An Effective Method to Solve Flexible Job-shop Scheduling Based on Cloud Model

Xiaobing Liu and Xuan Jiao
School of Management, Dalian University of Technology, Dalian 116024,China
jxis9ms@163.com

Tao Ning and Ming Huang
Institute of Software, Dalian Jiaotong University, Dalian 116045,China
daliannt@126.com

Abstract—**In order to solve the problem of flexible job-shop scheduling, this paper proposed a novel quantum genetic algorithm based on cloud model. Firstly, a simulation model was established aiming at minimizing the completion time, the penalty and the total cost. Secondly, the method of double chains structure coding including machine allocation chain and process chain was proposed. The crossover operator and mutation operator were obtained by the cloud model X condition generator because of its randomness and stable tendency. The non dominated sorting strategy was introduced to obtain more optimal solution. Finally, the novel method was applied to the Kacem example and a mechanical mould scheduling, the simulation results demonstrated that the proposed method can reduce the precocious probability and obtain more non dominated solutions comparing with the existing algorithms.**

Index Terms—**Flexible job-shop scheduling, Cloud model, Quantum genetic algorithm, Double chains structure coding**

I. INTRODUCTION

Flexible Job-shop Scheduling Problem (FJSS) is the extension of Job-shop Scheduling Problem (JSS). The domestic and foreign scholars have studied FJSS with various methods and achieved corresponding results[1].Bucker P.and Schlie R. [2] proposed FJSS in 1990, then the research hotspot about FJSS focused on the application of genetic algorithm and other intelligent algorithms. Chen H.[3] used genetic algorithm (GA) to solve FJSS aiming at minimizing the completion time, and simulated the chromosome with graph theory whose coding constituted with the routing and the process. Ho N.B.[4] proposed an optimization algorithm of three layer structure to solve FJSS. Najid N.M.[5] used simulated annealing algorithm(SAA) integrating neighbor function to minimize the maximum completion time of FJSS. Kacem I.[6] solved single objective and multi-objective FJSS respectively. He solved the machine allocation problem with the local search method and constructed the initial population firstly, and then improved the quality of solution with the optimization. D.Y.Sha[7] solved FJSS through updating the speed of particle swarm optimization(PSO) and combined tuba search algorithm(TS). B.Liu[8] used PSO based on genetic algorithm for permutation flow shop scheduling on the basis of the combination of PSO operator and local search operator. K.Fan[9] designed a novel algorithm to improve the binary PSO and obtained the approximate optimal solution. XIA W.J.[10] used the integration of PSO and SAA to solve FJSS. He solved the machine allocation with PSO and process scheduling with SAA. YU X.Y.[11] proposed multi workshop planning and scheduling based on the parallel cooperative evolutionary genetic algorithm. LIU A.J.[12] proposed multi-objective FJSS algorithm based on a multi population genetic algorithm by introducing fuzzy number to describe the completion time and delivery. ZHANG J[13] proposed the particle position update algorithm directly in the discrete domain on the basis of sequence and machine allocation. SHI J.F.[14] used continuous space ant colony algorithm to optimize the multi constraints of FJSS through establishing the simulation model of flexible routing.

There will be various shortcomings when the above methods are used such as low search efficiency, the weak ability of local search and premature convergence because of the loss of population diversity in later period. Considering the randomness and stable tendency of cloud droplets in the cloud model may improve the crossover operator and mutation operator of the adaptive genetic algorithm, this paper proposed a novel quantum genetic algorithm based on the cloud model to improve the convergence and robustness. The coding method of double chains was used on the basis of initializing the machine distribution chain with quasi level uniform design and heuristic initializing the process chain. The crossover operator and mutation operator were generated by the cloud model X condition generator, and the new population was obtained through rotation angle of quantum gates. The non dominated sorting strategy was introduced based on the fuzzy set theory. Finally, the proposed method is verified to be effective through the application to Kacem instances and the comparison with the existing algorithms.

doi:10.4304/jsw.9.11.2948-2954

II. Model Of Multi-Objective FJSS

A. Problem Description

FJSS is described as follows: there are N workpieces to be processed and M machines in workshop, each workpiece i($i\in \{1,2,\ldots,N\}$) includes n_i($n_i \geq 1$)processes, and the process should be processed with the specified route. R_{ij} means the j^{th} ($j\in \{1,2,\ldots, n_i\}$) process of workpiece i, M_{ij} ($M_{ij} \subseteq \{1,2,\ldots, M\}$) means the machine set, each R_{ij} may be processed by any machine m ($m\in \{1, 2,\ldots,M_{ij}\}$) with processing capacity, and m can process different workpieces [13]. The performance of different machines m makes the completion time different for R_{ij}.

B. Objective Function

The objective of FJSS is to select the suitable machine for each process and determine the optimum processing sequence, the objective function is established as follows:

1) To minimize the maximum completion time:

$$f1 = \min(F) = \min[\max(\sum_{m=1}^{M} F_m)] \quad (1)$$

$$F_m = \sum_{i=1}^{N}\sum_{j=1}^{n_i}(S_{ijm}b_{ijm} + S_{ijm}t_{ijm}) \quad (2)$$

In formula(1), F means the total completion time of all the machines, which acts as an important index to measure the machine load. In formula(2),F_m means the total completion time of machine m, b_{ijm} means the start time of R_{ij} in m, t_{ijm} means the processing time of R_{ij} in m, S_{ijm} takes the value of either 1(processed in machine m) or 0(not).

2) To minimize total cost:

$$f2 = \min(C) = \min[\sum_{i=1}^{N}(M_i + \sum_{j=1}^{n_i}\sum_{m=1}^{M} C_{ijm}S_{ijm})] \quad (3)$$

$$C_{ijm} = (\mu_{ijm} + \nu_{ijm}) \quad (4)$$

In formula(3), C means the total costs of workpiece i, M_i means the commodity cost of workpiece i, C_{ijm} means the processing cost of R_{ij} in m. In formula(4), μ_{ijm} and ν_{ijm} mean the labor cost and machine cost of R_{ij} in m respectively.

3) To minimize penalty:

$$f3 = \min(P) = \min\{\sum_{i=1}^{N}[pe_i \max((d_i - t_i),0) + pl_i \max((t_i - d_i),0)]\} \quad (5)$$

In formula(5), pe_i and pl_i mean the earliness penalty and tardiness penalty respectively, t_i and d_i mean the completion time and delivery for workpiece i.

4) To maximize the satisfaction:

$$f4 = \max[\frac{1}{N}\sum_{i=1}^{N} gI_i(\tilde{t}_i)] \quad (6)$$

In formula(6),$gI_i()$ means the satisfaction function for the customer to the completion time of workpiece i. Satisfaction is one of the important indexes to evaluate the fuzzy scheduling, its value depends on the fuzzy completion time $\tilde{t}_i$, $gI_i(\tilde{t}_i)=(\tilde{t}_i \wedge \tilde{D}_i)/\tilde{t}_i$, $\tilde{D}_i$ means the fuzzy delivery of workpiece i. In actual production, it is uncertain for the processing time and completion moment, and these factors will change in a certain interval to delivery.

C. Constraint Conditions

1) Process constraint

The sequence constraint of different processes in the same workpiece, where $S_{ijm}= S_{i(j-1)m}=1$:

$$\sum_{m=1}^{M} b_{ijm}S_{ijm} \geq \sum_{m=1}^{M}[(b_{i(j-1)m}t_{i(j-1)m})]S_{i(j-1)m} \quad (7)$$

2) Machine constraint

The same machine can only do one process at the same time, that is to say, if $\exists S_{ijm} = 1$ at the moment of t, then there mustn't be S_{xym}=1.

3) Continuity constraint

R_{ij} can't be interrupted in the processing, in formula(8), c_{ijm} means the completion time of R_{ij}.

$$c_{ijm} = \begin{cases} \max\{c_{i(j-1)m}, b_{ijm}\} + t_{ijm}, & j > 1; \\ b_{ijm} + t_{ijm}, & j = 1. \end{cases} \quad (8)$$

III. Improved Algorithm Based On Cloud Model

A. Concept of Cloud Model

The concept of cloud model was proposed by professor Li Deyi[15] on the basis of probability theory and fuzzy mathematics in 1995.It is a conversion model between qualitative and quantitative concept through a specific algorithm, and reveals the inherent relationship between randomness and fuzziness. During less than 20 years since the cloud model was proposed, it has been applied to many fields such as intelligent control, data mining and decision analysis successfully.

The definition of the cloud and the cloud droplet[15]: Assuming that T is a fuzzy subset on domain U, the mapping $C_T(x)$ *which is from* U *to* [0,1] is a random number with stable tendency, that is $\forall$ $x \in U$, $x \rightarrow C_T(x)$, and the distribution of $C_T(x)$ in U is called the membership cloud of T, or the cloud model of T, and then each variable x is called a cloud droplet. $\forall$ $x\in U$, $C_T(x)$ is not a clear membership curve but consists of a large number of cloud droplets. When the mapping $C_T(x)$ follows normal distribution, it is called the cloud model of normal distribution.

The cloud model describes some qualitative concept with three digital features including expected value E_x, entropy E_n and hyper entropy H_e. E_x is the expectation of distribution for the cloud droplet in domain. E_n is the uncertainty measure of the qualitative concept, H_e which can be also call entropy's entropy is the uncertainty measure of E_n.

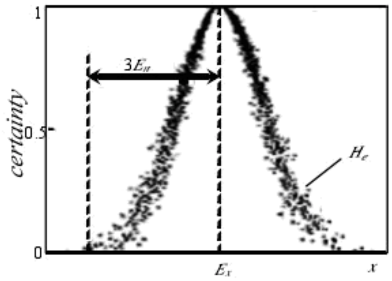

Figure 1. Digital feature of normal cloud model

The algorithm of normal cloud generator[16] is as follows:

Step 1: Generate a normal random number E_n' taking E_n as expectation, H_e as standard deviation according to the three digital features (E_x,E_n,H_e);

Step 2: Generate a normal random number x taking E_x as expectation, $|E_n|$ as standard deviation, x is a cloud droplet in domain U, $Drop(x_i,u_i)$;

Step 3: Calculate the certainty $\mu = e^{-\frac{(x-E_x)^2}{2(E_n')^2}}$ according to Step 1 and Step 2;

Step 4: Repeat Step 1 to 3 until generate N cloud droplets.

B. Improved Algorithm

1) Double chains quantum coding

This paper introduced a novel compensation factor $\gamma(\gamma \geq 1)$ based on probability coding. Assuming that p_i means a quantum chromosome, the encoding scheme of the i^{th} chromosome is as follows:

$$p_i = \begin{bmatrix} \begin{vmatrix} \alpha_{i1} \\ \beta_{i1} \end{vmatrix} \begin{vmatrix} \alpha_{i2} \\ \beta_{i2} \end{vmatrix} \cdots \begin{vmatrix} \alpha_{im} \\ \beta_{im} \end{vmatrix} \end{bmatrix} = \begin{bmatrix} \begin{vmatrix} \cos(\gamma t_{i1}) \\ \sin(\gamma t_{i1}) \end{vmatrix} \begin{vmatrix} \cos(\gamma t_{i2}) \\ \sin(\gamma t_{i2}) \end{vmatrix} \cdots \begin{vmatrix} \cos(\gamma t_{im}) \\ \sin(\gamma t_{im}) \end{vmatrix} \end{bmatrix} \quad (9)$$

Where $t_{ij} = 2\pi \times rad$,rad means a random number between (0,1), $i = 1, 2, \cdots, n$; j=1,2,…,m; n means the population size, and m means the number of qubits. γ extends the cycle from 2π to a multi-cycle, which can improve the convergence probability of the algorithm. Each chromosome consists of two parallel gene chains, which means the machine allocation chain and process chain of FJSS respectively. If each gene chain means an optimal solution, then each chromosome has two optimal solutions in the search space, that is:

$$p_{i\cos} = (\cos(t_{i1}), \cos(t_{i2}), \cdots, \cos(t_{in}))$$

$$p_{i\sin} = (\sin(t_{i1}), \sin(t_{i2}), \cdots, \sin(t_{in}))$$

$p_{i\cos}$ and $p_{i\sin}$ are called the cosine and sine solution. The two solutions can be updated synchronously in each chromosome iteration, which can expand the search space and increase the number of global optimal when the population size is the same.

2) Non dominated sorting

It is difficult to obtain the optimal solution which will meet all the objectives for FJSS. A sorting method of non dominated is proposed based on the fuzzy theory, and the method realizes the classification depending on the parameters N_p and n_p of individual p in population S, the specific steps are as follows:

Step 1: Initialize the parameter set N_p which includes all the individuals dominated by p and make $Np=\varnothing$;

Step 2: Initialize the variable n_p. n_p means the number of individuals which can dominate p;

Step 3: Calculate dominance relationship, $p,q\in$ S, if p can dominate q, then $N_p=N_p \cup \{q\}$, else if q can dominate p, then $n_p=n_p+1$; if n_p=0, then p is a non dominated individual, denoted as p_r=1,and p joins R_1, that is to say $R_1=R_1 \cup \{p\}$;

Step 4: Q means the set of residual individuals, making i=1,when $R_i \neq \varnothing$,Q$=\varnothing$. If $q\in N_p$, then $n_q= n_q-1$, else if $n_q=0$, then $q_r=i+1$; make $Q = Q \cup \{q\}$,$i=i+1$,$R_i=Q$;

Step 5: Judge whether R_i is empty or not, if it is empty, then stop, otherwise turn to Step 4.

In order to maintain the diversity of the population, the selection is made according to the crowding distance of the chromosomes based on non dominated sorting.

3) Cloud quantum genetic algorithm

Cloud quantum genetic algorithm(CQGA) is a novel optimization method of GA combining of cloud model theory and quantum theory. The specific steps are as follows:

Step 1: Initialize the population and execute the double chains quantum coding to the chromosome;

Step 2: Design the fitness function as $fit(x)= 1/Z(x)$, $Z(x)$ is the individual objective function value, the smaller the function value is, more excellent the individual is;

Step 3: Select the excellent individuals from the population into the next generation using the best one preservation strategy and fitness proportional;

Step 4: Generate the crossover operator p_{cr} using the cloud model X condition generator. Put the gene region between inter-section in the first of the sub generation and remove the same code in the father generation, then copy the rest code to the sub generation according to the order. If the individual in the sub generation is beyond the constraints, adjust the position of 0;

Step 5: Generate the mutation operator p_{mt} using the cloud model X condition generator. Select two codes r_1 and r_2 from the gene coding of mutational individuals randomly and exchange the selected codes to generate the new one;

Step 6: Construct new population which consists of *m* excellent individuals from the father generation and *m* individuals in the sub generation, and then to extend the size of the population and the search space;

Step 7: Get the next *m* generation population through the operation of GA;

Step 8: Update the quantum gates according to Schrodinger equation;

Step 9: Judge whether the stopping condition is met, if not, turn to Step 3, else stop running the algorithm.

C. Analysis of Convergence

CQGA is a novel hybrid method which consists of the diversity of the quantum population and generating new individual through rotation angle of the quantum gates. It won't influent the convergence of the algorithm because of using the cloud model X condition generator. The state transformation of CQGA is as follows:

Assuming the length of the chromosome is *L*, the population size is *P*, the size of the state space about GA is 2^{LP}, and the size of the state space about CQGA is u^{LP}(u is the dimension of the state space), then the state transfer process of the population described by Markov chain is as follows:

$$Q_t \xrightarrow{observe} P_k (\xrightarrow{cross} p_t' \xrightarrow{mutate} p_t'') \xrightarrow{keep\ the\ optimal\ solution\ and\ update\ Q_k} Q_{t+1}$$

The upgrading operation of CQGA is influenced not only by the evolutionary constraints of GA but also by that of quantum rotation gates. Its convergence is not affected after the transformation.

IV. Analysis And Verification

A. Analysis of Simulation Experiment

In order to verify the performance of the proposed method, this paper takes the minimization of completion time, minimization of penalty and minimization of cost as targets to test. The testing data based on the classic Kacem [17] example is as follows: the size of the population is 200, the maximum number of iterations is 100, the crossover probability is 0.45 and the mutation probability is 0.02.

Five standard Kacem examples are solved by CQGA and they were compared with the existing AL+CGA[17], PSO+TS[18] and HBCA [19] at the same time. The results are shown in Table Ⅰ. *n* is the number of the workpieces, *m* is the number of the machines, Sol_n(n=1,2,3,4) are the solutions obtained by different algorithms, T_x is the maximum completion time of the machine, M_t is the total load to the machines, M_x is the maximum load in balancing the load of the machines. It can be seen from Table 1 that the proposed method can obtain more non dominated solutions and has got the current optimal solution. For example, for case 10×7 although both HBCA and CQGA obtain three non dominated solutions, the solution (12,61,11) obtained by HBCA is dominated by (11,60,11),(12,61,10) and (12,60,11) obtained by CQGA, and (11,61,11) obtained by HBCA is dominated by (11,60,11) obtained by CQGA, and (12,60,12) obtained by HBCA is dominated by (11,60,11) and (12,60,11) obtained by CQGA.

TABLE I.
COMPARISON WITH DIFFERENT ALGORITHMS ON KACEM

n×m	Obj	AL+CGA		PSO+TS		HBCA			CQGA			
		Sol$_1$	Sol$_2$	Sol$_1$	Sol$_2$	Sol$_1$	Sol$_2$	Sol$_3$	Sol$_1$	Sol$_2$	Sol$_3$	Sol$_4$
4×5	*Tx*	16		11		11	12	13	11	11	12	11
	Mt	34		32		31	31	33	30	30	31	31
	Mx	10		10		10	8	7	9	8	7	8
8×8	*Tx*	15	16	15	15	14	15	16	14	14	15	14
	Mt	79	75	77	75	76	75	73	73	74	73	73
	Mx	13	13	12	12	12	12	13	12	12	12	11
10×7	*Tx*					12	11	12	11	12	12	
	Mt					61	61	60	60	61	60	
	Mx					11	11	12	11	10	11	
10×10	*Tx*	7		7		8	6	7	7	6	6	6
	Mt	45		43		41	42	41	41	40	41	40
	Mx	5		6		6	5	5	4	5	5	6
15×10	*Tx*	23		11		11	11		10	11		
	Mt	95		93		90	91		90	89		
	Mx	11		11		11	11		10	11		

B. Analysis of Scheduling Experiment

Further more tests were conducted in the mould workshop of a machinery company to verify the performance of the novel method for multi-objective FJSS. The data in Table Ⅱ (6workpieces×8machines) are the results after the raw data of the mould have been processed.

These data consist of machines, processing time and costs, in which the unit of time is minute and the unit of cost is yuan. The set of machines are as follows: rough turning lathe (M1), fine turning lathe(M2),rough milling machine(M3), fine milling machine(M4), boring machine(M5), planer(M6), grinder(M7) and machining center (M8).

TABLE II
.DATA OF 6×8 EXAMPLE

workpiece	process 1	process 2	process 3	process 4	process 5	process 6
1	M6,48,75.0	M2,45,69.8	M2,43,68.0	M4,48,62.0	M5,22,20.8	M8,32,27.0
2	M6,47,74.6	M5,25,24.5	M6,46,68.6	M3,28,34.8	M2,40,66.0	M8,30,25.0
3	M1,12,9.8	M4,46,57.0	M4,46,54.8	M7,24,40.9	M6,48,76.8	M4,48,63.5
4	M3,28,35.0	M7,24,40.2	M8,31,25.5	M6,49,77.8	M1,13,10.2	M3,28,34.6
5	M3,30,37.2	M8,32,26.2	M6,46,69.9	M7,26,48.2	M2,38,60.4	M4,44,55.8
6	M5,22,21.6	M3,32,38.0	M5,26,26.8	M1,14,10.1	M1,12,9.6	M4,45,57.6

The proposed CQGA is compared with several existing algorithms after being conducted 50 times. The result is shown in Table 3. It can be seen that the optimal solution and the average solution with CQGA are both better than those of other algorithms, besides, it can get less penalty than the other two.

TABLE III
.COMPARISON OF THREE ALGORITHMS

Objective	PSO[20]		IPSO[21]		CQGA	
	Opt Solution	Avg solution	Opt Solution	Avg solution	Opt Solution	Avg solution
Process time	1198	1246	1105	1148	998	1065
Process cost	1380	1428	1296	1378	1009	1135
Penalty	231	246	216	230	170	188

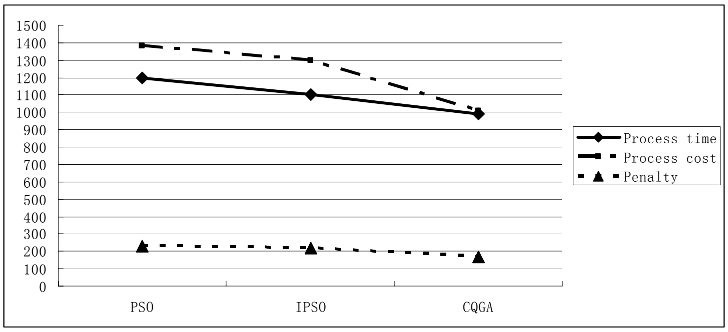

Figure 2. Comparison of optimal solutions

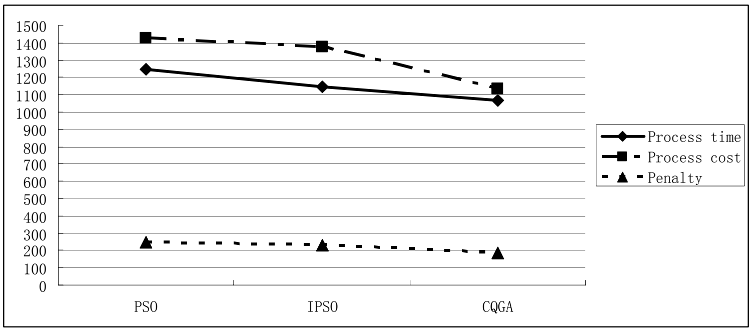

Figure 3. Comparison of average solutions

V. CONCLUSION

The mathematics model of multi-objective FJSS was established in this paper, and the coding method of double chains was used including the initialization of the machine distribution chain with quasi level uniform design and the heuristic initialization of the process chain. On the basis of the theory of cloud and quantum, the crossover operator and mutation operator were generated by the cloud model X condition generator, and the new population was obtained through rotation angle of quantum gates. The non dominated sorting strategy was introduced based on the fuzzy set theory. Finally, the proposed method was applied to Kacem instances and the scheduling of a mould workshop, and then compared the data with that of the existing algorithms. The comparison of the results verified the proposed method could not only reduce the maximum completion time and process cost but decrease the penalty effectively.

ACKNOWLEDGMENTS

This work is partially supported by the National Natural Science Foundation, China (No.61034003), the National Key Technology R&D Program, China (No.2012BAF12B08), the National 863 Plan Project, China (No.2012AA 041402-4), the Talented Young Scholars Growth Plan of Liaoning Province Education Department, China (No.LJQ2013048). The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

CONFLICT OF INTERESTS

The author declares no conflict of interests.

REFERENCES

[1] LIANG Xu,LIU Pengfei, HUANG Ming, “Genetic algorithm for multi-order Job Shop scheduling under mixed production patterns”, Computer Integrated Manufacturing Systems, vol.18,no.10, pp.2217-2223,2012.

[2] Bucker P. and Schlie R, “Job-shop scheduling with multi-purpose machines”, Computing, vol.45, no. 4,pp.369-375,1990.

[3] Chen H.,Ihlow J. and Lehmann C, “A genetic algorithm for flexible job-shop scheduling”, In: Proceedings of the 1999 IEEE International Conference on Robotics &Automation, Detroit Michigan:IEEE,vol.2,pp.1120-1125,1999.

[4] Ho N.B. and Tay J.C, “Genance:an efficient culture algorithm solving the flexible job shop problem”, In:Proceedings of Congress on Evolutionary Computation,pp.1759-1766,2004.

[5] Najid N.M.,Dauzere-Peres S.and Zaidat A, “A modified simulated annealing method for flexible job shop scheduling problem”, In:Proceedings of the IEEE International Conference on Systems Man and Cybernetics.NJ,USA,IEEE,pp.89-94,2002.

[6] Kacem I. “Genetic algorithm for the flexible job shop scheduling problem”, IEEE International Conference on Systems, Man and Cybernetics, vol.4,pp.3464-3469,2003.

[7] D.Y.Sha, C.Y.Hsu, “A hybrid particle swarm optimization for job-shop scheduling problem”, Computers and Industrial Engineering, pp.791-808, 2006.

[8] B.Liu,L.Wang,Y.H.Jin, “An Effective PSO-Based Memetic Algorithm for Flow Shop Scheduling”, IEEE Transactions on Systems, Man and Cybernetics-Part B: Cybernetics,

vol.37, no.1, pp.18-27,2007.
[9] Kun Fan,Ren-qian Zhang, Guoping Xia, "An Improved Particle Swarm Optimization Algorithm and Its Application to a Class of JSS Problem", Proceedings of IEEE International Conference on Grey Systems and Intelligent Services, Nanjing, 2007.
[10] Xia W J,Wu Z M,"An effective hybrid optimization approach for multi-objective flexible job shop scheduling problems", Computers & Industrial Engineering,vol.48,no.2, pp.409-425, 2005.
[11] YU Xiaoyi,SUN Shudong, CHU Wei, "Parallel collaborative evolutionary genetic algorithm for multi-workshop planning and scheduling problems", Computer Integrated Manufacturing Systems, vol.14,no.5,pp.991-1000,2008.
[12] LIU Aijun,YANG Yu,XING Qingsong,et al, "Multi-population genetic algorithm in multi- objective fuzzy and flexible Job Shop scheduling", Computer Integrated Manufacturing Systems, vol.17,no. 9,pp.1954-1961,2011.
[13] ZHANG Jing,WANG Wanliang,XU Xinli, et al, "Improved particle swarm algorithm for bath splitting flexible job shop scheduling", Control and Decision,vol.27,no.4,pp.35-40,2012.
[14] SHI Jinfa, JIAO Hejun, CHEN Tao, "Multi- objective Pareto Optimization on Flexible Job-Shop Scheduling Problem about Due Punishment", Journal of Mechanical Engineering, vol.48, no.12,pp. 188-196,2012.
[15] Li D.Y,Du Y, "Artificial Intelligence with Uncertainty", Beijing: National Defense Industry Press,2005.
[16] Dai C.H,Zhu Y.F.,Chen W.R, "Cloud model based genetic algorithm and its applications", acta electronic sinicaA,vol.35,no.7, pp.1419- 1424, 2007.
[17] KACEM I,Hammani S Borne P, "Approach by localization and multi-objective evolutionary optimization for flexible job-shop scheduling problems", IEEE Trans Syst Man Cyb C,vol.32, no.1, pp.1-13,2002.
[18] Partha Pratim Das, Sriyankar Acharyya, Hybrid Local Search Methods in Solving Resource Constrained Project Scheduling Problem, Journal of Computers,vol.8, no.5, pp.1157- 1166, 2013.
[19] Tran Quang Tuan,Phan Xuan Minh. Adaptive Fuzzy Model Predictive Control for non-minimum phase and uncertain dynamical nonlinear systems, Journal of Computers, vol.7, no.4, pp. 1014-1024,2012.
[20] Fadi A. Aloul, Syed Z. H. Zahidi, et al. Solving the Employee Timetabling Problem Using Advanced SAT & ILP Techniques. Journal of Computers, vol.8,no.4,pp.851-858, 2013.
[21] YANG Hongan,SUN Qifeng, SUN Shudong, et al, "Job Shop earliness/tardiness scheduling problem based on genetic algorithm", Transactions of the Chinese Society for Agricultural Machinery, vol.17,no.8,pp.1798-1805,2011.

Xiaobing Liu was born in Changchun city, Jilin Province of the P. R. China in 1956. He received his M.S.degree from Dalian University of Technology in 1984, and his Ph.D.degree from the Germany Dortmund University in 1992. He is now working in the Dalian University of Technology. His research is mainly in the area of artificial intelligence and computation integrated manufacturing system . He has involved with 8 projects supported by the national Natural Science Foundation of China, and 2 projects supported be the national High Technique Developing Program. He has published over 150 research papers.

Xuan Jiao was born in Baishan City, Jilin Province of China in1986. She received the M.S.degree from Liaoning University, China in 2012. She is pursuing Ph.D.degree from school of management of Dalian University of Technology under the supervision of Prof. Xiaobing Liu. Her current research interests are in the computer integrated manufacturing system.

Tao Ning was born in Penglai City, Shandong Province of China in 1979. He received his M.S.degree in Dalian Maritime University in 2006 and his Ph.D.degree in Dalian Maritime University in 2013. He majors in computer information management and computer integrated manufacture and is working as a professor in.Dalian Jiaotong University.

Ming Huang was born in Changchun City, Jilin Province of China in 1961. He received his B.S. degree in Jilin University, China in 1982. He is currently working as a professor in Dalian Jiaotong University, China. His research interest fields include computational geometry, large scale software development, design and analysis of algorithms.