Research on Robustness Prediction and Reactive Scheduling based on Aviation Supporting Resource Breakdown

Lianfei YU [1, 2, a)], Cheng ZHU [1,b)], Weiming ZHANG [1,c)]and Jin Liu [1,d)]

[1]Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology, Changsha 410073, China.

[2]PLA Urumuqi Nationalities Commander Academy, Urumuqi 830001, China.

a) ylf81242@163.com

b) Corresponding author: zhucheng @ nudt.edu.com

c) wmzhang @ nudt.edu.com

d) liujin22934@163.com

Keywords: robustness prediction, carrier-based aircrafts, reactive scheduling, evolution algorithm

Abstract. Due to the complicated environment and the supporting procedure of the carrier-based aircrafts, supporting resources are prone to fail when processing supporting operations. In order to reduce the impact of the failures and make the takeoff time of all carrier-based aircrafts minimum, a robust pre-scheduling model is established according to the predictable law of supporting resource failure. If a supporting operation is broken down by the unexpected supporting resource failure, then the affected operations will be rescheduled by shifting to right and replacing the supporting resource, which is named reactive scheduling. For the pre-scheduling and reactive scheduling, an improved differential evolution algorithm is proposed. Finally, the feasibility and validity of the model and algorithm are verified by experiments.

1 Introduction

When the combating is coming, the carrier-based aircrafts will take off wave by wave, a report [1] pointed out that aviation supporting resource scheduling is a key factor to affect the takeoff time of each wave..

The environment of aircraft carrier is complex, such as deck space is limited, the weather changes very often and so on, which may be make supporting resource failure, for example people injured, catapult failure, and so on, these failures may lead the scheduling scheme invalid.

In order to deal with supporting resource failures and shorten the takeoff time of each wave, it is reasonable to make the scheduling scheme with robustness and adaptability. Although there are many researches on robust scheduling in the field of production scheduling, such as job shop robust scheduling, there is no similar supporting procedure with carrier-based aircraft. So it is meaningful to schedule supporting resources for the supporting procedure of carrier-based aircraft including supporting resources failure.

2 Literature Review

There are some works on the analysis of aviation supporting procedure and the model of the supporting resource scheduling. Lin [2] analyzed the influencing factors of sortie rates and system work flow, and built the model of sortie rates generation based on Petri net theory. Wei established models of resource scheduling with two different dispatch modes: the supporting resource scheduling models of wavely dispatching aircraft[3] and continuously dispatching aircraft [4]. The two models are studied with the theory of flexible shop scheduling and rescheduling, respectively. Based on the system dynamics theory, Yue [5] established stock flow chart and mathematical model of dynamic aircraft transporting complex systems, verified two dynamic allocation procedure with hanging and no hanging when the aircraft turn around.

The above researches neglected the investigation on the aviation supporting procedure of the carrier-based aircraft, and supporting resource failures are less focused on. In order to overcome the above shortcomings, based on the "one-stop" supporting procedure of the carrier-based aircraft, according to the predictable supporting resource failures, a robust scheduling model is established. For the unexpected supporting resource failure, a reactive scheduling is proposed.

3 Robust Pre-Scheduling

3.1 Representation of aviation supporting resource failure

It is reasonable to assume that the supporting resource failure is subject to exponential distribution. When the probability of supporting resource failure is bigger than a threshold, the aviation supporting resource may fail, so a maintenance time should be inserted into the scheduling scheme. After the maintenance, the probability of the aviation supporting resource failure is reset to zero.

3.2 Robust pre-scheduling model

Robust pre-scheduling means that resource failures have been included when generating scheduling scheme, and take some measures, such as inserting a period of idle time into the scheduling, which will make the scheme can deal with most of the foreseen failures without adjusting. The assumptions of the robust pre-scheduling model are as follows:

(1) any type of carrier-based aircraft can be supported at time 0.

(2) the carrier-based aircraft with higher priority should take off earlier than the one with lower priority, which does not mean that each supporting operation of the carrier-based aircraft with higher priority should be finished earlier than the one with lower priority .

(3) there are some operations which can be supported at the same time in a carrier-based aircraft.

(4) Each supporting resource can only process one operation at a time.

(5) an operation only can be processed by a supporting resource each time.

(6) different supporting resources process the same operation may be with different time.

(7) the conversion time between the operations is negligible.

Parameter definition:

K : the number of carrier-based aircraft type, the index is k ;

N_k : the number of carrier-based aircraft type k ;

N : The total number of carrier-based aircrafts, the index is i ;

m_k : the supporting operations number of carrier-based aircraft type k , the index is j ;

M : The total number of aviation supporting resources, the index is m ;

M_{kj} : the supporting resource number of the j th operation of carrier-based aircraft type k ;

O_{ij} : operation j of carrier-based aircraft i ;

P_{kij} : Immediate precedent operation O_{ij} in the carrier-based aircraft type k ;

T_{kijm} : processing time if operation O_{ij} in the carrier-based aircraft type k is to be performed on resource m ;

S_{kij} : start time of operation O_{ij} in the carrier-based aircraft type k ;

C_{kij} : Completion time of operation O_{ij} in the carrier-based aircraft type k ;

L: A very large positive number.

$P_m(t)$: The probability of the failure of the m th supporting resource at time t ;

t_{mr} : the time of completing the r th maintenance for the m th supporting resource;

R_m : The time required to maintenance the m th supporting resource;

LT_m : The accumulated working time of the m th supporting resource between two failures;

FT_m : The accumulated free time of the m th supporting resource between two failures;

AVT_m : The average working time of the m th supporting resource between two failures;

TV_m : The probability threshold of the m th supporting resource failure.

3.3 Probability distribution of aviation supporting resource failure

Assume that the failure of supporting resource is subject to exponential distribution, which can be described as follows: $P_m(t)=1-e^{\frac{LT_m}{AVT_m}}$, $LT_m=t-FT_m-t_{mr}$, $t>0$, $r>0$. Where the formula calculates the probability of the $(r+1)th$ failure of the m th supporting resource at time t, FT_m is the accumulated free time of the m th supporting resource between the r th failure and the $(r+1)th$ failure.

3.4 Robustness pre-scheduling algorithm of aviation supporting resource failure

Based on the pre-scheduling algorithm, the DE algorithm is improved, when the chromosomes are initialized, the global selection and random selection are combined. When decoding, according to the processing time and the probability of failure, the maintenance time should be inserted and the algorithm is as follows:

(1) Take a chromosome;

(2) Take an operation O_{ij} and its corresponding supporting resource $m \in M_{kj}$ from the chromosome, the processing time of the resource is T_{kijm};

(3) The completing time is recorded as CT_1 in P_{kij}, if O_{ij} is the first operation of the carrier-based aircraft, then $CT_1 = 0$.;

(4) The completing time of immediate precedent operation O_{ij} in m is recorded as CT_2, if O_{ij} is the first operation in m, then $CT_2 = 0$, if the maintenance time is inserted before O_{ij}, then $CT_2 = CT_2 + R_m$;

(5) If $CT_1 \geq CT_2$, $S_{kij} = CT_1$, or else $S_{kij} = CT_2$;

(6) The accumulated working time of m is the bigger one between $LT_m = CT_1 + T_{kijm} - FT_m - t_{mr}$ and $LT_m = CT_2 + T_{kijm} - FT_m - t_{mr}$;

(7) Calculate $P_m(t)$, if $P_m(t) > \mathrm{TV}_m$, then insert R_m, reset $LT_m = 0$, $r = r+1$;

(8) Repeat (2) - (7), until all the operations have been taken;

(9) Repeat (1) - (8) until all chromosomes have been selected.

4 Reactive Scheduling

4.1 The robust index of reactive scheduling

The robust pre-scheduling scheme must be adjusted if the unexpected supporting resource failure occurs, named reactive scheduling. Assuming that $m \in M_{kj}$ breaks down when supporting the operation O_{ij} at time t, if $t + T_{kijm} + R_m \leq \min(\mathrm{S}_{ij+1}, \mathrm{S}_{ij,m})$, where $\mathrm{S}_{ij,m}$ is the immediate succeeding operation of O_{ij} in m, then O_{ij} waits until m is maintained(shifted to right), which will not affect other operations, or else many operations may be delayed, the target and scheme of scheduling will become bad. Thus, in order to reduce the influence, a reasonable robust index is needed to guide the reactive scheduling.

The robustness of scheduling is usually measured by scheme robustness SB and performance robustness OB.

At present, there are two types of scheme robustness index [6]: the time difference and the order difference of each task between the new scheduling scheme and the pre-scheduling scheme. Because the takeoff time of all aircrafts is concerned, the time difference will be chosen as the scheme robustness index, which can be calculated as follows:

$$SB = \frac{\sum_{i=1}^{N} \sum_{j=1}^{m_k} (C_{kijR} - C_{kijp})}{\sum_{k=1}^{K} \sum_{i=1}^{N} \sum_{j=1}^{m_k} O_{kij}} \tag{1}$$

Where, C_{kijR} is the completing time of operation in actual scheduling, C_{kijp} is the completing time of operation in pre-scheduling.

The performance robustness index also includes two types [7]: one is the absolute performance, the other is the object difference between the new scheduling and the pre-scheduling, including the absolute difference and the relative difference. Because the robustness index is mainly for reactive scheduling, the relative difference will be chosen as performance robustness index, the formula is as follows:

$$OB = \frac{R_{ms} - P_{ms}}{R_{ms}} \tag{2}$$

Where, R_{ms} is the object of actual scheduling, P_{ms} is the object of pre-scheduling.

Finally, the robustness index of reactive scheduling is $B = \min(a * OB + (1-\mathrm{a}) * \mathrm{SB})$, where $a \in [0,1]$ is the weight.

4.2 The method of reactive scheduling

There are three kinds of reactive scheduling methods: shifting to right rescheduling, partial rescheduling, and full rescheduling.

Because a number of maintenance time have been inserted into the pre-scheduling, so when the unexpected failure occurs, it is the simplest way to delay the affected operations until the resources are maintained, that is shifting to right. The partial rescheduling means some operations affected will be rescheduled, the full rescheduling means all operations affected will be rescheduled.

If there are some operations affected by the supporting resource failure, shifting to right may result in performance robustness and scheme robustness worse together, full rescheduling may lead to scheme robustness worse but with little impact on performance robustness; partial rescheduling has a balance and less impact on performance robustness and scheme robustness. So in order to make the reactive scheduling has some versatility, the reactive scheduling is the combination of shifting to right and partial rescheduling (replacing failing resource).

4.3 The reactive scheduling algorithm

The algorithm is as follows:

(1) According to the moment of the supporting resource failure, mark the affected operations directly;

(2) From back to front, one operation is taken from the affected operations directly, whose supporting resources will be marked;

(3) Select a supporting resource from the marked, if the selected one is failing, the selected operation and other indirectly affected operations will shift to right, and calculate the robustness index;

(4) If the selected supporting resource is not failing, the selected operation and other indirectly affected operations will shift to right, and calculate the robustness index;

(5) When the affected operation is processed by other marked supporting resource, the immediate succeeding operations in the same aircraft will shift to right, and calculate the robustness index;

(6) Comparing the robustness index when the operation is processed by different marked supporting resource, the operation should be processed by the supporting resource with the least robustness index;

(7) Repeat (2) - (6), until all the operations affected directly by the resource failure have been processed.

5 Improved Differential Evolution Algorithm

DE (Differential Evolution, DE) is an optimization algorithm jointly proposed by Storn and Price [8] for solving Chebyshev polynomials in 1996, which uses floating-point vector encoding for random search in continuous space. DE algorithm is an intelligent optimization method based on population, and seek the optimal solution of the problem by the greedy competition mechanism. The traditional DE is originally designed for continuous optimization problems where chromosomes are floating-point numbers which cannot map the discrete/combinatorial search space, so a new improved DE will be introduced.

5.1 Encoding

The supporting resource scheduling of carrier-based aircraft can be decomposed into two sub-problems: sequencing and routing. So the chromosome will contain two parts, part1: sequencing that determines operations order within each carrier-based aircraft and a sequence of all operations of aircrafts, part2: routing that assigns each operation to an appropriate supporting resource. The method of sequencing and routing has been applied in some works [9]. This paper adopts the method.

5.2 Population initialization

Population initialization is a crucial task in evolutionary algorithms because it can affect the convergence speed and the quality of the final solution. In this section, we mainly present one method to solve the second sub-problem through assigning each operation to a suitable supporting resource. The method takes into account both the processing time and the workload of supporting resource, named the global assignment, which is improved based on the previous work [10], the steps are as follows:

Step 1: Create a new array to record the time of each supporting resource has worked, at first each element is 0;

Step 2: Follow the order from front to back, take an operation from part1;

Step 3: Add the operation processing time of each available supporting resource to the corresponding location in the array, then find the shortest time and the corresponding supporting resource will be assigned to the operation, if the minimum is multiple, then randomly selects one for the operation;

Step 4: The order of the selected resource in the set will be added to the part2 corresponding gene bit;

Step 5: In the array, the total time of the selected resource has work is modified, the others do not change;

Step 6: Take the next operation from part1, and continue the Step 3 - 6 until all operations are taken.

5.3 Mutation

The mutation of DE is not suitable for the discrete problems, so it will be replaced by two methods: Precedence Operation Crossover (POX)[11] and Multiple Point Crossover (MPX)[12]. The POX inherits the operation sequence from parent chromosome to child chromosome, the MPX transits the supporting resource assignment from parent chromosome to child chromosome.

5.4 Crossover

The binomial crossover is used, at least one component of the experimental chromosome is from the mutated chromosome by random.

5.5 Selection

The greedy selection strategy is adopted to select the best individual according to the fitness $f(\cdot)$ of the target chromosome X_i^g and the test chromosome U_i^g. The equation is chosen as follows:

$$X_i^{g+1} = \begin{cases} U_i^g, & f(U_i^g) < f(X_i^g) \\ X_i^g, & otherwise \end{cases} \tag{3}$$

The improved DE algorithmic flow is depicted as follows:

Step 1: Set the population size NP, scale factor F, crossover probability CR, iterations G_{max};

Step 2: Initialize the population, when assigning supporting resources, the number of chromosomes is $0.8*NP$ with the global assignment, the others are random;

Step 3: Evaluate each chromosome;

Step 4: Mutation; Crossover; Selection;

Step 5: Check whether each chromosome in the next generation meets the requirements of takeoff order. If not, adjust the chromosome to meet the requirement;

Step 6: If the termination condition is satisfied, output the optimal solution. Otherwise, go to Step 3.

6 Experiment Design

The improved DE is implemented in Matlab 2013(a) on an 2.50GHz Intel Core i5-3210m CPU, 4GB of RAM, Windows 7.0.

There are 5 types of carrier-based aircraft will take off from the aircraft carrier in the first wave and the order of taking off is: a helicopter (numbered 1), a warning airplane(numbered 2), an escort fighter(numbered 3), an antisubmarine aircraft(numbered 4) and 12 fighters(numbered 5-12).

In order to make all the carrier-based aircrafts take off as soon as possible, a supporting resource scheduling needs to be planned according to the supporting procedure as shown in Fig.1, there are

12 operations (A-L, numbered 1-12) in the procedure of supporting for a carrier-based aircraft, but the helicopter and warning airplane without loading bombs, the helicopter does not need to catapult.

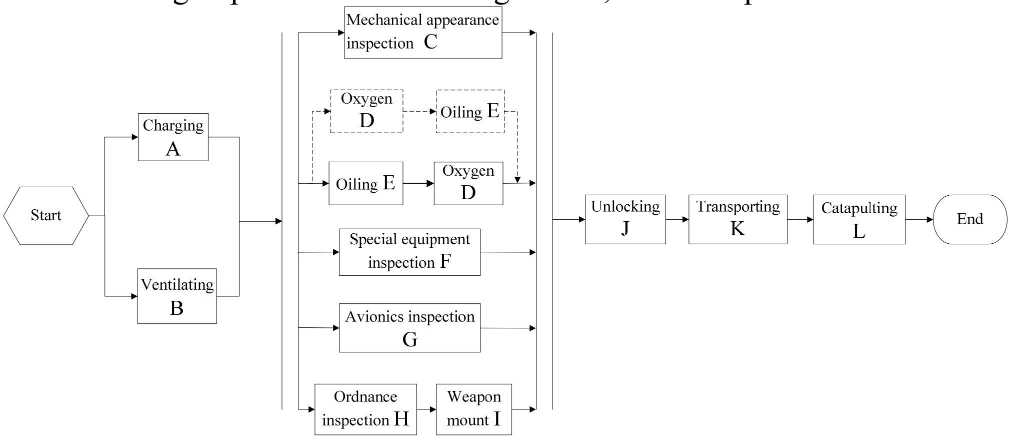

FIGURE 1 The supporting procedure of a carrier-based aircraft

There are 30 supporting resources, operations each supporting resource can process and time(in minute) are shown in Tab.1.

TABLE1. Operation and its processed time by supporting resources

Resource	(Operation, Time)	Resource	(Operation, Time)	Resource	(Operation, Time)
1	(A,2)	11	(E,10)	21	(I,14)
2	(A,3)	12	(E,11)	22	(I,12)
3	(B,2)	13	(F,7)	23	(J,4)
4	(B,1)	14	(F,8)	24	(J,3)
5	(C,3)	15	(G,4)	25	(K,6)
6	(C,2)	16	(G,2)	26	(K,7)
7	(D,3)	17	(H,3)	27	(L,2)
8	(D,4)	18	(H,2)	28	(L,3)
9	(E,11)	19	(I,13)	29	(L,4)
10	(E,12)	20	(I,15)	30	(L,3)

Each supporting resource may fail, for the 30 supporting resources, the threshold of supporting resource failure is TV =[0.7,0.7,0.7,0.7,0.8,0.8,0.7,0.7,0.6,0.6,0.6,0.6,0.7,0.7,0.7,0.7,0.7,0.7,0.6,0.6,0.6, 0.6,0.7,0.7,0.7, 0.7,0.7,0.7,0.7, 0.7]； the average working time between failures ATV =[10,10,10,10,10, 10,10,10,20,20,20,20, 10,10,10,10,10,10, 20,20,20,20,10,10,10,10,10,10,10,10] ; the maintenance time of the failure R =[4,4,4,4,4,4,4,4,8,8,8,8,4,4,4,4,4,4, 8,8,8,8,4,4,4,4,4,4,4,4]. The above parameter values are from the relevant public information. In the robustness index $B = \min(a * OB + (1-\mathrm{a}) * \mathrm{SB})$, the weight $a = 0.5$, which means that scheme robustness and performance robustness are important equally. The parameters of the improved DE algorithm are: population size $NP = 100$, scaling factor $F = 0.5$, crossover rate $CR = 0.9$, iterations $G_{\max} = 400$.

7 Experimental results and analysis

The idle time is inserted into the scheduling to generate robust pre-scheduling to deal with the failure, as shown in Fig.2. The vertical axis represents the number of the supporting group, the horizontal axis represents the total supporting time, the first or two digits of the digital representation is the number of carrier-based aircraft, the latter two represent the number of operation, for example, “507” indicates the operation 7(operation G) of the carrier-based aircraft 5, “1203” indicates the operation 3(operation C) of the carrier-based aircraft 12. It can be seen from the Gantt chart, the procedure of carrier-base aircraft and the order of take-off meet requires. The total supporting time is 103 minutes.

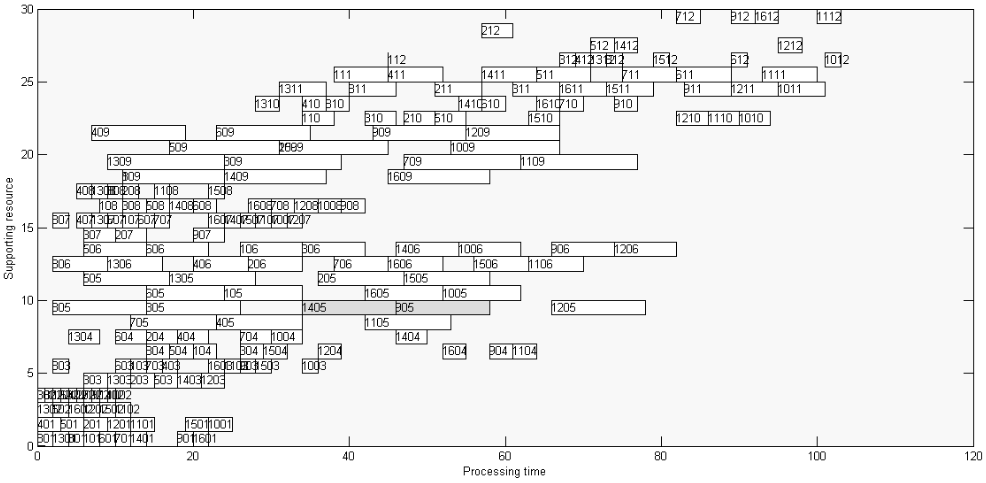

FIGURE 2. Gantt chart of robust pre-scheduling

Now assuming that the failure of the 10th aviation supporting resource occurs in the 45th minute, there are 2 operations directly affected by the failure, as shown in Fig. 2, the darker color parts. Due to the processing time of the operation '1405' (the 5th operation of the carrier-based aircraft numbered 14) is from 34 to 46 minute, the processing time of the operation '905' (the 5th operation of the carrier-based aircraft numbered 9) is 46-58 minute, while maintenance of the aviation supporting resource needs 8 minutes, so the supporting resource cannot work in 45-53 minute.

Since there are several supporting resources can support the 5th operation, thus, the rescheduling method of shifting to right and replacing failing resource can be used to deal with the affected operations, the result is shown in Fig. 3.

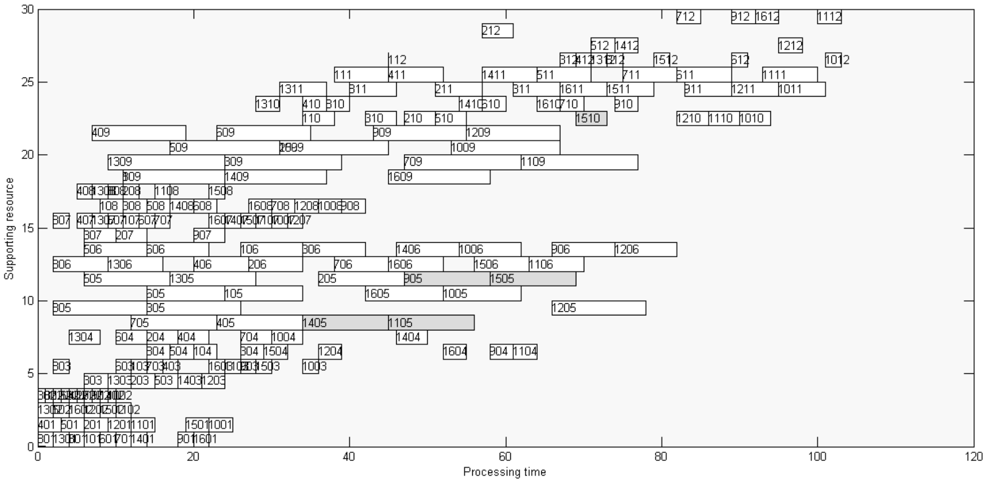

FIGURE 3. Gantt chart of rescheduling

Compared with the pre-scheduling scheme shown in Fig. 2, only five operations of '1405', '905', '1105', '1505', '1510' changed in Fig. 3, and the total time for all carrier-based aircraft to take-off is 103 minutes which shows the effectiveness of the reactive scheduling.

8 Conclusion

Based on the supporting procedure of the "one-stop" carrier-based aircraft, a robust pre-scheduling and rescheduling is presented to deal with the supporting resource failures with two strategies: inserting the idle time in the pre-scheduling, shifting to right and replacing supporting resource in the rescheduling. The experiment shows the feasibility of the model and the effectiveness of the improved algorithm.

References

[1] A. Jewell, Sortie generation capacity of embarked airwings. Center for Naval Analyses,1998.

[2] J. P. LIN, “Research on aircraft sortie generation rate based on discrete event,” Ph.D. thesis, Harbin Engineering University, 2011.

[3] C. Q. WEI, C. L. CHEN and B. R. WANG, Control Engineering of China19, 108-111(2012).

[4] C. Q. WEI, C. L. CHEN and B. R. WANG, Command Control & Simulation34(3), 23-27(2012).

[5] K. Z. YUE, C. SUN and M. Q.LUO, et al, Journal of Beijing University of Aeronautics and Astronautics39(8), 1063-1069(2013).

[6] S. D. Wu, R. H. Storer and P. C. Chang, Computers & Operations Research 20 (1), 1–14(1993).

[7] P. Kouvelis and Y. Gang, Springer Science & Business Media14 (2013).

[8] R. Storn and K. Price, Differential evolution-a simple and efficient adaptive scheme for global optimization over continuous spaces.[R]Technical report International Computer Science Institute, Berkley, 1995.

[9] N. B. Ho, J. C. Tay, E. M. K. Lai, European Journal of Operational Research179, 316-333(2007).

[10] G. H. ZHANG, “Research on Methods for Flexible Job Shop Scheduling Problems,” Ph.D. thesis, Huazhong University of Science and Technology, 2009.

[11] C. Y. ZHANG, Y. Q. RAO and X. J. LIU, et al, China Mechanical Engineer15(23), 2149-2153(2004).

[12] G. Y. SHI, International Journal of Systems Science 28(1), 25-32(1997).