Int j simul model 16 (2017) 2, 313-321
ISSN 1726-4529 Original scientific paper

MULTI-OBJECTIVE SCHEDULING SIMULATION OF FLEXIBLE JOB-SHOP BASED ON MULTI-POPULATION GENETIC ALGORITHM

Zhang, W.*; Wen, J. B.**; Zhu, Y. C.***,# & Hu, Y.*

* School of Information, Central University of Finance and Economics, Beijing 100081, China

** School of Foreign Studies, Central University of Finance and Economics, Beijing 100081, China

*** School of Economics and Business Administration, Beijing Normal University, Beijing 100875, China

E-Mail: zhuyanchun@bnu.edu.cn (# Corresponding author)

Abstract

In view of the difficulty of obtaining the optimal solution to the multi-objective scheduling of flexible job-shop by the general genetic algorithm, this paper takes into account the shortest processing time and the balanced use of machines, and puts forward the multi-population genetic algorithm based on the multi-objective scheduling of flexible job-shop. The method attempts to minimize the longest make-span of workpieces, the load of each machine, and the total machine load through the overall process scheduling of the job-shop. Research results reveal that the proposed method is highly efficient in seeking the optimal machine allocation chain, and effective in avoiding the complex process of intermediate assignment, making it easier to obtain the said optimal solution. The feasibility and effectiveness of the proposed method are also validated by two instances. Compared with the conventional flexible job-shop scheduling algorithms, the proposed algorithm boasts better population quality, algorithm starting point, and initial expression. Besides, it is far better than other algorithms in terms of the initial solution quality and the convergence rate. Despite the local fluctuations in the early phase of the genetic process, the total machine load and the machine load variance are gradually declining and the curves start to converge after the 50th generation.

(Received, processed and accepted by the Chinese Representative Office.)

Key Words: Flexible Job-Shop, Scheduling, Genetic Algorithm, Multi-Objective, Optimization

1. INTRODUCTION

With the rapid economic development in recent years, large-scale production enterprises are increasingly concerned about centralized and automated production. More and more people are paying attention to improving efficiency in job-shop scheduling, one of the major influencing factors on the production and logistics efficiency of enterprises. Developed on the basis of conventional job-shop scheduling, flexible job-shop scheduling breaks through the limitation that each type of resource must be used for a unique purpose, and designs the optimal process chain to improve the production efficiency of enterprises. Hence, the new dispatching mode has important theoretical and practical significance for the research of flexible job-shop [1-6].

At present, flexible job-shop scheduling basically falls into such three categories as static scheduling, single-objective scheduling, and multi-objective dynamic scheduling. All of the three categories have been tackled by scholars at home and abroad. Betterton and Cox, Chen and Chen analysed static scheduling based on the DBR (Drum-Buffer-Rope) theory [7, 8]; Sun and Xue, Chtourou and Haouari conducted a re-planning of job-shop in view of the dynamic factorial-based fuzzy-stochastic (FFS) problem [9, 10]; Kacem et al. examined flexible job-shop from the angle of the localized genetic algorithm [11]; Xia and Wu proposed a multi-objective flexible job-shop scheduling method that combines the particle swam optimization (PSO) and simulated annealing [12]. Furthermore, multi-objective flexible job-

https://doi.org/10.2507/IJSIMM16(2)CO6

shop dispatching has been explored through the integration of the genetic algorithm with the neighbourhood descent method [13], the combination of the PSO with the Tabu search algorithm [14], and the use of improved algorithms [15-18].

In view of the difficulty of obtaining the optimal solution to the multi-objective scheduling of flexible job-shop by the general genetic algorithm, this paper takes into account the shortest processing time and the balanced use of machines, and puts forward the multi-population genetic algorithm based on the multi-objective scheduling of flexible job-shop. The findings provide a theoretical reference for flexible job-shop dispatching.

2. DESCRIPTION OF THE FLEXIBLE JOB-SHOP DISPATCHING PROBLEM

Let there be *m* workpieces J_i to be processed and *n* machines M_j that can process the workpieces in a job-shop. Suppose J_i has to go through N_i processes. Table I lists the five-step process flexible job-shop scheduling for three workpieces (5×3). The processing time of each workpiece is recorded in the intersections between the corresponding rows O_{ij} and columns M_j.

Table I: Processing time of general flexible job-shop scheduling.

Workpiece	Process	Processing time				
		M_1	M_2	M_3	M_4	M_5
J_1	O_{11}	24	—	16	19	—
	O_{12}	—	21	—	13	11
	O_{13}	—	—	15	14	11
J_2	O_{21}	—	5	—	5	4
	O_{22}	9	—	—	6	8
	O_{23}	—	—	—	4	3
J_3	O_{31}	7	—	—	4	3
	O_{32}	—	—	8	6	6

All workpieces are subject to the following constraints during the processing:

(1) There is a one-to-one correspondence between machine *M* and workpiece *J*; that is, M_j can only process one J_i at a time;

(2) None of the processes O_{ij} can stop halfway;

(3) The processing of J_i must follow the pre-set processing route.

The solution accuracy evaluation indices of flexible job-shop scheduling can be expressed by the following objective functions:

The minimization of the longest make-span T_{max} of a workpiece:

$$T_{\max} = \min\left(\max_{i=1}^{n}\left(\max_{k=1}^{m} t_{ik}\right)\right) \tag{1}$$

The minimization of the running load of each machine W_M:

$$W_M = \min\left(\max_{k=1}^{m} W_k\right) \tag{2}$$

The minimization of the total load of all machines in the job-shop W_T:

$$W_T = \min\left(\sum_{k=1}^{m} W_k\right) \tag{3}$$

In the above formulas, t_{ik} stands for the time it takes for M_k to process J_i, and W_k is the load on the k^{th} machine during the processing.

3. FLEXIBLE JOB-SHOP MULTI-POPULATION GENETIC ALGORITHM

3.1 Algorithm flow

In this paper, the genetic algorithm based on vertical and horizontal double-layer co-evolution is shown in Fig. 1. The algorithm initializes n populations, sorts them out in descending order of fitness, extracts the excellent individuals to form the $(n+1)^{th}$ excellent population, processes all populations synergistically, and selects the optimal solution according to the fitness values thus acquired.

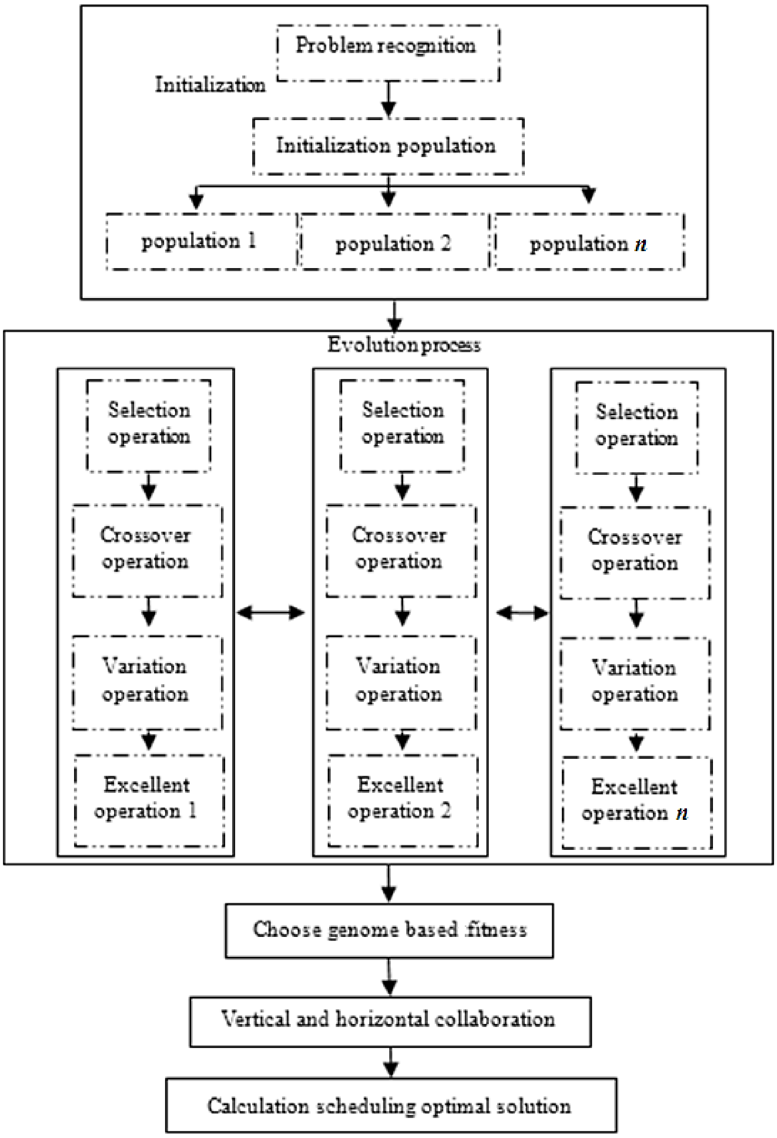

Figure 1: Flow chart of multi-population genetic algorithm.

The specific steps of the proposed algorithm are as follows:

(1) Initialize the parameters. Let there be *b* sub-populations in the system, each of which can generate *p* individuals, define the maximum number of evolutions as *t*, and set the probabilities of cross-operation and variation between different sub-populations as P_c and P_v, respectively. After the evolution of the sub-populations, there will be *b* initial populations.

(2) Let the fitness function be $f(x)$, calculate the $f(x)$ of the individuals in all sub-populations, sort out all the individuals by the fitness value, form the $(n+1)^{th}$ population with

the first *p* individuals, and retain the optimal fitness and the smallest chromosome I in the first *n* populations.

(3) Replace the worst chromosome of each sub-population with I and generate a new sub-population operation. The probability that a sub-population individual is selected can be calculated by the following formula:

$$P_{ij} = f\left(x_{ij}\right) \Big/ \sum_{j=1}^{p} f\left(x_{ij}\right) \tag{4}$$

(4) Perform cross-operation and variation on population individuals according to P_c and P_v. Repeat Steps (1) – (3) for the newly generated populations.

(5) According to the sorting order, select *p* excellent chromosomes to form the population POP1; replace the inferior individuals in the POP1 with the excellent individuals selected from the $(n+1)^{th}$ population and regroup the resulting population into population POP2; calculate the fitness of all individuals in POP2. Terminate the calculation when the optimal individual satisfies the pre-set convergence threshold; otherwise, repeat the iteration process.

3.2 Machine allocation strategy

This section deals with the problem of machine assignment for flexible job-shop scheduling in conjunction with uniform testing. As shown in Fig. 2, the workpiece processing strategy for the machines in the job-shop gives comprehensive consideration to the parameters of the total load and the longest make-span, and finds the initial solution of the population in such three cases as local selection, global selection, and random selection. It defines the length of the array of machine make-spans such as the number of machines, and the array elements such as the sum of the make-spans of all workpieces being processed. The length of the array of machine make-spans is then updated using the shortest machine make-span as the benchmark. On the whole, the method proposed in this paper mainly considers the shortest processing time and the balanced use of a machine, and aims at minimizing the longest make-span of a workpiece, the running load of each machine, and the total load of all machines through the overall process scheduling of the job-shop.

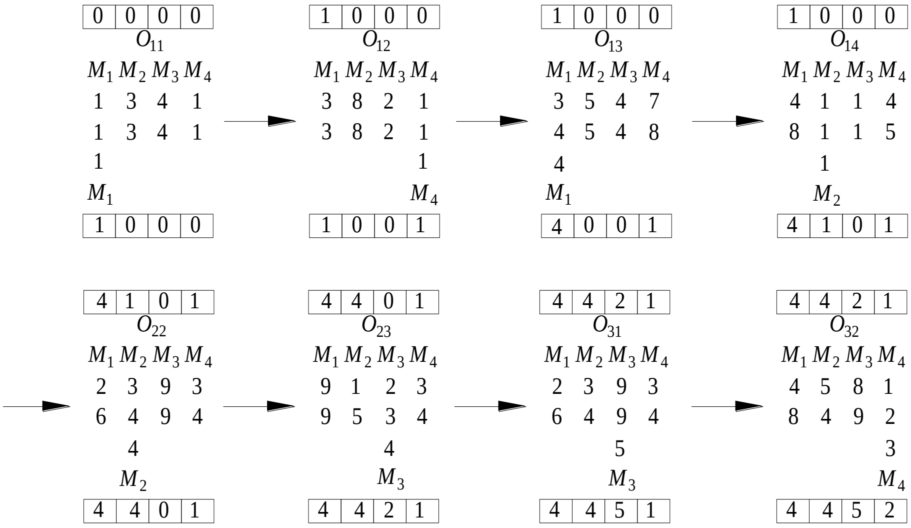

Figure 2: Flow chart of machine allocation strategy.

Fig. 3 is the Gantt chart for 3×4 flexible job-shop scheduling, where the shaded areas are empty rectangles, and the number-containing rectangles describe the processing steps. The Gantt chart visually displays the features of the work of the flexible job-shop. The total load of the job-shop is minimized when the total area of the process rectangles of all machines reaches the lowest point. The strategy of the shortest make-span refers to the reduction of the size of each rectangle. The closer the areas of the rectangles, the more balanced the load distribution in the job-shop. T_{max} is related to the latest make-span in the figure: the more left-leaning the position of the make-span, the smaller the value of T_{max}. The selection of a machine determines the area of the corresponding process rectangles. At this time, it is necessary to adjust the processing sequence so as to reduce the total load of the job-shop. Taking the total load of the job-shop and the balanced load of each machine as the evaluation indices, the author sets up a machine allocation chain, thereby converting the machine allocation problem into a multi-objective combinatorial optimization of the flexible job-shop.

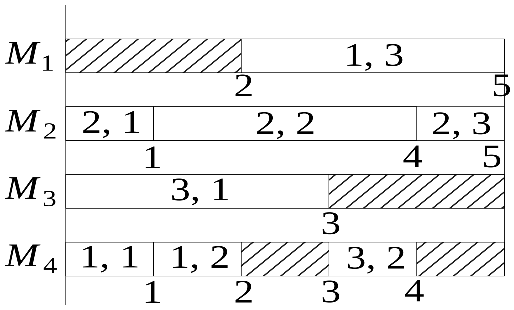

Figure 3: Gantt chart of flexible job-shop scheduling.

During the initialization of the machines in the flexible job-shop, the machine with the shortest make-span is the only one that is taken into account. Taking the shortest make-span as the horizontal reference, the author expands the search scale based on variation operation. Two instances of the machine combination problem, 10×10 and 15×10, are converted to test models for solving the uniform calculation. The relevant parameters are shown in Table II. DPS software is used for the uniform design of the flexible job-shop.

Table II. Test models of uniform design.

Number	Instance	Number of elements	Level figure
1	10×10	20	3
2	15×10	10	2

The random weight method is used to find the total load of all machines in the job-shop and the machine load variance. The fitness function after the combination and arrangement of weights can be expressed as:

$$F_p = M - \sum_{q=1}^{k} \omega_{p,q} \cdot f_q \tag{5}$$

where M is a sufficiently large positive number; f_q is the objective function; $\omega_{p,q}$ is the weight coefficient. During the calculation, the excellent individuals are selected from the parent population and inputted into the next generation based on the genetic algorithm. The chain of operation sequence is determined in light of the linear order and the priority crossover algorithm.

4. EXAMPLE VALIDATION AND ANALYSIS

The 10 × 10 and 15 × 10 instances (Table II) are resolved and analysed to validate the effectiveness of the multi-objective population genetic algorithm of flexible job-shop scheduling proposed in this paper. In the meantime, the proposed method is also contrasted with two conventional algorithms, i.e. the stochastic algorithm and the Kacem algorithm, for accuracy verification. The population size of the flexible job-shop is put at 2,500; the number of initial individuals in the proposed method, stochastic algorithm, and Kacem algorithm is set as 2,000, and the remaining 500 individuals are regarded as the initial values for all processing machines; the maximum number of generations is defined as 100; $P_c = 0.65$; $P_v = 0.15$. In order to ensure the accuracy of the calculated results and reduce the rate of accidental errors, the 10 × 10 and 15 × 10 instances are repeatedly calculated fifteen times, and the 15 calculated results are averaged to yield the final results of each instance. Fig. 4 is the Gantt chart of the 10 × 10 flexible job-shop scheduling instance.

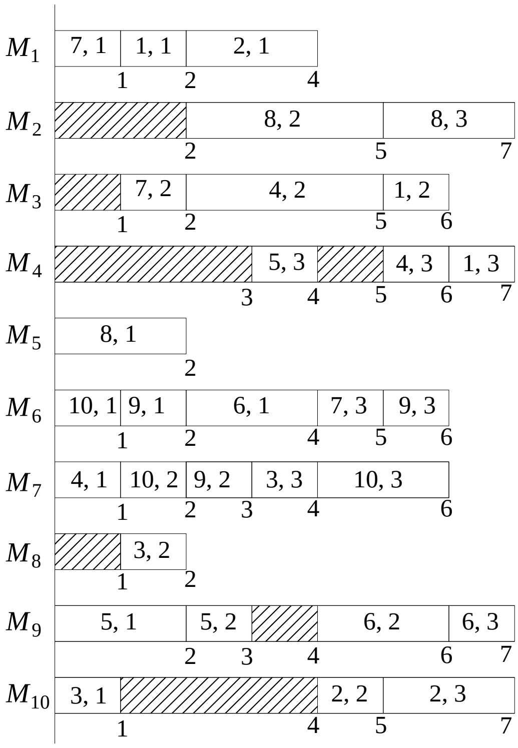

Figure 4: Gantt chart of 10 × 10 flexible job-shop scheduling.

Fig. 5 is the T_{max} contrast histogram for the first and final iterations of the three algorithms in the 10 × 10 instance. On the x-axis, 1 and 2 represent the optimal and mean values of T_{max} in the first generation of the genetic algorithm, respectively; 3 and 4 represent the optimal and mean values of T_{max} in the 100th generation of the genetic algorithm, respectively. As shown in the figure, the proposed algorithm boasts better population quality, algorithm starting point, and initial expression than the stochastic algorithm and the Kacem algorithm, no matter whether in the first generation or the 100th generation.

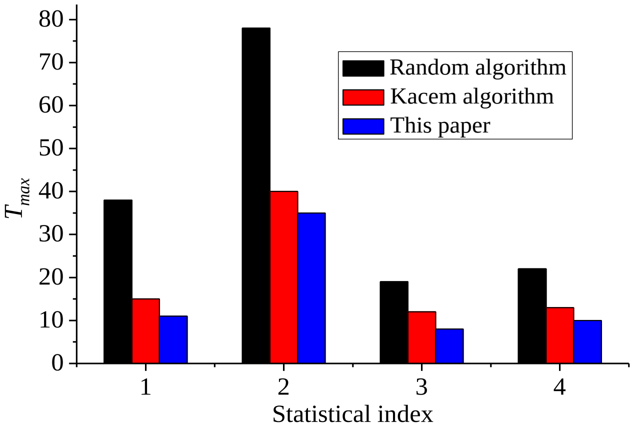

Figure 5: Statistical indices of the 10 × 10 instance of the 3 algorithms.

Table III displays the final results of the 10 × 10 and 15 × 10 instances by the 3 algorithms. The proposed method is proved to be effective for it has obtained the best test results for the three objective functions of T_{max}, $W_{T,}$ and W_M.

Table III: Optimal results comparison of the 3 algorithms.

Algorithm	10 × 10			15 × 10		
	T_{max}	W_T	W_M	T_{max}	W_T	W_M
Random	7	45	6	13	93	11
Kacem	7	43	5	12	91	11
The proposed algorithm	6	42	5	11	91	11

Fig. 6 presents the T_{max} convergence curves of the 10 × 10 instance by the three algorithms. The x-axis refers to the number of generations. In the first 30 generations, the scholastic algorithm boasts the longest make-span with the Kacem algorithm trailing far behind. The T_{max} of the proposed algorithm, however, always stays at a very low level. Besides, the proposed algorithm is far better than the other algorithms in terms of initial solution quality and convergence rate.

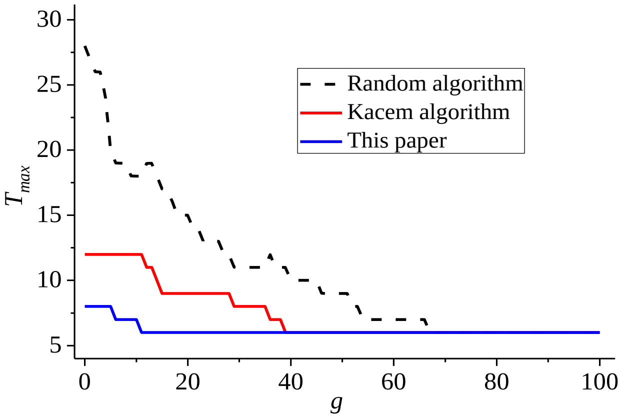

Figure 6: T_{max} convergent curves of 10 × 10 instance by the 3 algorithms.

Fig. 7 shows how the total load of all machines in the job-shop W_T and the machine load variance D change with the increasing number of generations in the 10 × 10 instance when the proposed algorithm is applied to the case of randomly generated populations and selected machines.

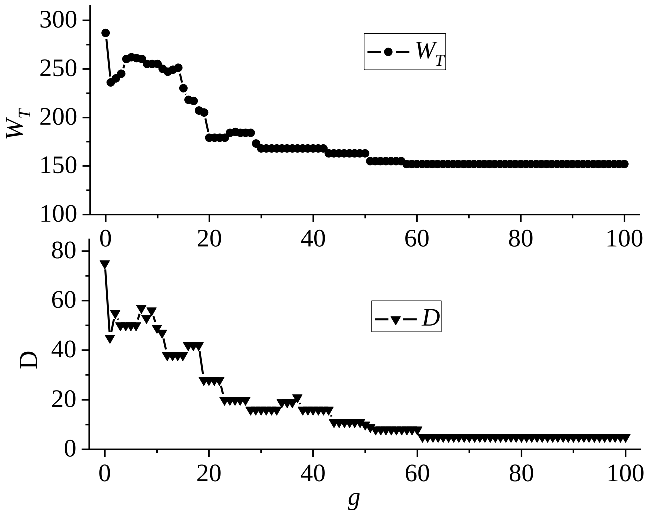

Figure 7: Total load and variance curves of the 10 × 10 instance.

It can be seen from the figure that W_T and D fluctuate in local areas during the early phase of the genetic process, but the curves gradually decline and start to converge after the 50th generation. This once again proves the validity of the proposed method.

5. CONCLUSION

In view of the difficulty of obtaining the optimal solution to the multi-objective scheduling of flexible job-shop by the general genetic algorithm, this paper considers factors like the shortest processing time and the balanced use of machines, and develops the multi-population genetic algorithm based on the multi-objective scheduling of flexible job-shop. The conclusions are given below.

(1) The proposed method is highly efficient in seeking the optimal machine allocation chain, and effective in avoiding the complex process of intermediate assignment, making it easier to obtain the said optimal solution. In addition, the overall process scheduling of the job-shop has successfully minimized the longest make-span of each workpiece, the load of each machine, and the total machine load.

(2) The feasibility and effectiveness of the proposed method is validated by two instances. Compared with the conventional flexible job-shop scheduling algorithms, the proposed algorithm boasts better population quality, algorithm starting point, and initial expression. Besides, it is far better than other algorithms in terms of the initial solution quality and the convergence rate. Despite the local fluctuations in the early phase of the genetic process, the total machine load and the machine load variance are gradually declining and the curves start to converge after the 50th generation.

REFERENCES

[1] Gao, J.; Gen, M.; Sun, L.; Zhao, X. (2007). A hybrid of genetic algorithm and bottleneck shifting for multiobjective flexible job shop scheduling problems, *Computers & Industrial Engineering*, Vol. 53, No. 1, 149-162, doi:10.1016/j.cie.2007.04.010

[2] Huang, R.-H. (2010). Multi-objective job-shop scheduling with lot-splitting production, *International Journal of Production Economics*, Vol. 124, No. 1, 206-213, doi:10.1016/j.ijpe.2009.10.026

[3] Lin, R.; Liao, C.-J. (2012). A case study of batch scheduling for an assembly shop, *International Journal of Production Economics*, Vol. 139, No. 2, 473-483, doi:10.1016/j.ijpe.2012.05.002

[4] Jeong, H.; Woo, S.; Kang, S.; Park, J. (1997). A batch splitting heuristic for dynamic job shop scheduling problem, *Computers & Industrial Engineering*, Vol. 33, No. 3-4, 781-784, doi:10.1016/s0360-8352(97)00252-0

[5] Wang, H. (2005). Flexible flow shop scheduling: optimum, heuristics and artificial intelligence solutions, *Expert Systems*, Vol. 22, No. 2, 78-85, doi:10.1111/j.1468-0394.2005.00297.x

[6] Gong, D.; Tang, M.; Liu, S.; Li, Q. (2017). Reconsidering production coordination: A principal-agent theory-based analysis, *Advances in Production Engineering & Management*, Vol. 12, No. 1, 51-61, doi:10.14743/apem2017.1.239

[7] Betterton, C. E.; Cox III, J. F. (2009). Espoused drum-buffer-rope flow control in serial lines: a comparative study of simulation models, *International Journal of Production Economics*, Vol. 117, No. 1, 66-79, doi:10.1016/j.ijpe.2008.08.050

[8] Chen, C. L.; Chen, C. L. (2009). A bottleneck-based heuristic for minimizing makespan in a flexible flow line with unrelated parallel machines, *Computers & Operations Research*, Vol. 36, No. 11, 3073-3081, doi:10.1016/j.cor.2009.02.004

[9] Sun, J.; Xue, D. (2001). A dynamic reactive scheduling mechanism for responding to changes of production orders and manufacturing resources, *Computers in Industry*, Vol. 46, No. 2, 189-207, doi:10.1016/s0166-3615(01)00119-1

[10] Chtourou, H.; Haouari, M. (2008). A two-stage-priority-rule-based algorithm for robust resource-constrained project scheduling, *Computers & Industrial Engineering*, Vol. 55, No. 1, 183-194, doi:10.1016/j.cie.2007.11.017

[11] Kacem, I.; Hammadi, S.; Borne, P. (2002). Pareto-optimality approach for flexible job-shop scheduling problems: hybridization of evolutionary algorithms and fuzzy logic, *Mathematics & Computers in Simulation*, Vol. 60, No. 3-5, 245-276, doi:10.1016/s0378-4754(02)00019-8

[12] Xia, W.; Wu, Z. (2005). An effective hybrid optimization approach for multi-objective flexible job-shop scheduling problems, *Computers & Industrial Engineering*, Vol. 48, No. 2, 409-425, doi:10.1016/j.cie.2005.01.018

[13] Gao, J.; Sun, L.; Gen, M. (2008). A hybrid genetic and variable neighborhood descent algorithm for flexible job shop scheduling problems, *Computers & Operations Research*, Vol. 35, No. 9, 2892-2907, doi:10.1016/j.cor.2007.01.001

[14] Zhang, G.; Shao, X.; Li, P.; Gao, L. (2009). An effective hybrid particle swarm optimization algorithm for multi-objective flexible job-shop scheduling problem, *Computers & Industrial Engineering*, Vol. 56, No. 4, 1309-1318, doi:10.1016/j.cie.2008.07.021

[15] Li, J.-Q., Pan, Q.-K.; Liang, Y.-C. (2010). An effective hybrid tabu search algorithm for multi-objective flexible job-shop scheduling problems, *Computers & Industrial Engineering*, Vol. 59, No. 4, 647-662, doi:10.1016/j.cie.2010.07.014

[16] 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*, Vol. 129, No. 1, 14-22, doi:10.1016/j.ijpe.2010.08.004

[17] Jia, H. Z.; Nee, A. Y. C.; Fuh, J. Y. H.; Zhang, Y. F. (2003). A modified genetic algorithm for distributed scheduling problems, *Journal of Intelligent Manufacturin*g, Vol. 14, No. 3-4, 351-362, doi:10.1023/A:1024653810491

[18] Wang, J. F.; Du, B. Q.; Ding, H. M. (2011). A genetic algorithm for the flexible job-shop scheduling problem, Shen, G.; Huang, X. (Eds.), *Advanced Research on Computer Science and Information Engineering, Part I (CCIS 152)*, Springer-Verlag, Berlin, 332-339