ Li, L., Huo, J. Z. (2009). Multi-objective flexible job-shop scheduling problem in steel tubes production. *Systems Engineering-Theory & Practice*, 29(8), 117-126.
- ▶ Moslehi, G., Mahnam, M. (2011). A pareto approach to multi-objective flexible job-shop scheduling problem using particle swarm optimization and local search. *International Journal of Production Economics*, 129(1), 14-22.
- ▶ Pezzella, F., Morganti, G., Ciaschetti, G. (2008). A genetic algorithm for the flexible job-shop scheduling problem. *Computers and Operations Research*, 35(10), 3202-3212.
- ▶ Pintrich, P. R. (2000). Multiple goals, multiple pathways: the role of goal orientation in learning and achievement. *Journal of Educational Psychology*, 92(3), 544-555.
- ▶ Remenyi, D., Sherwood-Smith, M. (1998). Business benefits from information systems through an active benefits realisation programme. *International Journal of Project Management*, 16(2), 81-98.
- ▶ Yan, H. S. (2003). Practical solution approaches to solve a hierarchical stochastic production planning problem in a flexible automated workshop in china. *IIE Transactions*, 35(2), 103-115.