A Batch Splitting Job Shop Scheduling Problem with bounded batch sizes under Multiple-resource Constraints using Genetic Algorithm

WANG Hai-yan, ZHAO Yan-wei*
Key Laboratory of Mechanical manufacture and Automation of Ministry of Education
Zhejiang University of Technology
Hangzhou, P. R. China
(*Corresponding author, e-mail: zyw@zjut.edu.cn)

XU Xin-li, WANG Wan-liang
College of Information Engineering
Zhejiang University of Technology
Hangzhou, P. R. China

***Abstract*—Considering alternative resources for operations, requirement of multiple resources to process an operation and a job's batch size greater than one in the real manufacturing environment, a study is made on the batch splitting scheduling problem with bounded batch sizes under multiple-resource constraints, based on the objective to minimize the maximum completion time. A genetic algorithm which is suitable for this problem is proposed, with a new chromosome representation, which takes into account the batch splitting of the original batches of jobs. And a new crossover method and a new mutation method are proposed based on the new chromosome representation. The results of the simulation indicate that the method is feasible and efficient.**

***Keywords*—batch splitting scheduling, multiple-resource constraints, genetic algorithm, fuzzy processing time**

I. INTRODUCTION

The job shop scheduling problem has been a major target in manufacturing systems. Approaches to this problem have been studied a lot and great results have been achieved, especially in the field of intelligent algorithms as Genetic Algorithm(GA), Neural Network(NN) and Fuzzy Logic (FL) [1,2,3] . Most of the existing researches about the job shop scheduling problem was of single-resource constrained (machines only), and assumed that a job consisted of only a single part. In the real manufacturing environment, a job consists of a batch size greater than one, and there may require multiple resources to process an operation and may exist alternative resources for operations[2]. By splitting the original batch into many smaller batches, these smaller batches can be processed on different resources at the same time, thus the production efficiency can be improved.

In the book of Wang Wanliang's in 2007, the job shop scheduling problem in electroacoustic corporations was studied, and a model for multiobjective jobshop scheduling with fuzzy processing time and fuzzy duedate under multiple-resource constraints was presented[3]. Eimaraghy H, Patel V and Abdallah I B studied the scheduling problem in manufacturing systems constrained by both machines and workers, and presented a genetic algorithm with a new chromosome representation, which took into account machine and worker assignments to jobs[4]. Christoph S. Thomalla converted a multiple-resource problem to a single-resource problem by grouping multiple resources into many groups that may process an operation, and every group was regarded as a single machine[5]. Scheduling problems with alternative resources were studied in many researches[6,7]. Li Fu-Ming et al.(2006) adopted a parallel chromosome coding method, one for the machine allocation, the other for the scheduling of operations[7]. There also exists many researches about the batch-splitting scheduling problems[8,9,10]. Gur Mosheiov and Daniel Oron studied batch scheduling problem with a lower bound that all batch sizes shared and an upper bound that all batch sizes could not exceed [10].

This paper addresses a batch splitting job shop scheduling problem with bounded batch sizes, considering multiple-resource constraints and alternative resources for operations. And a solution to this problem should consist of the number of batches and their sizes and resources allocated for those batches and their start and completion times.

II. PROBLEM FORMULATION

Consider several jobs to be processed and some processing resources in a manufacturing system. Each job consists of a batch size greater than one and a given number of operations to be processed in a specific order. In this scheduling problem, operations may require multiple resources for processing with different processing times on each resource and there may exist alternative resources for operations. Here, resources refer to machines, moulds and workers. We try to get an improved schedule by splitting the original batch into smaller batches according to the algorithm we are proposing.

In the real manufacturing environment, various factors involved in the problems are often imprecisely known to the researchers. Here we formulate a batch-splitting job shop scheduling problem with bounded batch sizes constrained by machines, moulds and workers with fuzzy processing time using the following variables:

J_i i th job

AM_i the original batch size of job J_i

OP_{ij} j th operation of the i th job, whose relationship with W is $OP_{ij} \in W$

N number of jobs

n_i number of operations of job J_i

W_i i th operation in W which contains all the operations that can be processed in the workshop

Q number of operations in W

$$ER_i = \begin{cases} 1, \text{ if machines are required to process } W_i \\ 0, \text{ if machines are not required to process } W_i \end{cases}$$

$$MR_i = \begin{cases} 1, \text{ if moulds are required to process } W_i \\ 0, \text{ if moulds are not required to process } W_i \end{cases}$$

E_i i th machine in E which contains all the available machines in the workshop

Because the relationship between a machine and a worker who manipulates it is normally fixed in a shift, here E_i includes a worker who manipulates the machine. Therefore, for operations that need machines, allocating machines for the process of these operations means allocating both machines and workers for the process of these operations.

H number of machines in E

M_i i th mould in M which contains all the available moulds in the workshop

A mould can't be able to work until it is installed on a machine. And a mould that can be used to process an operation can be installed on any machines that process the same operation. When the process of an operation does not need machine, it obviously does not need mould.

F number of moulds in M

P_i i th worker in P which contains all the available workers in the workshop except for those that are included in E.

Obviously, workers are required to be allocated for the process of operation W_i when $MR_i = 0$ and $ER_i = 0$.

G number of persons in P

$$W\_M_{ij} = \begin{cases} 1, \text{ if mould } M_j \text{ can be used to process } W_i \\ 0, \text{ if mould } M_j \text{ can't be used to process } W_i \end{cases}$$

$$W\_E_{ij} = \begin{cases} 1, \text{ if operation } W_i \text{ can be processed on } E_j \\ 0, \text{ if operation } W_i \text{ can't be processed on } E_j \end{cases}$$

$$W\_P_{ij} = \begin{cases} 1, \text{ if person } P_j \text{ can process operation } W_i \\ 0, \text{ if person } P_j \text{ can't process operation } W_i \end{cases}$$

Since workers in P deal with operations that don't need machines as mentioned above, $W\_P_{ij} = 1$ may occur only when $MR_i = 0$ and $ER_i = 0$.

$ET_i = (et_i^L, et_i^M, et_i^U)$ processing time of machine E_i for a single part

$PT_i = (pt_i^L, pt_i^M, pt_i^U)$ processing time of worker P_i for a single part

We can get the processing times of operations that need machines from ET and the processing times of operations that don't need machines from PT.

$B\_\min$ the lower bound of batch size

$B\_\max$ the upper bound of batch size

To simplify this scheduling problem, we assume that all jobs and resources are available for processing at time zero. Furthermore, several additional constraints as follows have to be met:

1) Every small batch, thanks to the batch splitting of the original batch, is regarded as a single part. When scheduling, every small batch should not precede the subsequent operations when the predecessor has not been finished.

2) Only one operation can be processed on each resource at a time.

3) There is no precedence constraint between operations from different jobs.

4) When a small batch of a job is still being processed, it can't be interrupted until this whole batch is finished.

III. OPERATIONS ON FUZZY PROCESSING TIME

Considering a number of man-made factors that exist in operations, in the scheduling problem considered in this paper, the fuzzy processing time is represented by a triangular fuzzy number(TFN)[11] as shown in Fig.1, where t^L 、 t^M 、 t^U stands for the optimistic value, the most likely value and the pessimistic value respectively.

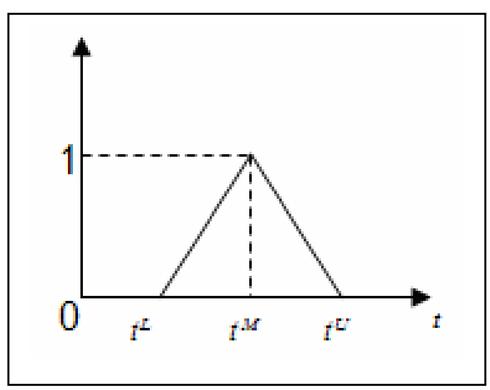

Figure 1. Triangular Fuzzy Number(TFN)

In the scheduling problem formulated in section 2, there exists precedence constraints between operations of a job and resources (machines, moulds, workers in this paper) constraints. When calculating the fuzzy starting time for each operation, we need $\vee$(max) operation between two triangular fuzzy numbers. And after we get the fuzzy starting time for an operation, we use the addition of two triangular fuzzy numbers to calculate the fuzzy completion time of the operation. The $\vee$ (max) and the addition of two triangular fuzzy numbers $t = (t^L, t^M, t^U)$ and $t' = (t^{L'}, t^{M'}, t^{U'})$ are shown by the following formulas[12]:

$$t \vee t' = (t^L \vee t^{L'}, t^M \vee t^{M'}, t^U \vee t^{U'})$$

$$t + t' = (t^L + t^{L'}, t^M + t^{M'}, t^U + t^{U'})$$

To minimize the maximum completion time, as is the objective in this scheduling problem, ranking method is necessary to rank the fuzzy completion times to get the maximum completion time. We adopt the following three criterions to rank two triangular fuzzy numbers t and t' [12]:

Criterion 1. $C1(t) = t^L + 2 * t^M + t^U$

If criterion 1 dose not rank the two triangular fuzzy numbers, try the second criterion:

Criterion 2. $C2(t) = t^M$

If criterion 1 and criterion 2 do not rank the two triangular fuzzy numbers, try the third criterion:

Criterion 3. $C3(t) = t^U - t^L$

IV. AN IMPROVED GENETIC ALGORITHM

In this scheduling problem, a job's batch size is probably greater than one, as in the real manufacturing environment. By splitting the original batch into many smaller batches, and regarding every smaller batch as a single part, we can process these smaller batches on different resources at the same time, and the idle times of resources with jobs which have smaller processing times by splitting the original batches can be filled up[8]. Thus, an improved schedule can be obtained. A genetic algorithm which is suitable for solving the formulated problem is proposed.

A. *Individual representation*

In creating the chromosomes, we adopt parallel chromosome coding method, which can be seen from table Ⅰ and table Ⅱ. Genes in one chromosome represent operations of the jobs, whereas genes in the other represent batch sizes of operations in the former chromosome. We regard those two chromosomes as two aspects of one chromosome. We can see the original batch size of job 1 is 20 from table Ⅰ, and we can get that the second and the third genes in the chromosome are two batches of the first operation of job 1, because this is the first time that the addition of the two batch sizes equals the original batch size of job 1. The fifth gene in the chromosome represents the second operation of job 1, as the same thing mentioned just now appears the second time.

TABLE I. AN EXAMPLE OF SCHEDULING

Job	Original batch size	Operation	
Job1	20	O1	O2
Job2	15	O1	---

TABLE II. A CHROMOSOME FOR THE PROBLEM IN TABLE Ⅰ

Operation	2	1	1	2	1
Batch Size	7	10	10	8	20

A schedule can be obtained from a chromosome according to the following steps:

1) Set the fuzzy idle times of all the resources to zero: $END_i = (0,0,0)$, $MND_i = (0,0,0)$ and $PND_i = (0,0,0)$. END , MND and PND are the fuzzy idle times of machine, moulds and workers respectively.

2) Find a gene that has not been scheduled from left to right in the chromosome. If the operation the gene represents needs mould, chose the mould that has the earliest idle time. If the operation needs machine (or worker), chose the machine (or worker) that completes the whole batch of the operation with the earliest completion time.

3) Calculate the completion time of the operation. And adjust the idle times of the chosen resources to the completion time of the operation.

4) Repeat steps 2~3, until all the operations in the chromosome are scheduled.

5) Evaluate the fitness.

B. *Method for generating initial population*

An initial population consists of a number of chromosomes such as the chromosome in section 4.1. Random number generation is applied to keep the diversity in the initial population. Lengths of two chromosomes may be different and may be very long or very short due to randomicity of the method. Considering that batch sizes may be bounded, as in many practical situations[10], here we set lower bound $B\_\min$ and upper bound $B\_\max$ of batch size to make length of a chromosome suitable. All genes' batch sizes in chromosomes cannot be less than $B\_\min$ or exceed $B\_\max$.

C. *The fitness function*

The objective in this scheduling problem is to minimize the maximum completion time. Assume that the maximum completion time is $C_{\max} = \{C^L_{\max}, C^M_{\max}, C^U_{\max}\}$. When calculating the fitness of a chromosome, we use the most likely completion time $C^M_{\max}$: $Fitness = \frac{1}{C^M_{\max}}$

D. Selection operator

We adopt roulette wheel selection[13,14] method to select individuals to process crossover and mutation in this paper. The best individual in the generation is protected from crossover and mutation.

E. Crossover operator

Based on the new chromosome representation introduced in section 4.1, a new crossover method is proposed. If two parents are chosen, follow the following steps when processing the crossover procedure:

1) Randomly chose a gene in parent 1.

2) Find two segments from parent 1 and parent 2 respectively, that have the shortest length but contain all the genes having the same job and the same operation as the chosen gene's. Sign the segment from parent 1 as segment 1 and the other as segment 2.

3) Compare segment1 with segment 2.

(1) If the total batch size of an operation of a job in segment1 is more than it is in segment 2. Subtract the difference of the total batch size from batch sizes of genes of the same job and the same operation in segment 1. If a gene's batch size is smaller than $B\_\min$, batch sizes of the other genes in segment 1 that are of the same job and the same operation share this gene's batch size alike, and this gene are removed from segment 1.

(2) If the total batch size of an operation of a job in segment1 is less than it is in segment 2. If the difference of the total batch size is less than $B\_\min$, batch sizes of other genes in segment 1 that are of the same job and the same operation share the difference alike. If the difference of batch size is between [$B\_\min$, $B\_\max$], randomly chose a location in segment 1 and insert a gene with the same job and the same operation and a batch size of the difference. If the difference of batch size exceeds $B\_\max$, randomly generate a number between [$B\_\min$, $B\_\max$] and randomly select a location in segment 1 and insert a gene with the same job and the same operation and a batch size of the number generated. And this is repeated until the difference of the batch size is added into segment 1 totally.

4) The same processes in step 3 are done to segment 2 comparing with segment 1.

5) Replace the original segment 1 in parent 1 with the new segment 2, and offspring 1 is obtained. Replace the original segment 2 in parent 2 with the new segment 1, and offspring 2 is obtained.

Fig.2 illustrates the crossover procedure of the scheduling problem in table Ⅰ.

F. Mutation operator

If one parent is chosen, follow the following steps when processing the mutation procedure:

1) Randomly chose a gene in the parent, signed as gene 1. Find a segment that have the shortest length but contain all the genes having the same job and the same operation as gene 1's.

2) Randomly find a gene in the segment, signed as gene 2. If gene 2 is of the same job as gene 1's, randomly change batch sizes of gene 1 and gene 2 within [B_min, B_max], on the condition that the addition is changeless. Else, move gene 1 to where gene 2 is, and gene 2 is moved to the next, as is showed below.

Fig.3 illustrates the mutation procedure of the scheduling problem in table Ⅰ, based on the new chromosome representation.

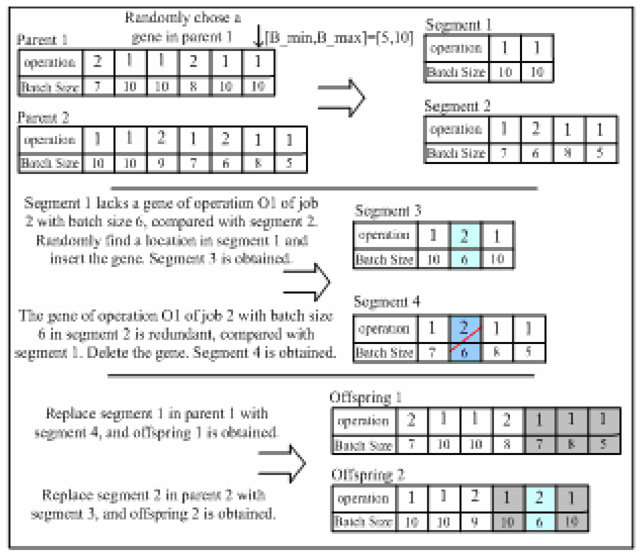

Figure 2. Crossover procedure

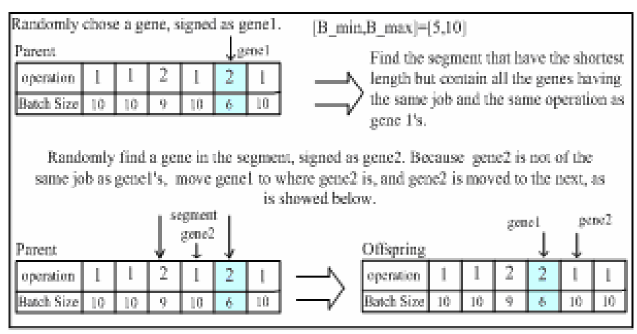

Figure 3. Mutation procedure

V. THE EXPERIMENT AND ANALYSIS

The information of the jobs, resources and the relationships between operations and resources of an example of scheduling are shown in table Ⅲ, table Ⅳ and table Ⅴ.

TABLE III. JOB INFORMATION

Job	Original batch	Bounded batch sizes	Operations				
			1	2	3	4	5

J_1	150	[50,100]	O_2	O_3	O_1	---	---
J_2	150		O_4	O_5	O_6	---	---
J_3	200		O_1	O_3	O_5	O_4	---
J_4	200		O_1	O_2	O_3	O_4	---
J_5	200		O_6	O_5	O_4	O_3	---
J_6	200		O_1	O_2	O_4	O_3	O_5

TABLE IV. RESOURCES INFORMATION

mark symbol	E_1	E_2	E_3	E_4	E_5	E_6	E_7	E_8
resource	Machines							
Processing time for a single part	3,4,5	5,7,9	1,2,3	2,3,4	4,5,6	5,6,7	3,5,7	1,3,5
mark symbol	P_1	P_2	P_3	P_4	P_5	---	---	---
resource	Workers					---	---	---
Processing time for a single part	1,3,5	2,4,6	1,2,3	3,4,5	2,3,4	---	---	---
mark symbol	M_1	M_2	M_3	M_4	M_5	---	---	---
resource	Moulds					---	---	---
Processing time for a single part	---					---	---	---

TABLE V. AVAILABLE RESOURCES FOR OPERATIONS

Operation	O_1	O_2	O_3	O_4	O_5	O_6
Moulds	M_1, M_2, M_3	M_4, M_5				
Machines	E_1, E_2	E_3, E_4	E_5, E_6	E_7, E_8		
Workers					P_1, P_2, P_3	P_4, P_5

Using the method introduced above to solve the scheduling problem while the crossover rate is 0.75, the mutation rate is 0.1, and the population size is 50 with 500 generations. Results are shown in Fig.4, Fig.5 and Fig.6.

We use the most likely values of completion times and start times of operations to draw gannt charts on machines, moulds and workers in Fig.4, Fig.5 and Fig.6 respectively.

The first number in brackets on level lines in those three figures represents the job number, while the second represents the operation of the job that the first number represents. Numbers outside brackets on level lines represent batch sizes of the operation. For example, (13)66 3484 3748 on machine 1 illustrated in Fig.4 means that a batch of the third operation of job 1 with batch size 66 starts at time 3484 and ends at time 3748. The number 13 in brackets appears two times in Fig.4, and this means when processing the third operation of job 1, the original batch is split into two smaller batches where one's batch size is 66, and the other's is 84.

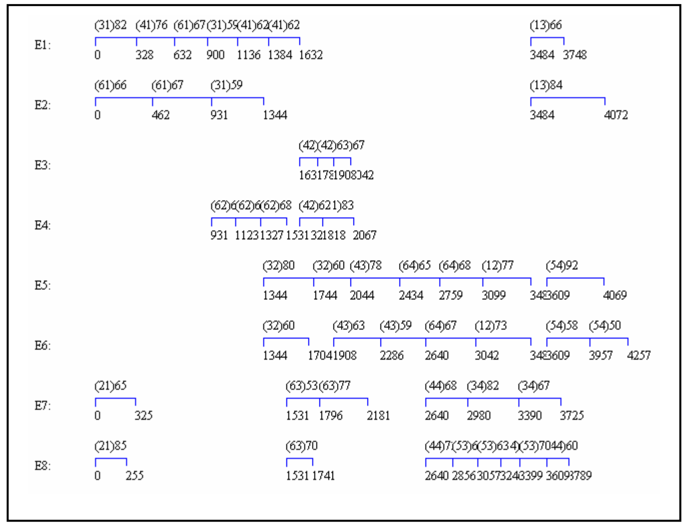

Figure 4. Gannt chart on machines

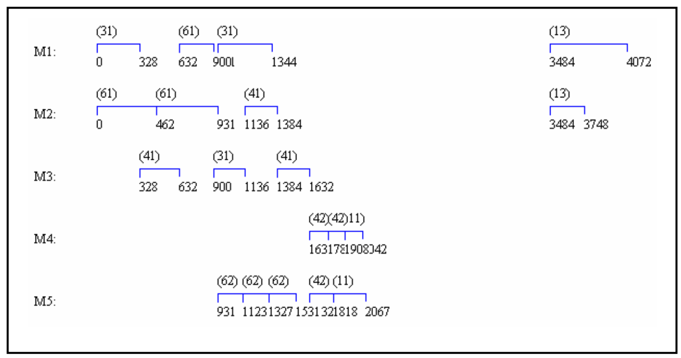

Figure 5. Gannt chart on moulds

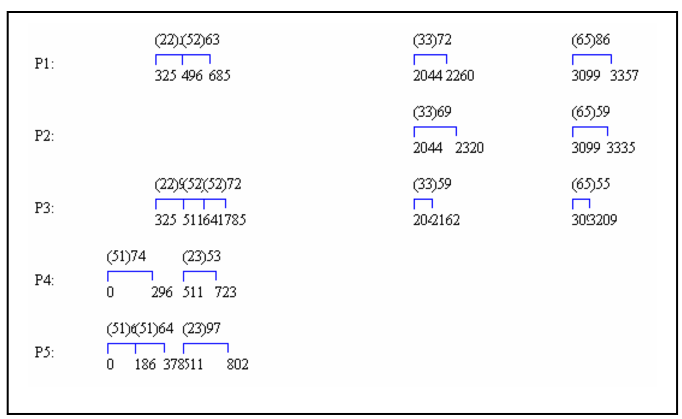

Figure 6. Gannt chart on workers

The emluator runs eight times and results are listed in table Ⅵ. The fitness value in the table below is obtained by using the most likely completion time with the time unit of the hour converted from the original time unit of the second.

TABLE VI. RESULTS OF THE EXPERIMENTS

Times	1	2	3	4	5	6	7	8
Fitness value	0.86	0.78	0.81	0.93	0.80	0.72	0.90	0.85

It is obvious that the GA proposed in this paper is capable of solving the batch splitting job shop scheduling problem with bounded batch sizes under multiple-resource constraints with optimal solutions.

VI. CONCLUSION

This paper addressed the batch splitting scheduling problem in manufacturing systems, constrained by machines, moulds and workers, with bounded batch sizes. A genetic algorithm with a new chromosome representation taking into account the batch splitting of the original batches of jobs, a new crossover method and a new mutation method based on the representation was proposed to solve the formulated problem. When scheduling, the original batch was split into many smaller batches according to chromosomes, which may have different batch sizes, and every smaller batch was regarded as a single part. An example of scheduling was given, and the results indicated that the method was feasible and efficient. The method deals with job shop scheduling problem in this paper, however we can also extend its application in scheduling problems of other types.

ACKNOWLEDGMENT

This work is supported by the National High-Tech Research and Development Program of China (No.2007AA04Z155), and the National Science Foundation of China (60573123).

REFERENCES

[1] Wang Wan-Liang, Wu Qi-Di, Xu Xin-Li. Hopfield neural network approach job-shop scheduling problems[J]. Acta Automatica sinica, 2002, 28(5):838-844

[2] Candido M A B, Khator S k, Barcia M A B. A genetic algorithm based procedure for more realistic job shop[J]. International Journal of Production Research, 1998, 36 (12): 3437-3457

[3] Wang Wan-Liang., Wu Qi-Di. Principles and Applications of Itelligent Agorithms for Production Scheduling[M]. Bei Jing: Science Press, 2007: 274-279

[4] Eimaraghy H, Patel V, Abdallah I B. Scheduling of Manufacturing Systems under Dual- resource Constrains Using Genetic Algorithms[J]. Journal of Manufacturing Systems, 2000,19(3): 186-203

[5] Christoph S. Thomalla*, Job shop scheduling with alternative process plans[J]. International Journal of Production Economics, 2001, 74:125~134

[6] Satyaki Ghosh Dastidar, Rakesh Nagi*. Batch splitting in an assembly scheduling environment[J]. International Journal of Production Economics, 2007, 105:372~384

[7] Li Fu-Ming, Zhu Yun-Long, Yin Chao-Wan, Song Xiao-Yu. Research on fuzzy job shop scheduling with alternative machines[J]. Computer Integrated Manufacturing Systems, 2006, 12(2): 169-173

[8] Jeong H, Park J, Leachman R C. A batch splitting method for a job shop scheduling problem in an MRP environment[J]. International Journal of Production Research, 1999, 37(15): 3583~3598

[9] Pan Quanke, Zhu Jianying. Optimization Method for A Job-Shop Scheduling Problem With Alternative Machines in The Batch Process[J]. Chinese Journal of mechanical Engineering, 2004,40(4): 36-39

[10] Gur Mosheiov, Daniel Oron. A single machine batch scheduling problem with bounded batch size[J]. European Journal of Operational Research, 2008, 187(3): 1069-1079

[11] Mitsuru Kuroda, Zeng Wang. Fuzzy job shop scheduling[J]. International Journal of Production Economics, 1996, 44(1-2):45-51

[12] Masatoshi Sakawa *, Ryo Kubota， Fuzzy programming for multi-objective job shop scheduling with fuzzy processing time and fuzzy due date through genetic algorithms[J]. European Journal of Operational Research, 2000, 120:393~400

[13] Parker, Paul Bradley. Genetic Algorithms and their use in geophysical problems[D]. Berkeley: University of California, 1999

[14] Wang Wan-Liang. Artificial Intelligence Principles and Applications[M]. Bei Jing: Higher Education Press, 2006: 263-264