ISSN : 0974 - 7435
Volume 10 Issue 24
2014
BioTechnology
An Indian Journal
FULL PAPER
BTAIJ, 10(24), 2014 [15855-15865]

Study on examination center scheduling problem based on genetic algorithm

Tao Ze[1], Liang Ying[2*]
[1]School of Mechanical Engineering, Shenyang Ligong University, Shenyang 110159, (CHINA)
[2]School of Information Science and Engineering, Shenyang Ligong University, Shenyang 110159, (CHINA)
Email: liangying.slg@163.com

ABSTRACT

A new hybrid flow shop scheduling problem is studied, which is abstracted from the physical examination centre. The problem is characterized by multi-stage unrelated parallel rooms and continuous examination. Firstly, the math model of this hybrid scheduling problem is constructed. And then the computation method of examination period is given. In order to keep examining continuously, the examination period is obtained through reversely pushing from the examination scheduling problem with multi-stage unrelated parallel rooms and examining interruption. Then the scheduling results are obtained based on genetic algorithm and simulated annealing algorithm. Function objective of the proposed method was to minimize the examination period. Finally scheduling results based on two cases show that the method proposed in this paper is effective and feasible.

KEYWORDS

Hybrid flow shop scheduling; No-wait; Parallel rooms; Examination period.

INTRODUCTION

In recent years, with the improvement of person's living standard, they pay more attention to health care. But when they go to the physical examination center, they usually wait for long time because of the limited resources of the physical examination center, so the satisfaction will be decreased. The less person have to wait, the better is the person experience. According to the examination process, some examinations must be checked first, such as drawing blood, and so on. The examination process scheduling can be equated to a no-wait hybrid flow shop scheduling problem. In order to shorten the waiting time, it is assumed to be a no-wait hybrid flow shop scheduling problem.

Hybrid flow shop scheduling problem (HFSP) is extended based on classical flow shop scheduling, in which the parallel machines are considered. Due to its application in petroleum, chemical engineering, metallurgy, and rail transportation, HFSP has been widely studied by many researchers[1-11]. Liang et al.[1] proposed a dynamic multi-swarm particle swarm optimizer for solving the blocking flow shop scheduling problem with the objective to minimize make-span. Rashidi et al.[2] addressed the hybrid flow shop scheduling problems with unrelated parallel machines, sequence-dependent set up times and processor blocking to minimize the make-span and maximum tardiness criteria. Akhshabi[3] proposed a particle swarm optimization algorithm based on memetic algorithm that hybridizes with a local search method for solving a no-wait flow shop scheduling problem. The bi-objective hybrid flow shop problem with sequence-dependent set up times and limited buffers was mentioned in[4]. In that environment, there are limited buffer spaces between any two successive stages; thus, maybe there is not enough room for queues of jobs that are waiting in the system for their next operations.

For some flow shop scheduling problems, the operations of a job have to be processed continuously from start to end without interruption. That is, once, the first operation is processed over on one machine, then transported to another machine to process the next operation till finish the last operation. Such flow shop scheduling problems are called no-wait flow shop scheduling problems[4-5]. Pan et al.[6] proposed an ant-colony heuristic algorithm for the no-wait flow shop problem with make-span criterion. A permutation was constructed according to trail intensities and solution best so far and a local search based on multi-insert. SONG and TANG[7], for the no-wait constraint between two sequential operations of a job, designed the no-wait algorithm of grading, and the number restriction of machines was embedded into the algorithm. On this basis, the discrete particle swarm optimization (DPSO) algorithm was proposed for the first time to solve such problems. PAN et al.[9] developed a speed-up method with the computational complexity for calculating the make-span of a permutation, and presented a discrete particle swarm optimization algorithm for solving no-wait flow shop scheduling problem. Bassem et al.[10], proposed a hybrid genetic algorithm to minimize the make-span and the total flow time in the no-wait flow shop scheduling problem. The variable neighborhood search was used as an improvement procedure in the last step of the genetic algorithm.

In this paper, the optimization of the no-wait hybrid flow shop scheduling problem with multi-stage unrelated parallel machines (NWHFSP-UPM) is investigated based on genetic algorithm and simulated annealing algorithm (GASA). The objective is to plot a processing routing for each person, and to find a permutation of person that minimizes the maximum completion time of the physical examination center.

SCHEDULING MODEL

Problem description

NWHFSP-UPM corresponds to the physical examination center can be defined as follows. n person are to be scheduled on p stages, and each person has the same p examinations. Each examination is specified by the required rooms and the checking time. Each stage involves m_j examination rooms, and the person checking time is unrelated on different rooms of the same stage. The examination of each person has to be checked without interruptions between consecutive rooms. Following assumptions are considered for NWHFSP-UPM:

Person are available at zero time;
Person checking cannot be interrupted;
At least one stage exists parallel machines;
Parallel rooms on the same stage are unrelated in capability and checking rate;
Each room can check only one person at the same time;
Each person must be checked on each stage, but the checking room is randomly;
The checking time of each person examination is pre-specified;
One person cannot be checked on more than one room at the same time;
Rooms are available for examining at a stage immediately after examining completion at the previous stage.

Mathematical model

The scheduling objective is:

$$\min Z. \quad \textbf{(1)}$$

Constrained conditions:
the last examination of person *i*

$$T_{ijk} \le Z \quad \textbf{(2)}$$

the non-last examination of person *i*

$$T_{ijk} - T_{i(j-1)g} = t_{ijk} \quad \forall i, j, g, \quad j \ne 1 \quad \textbf{(3)}$$

Examination of person *i* and person *q* in the stage *j* both need checking in the room *k*

$$T_{ijk} - T_{qjk} \ge t_{ijk} \quad \textbf{(4)}$$

$$T_{qjk} - T_{ijk} \ge t_{qjk} \quad \textbf{(5)}$$

It is necessary to ensure that only one parallel room is selected on each stage for every person

$$\sum_k X_{ijk} = 1 \quad \textbf{(6)}$$

Any examination of person *i* on the room *k*

$$T_{ijk} > 0 \quad \textbf{(7)}$$

where, Z is the total completing time of the scheduling, and $Z = \max\{C_1, C_1, \cdots, Cn\}$; C_i is the completion time of person *i*. *i* represents one person, $i = 1,2,\cdots n$; m_j is the parallel machine number of each stage, $j = 1,2,\cdots p$, p is the total number of stage; t_{ijk} and T_{ijk} represents checking time and completion time of person *i* for its *jth* examination on room *k*. Equation (2) is a natural constraint to ensure that last examinations should be completed before make-span; (3) is to ensure examination *j-1* on machine g precedes the next examination *j* on machine *k* for the same person, and there is no interruption between examination *j-1* and examination *j*; (4) and (5) are to ensure that each machine can check at most one person at a time; (4) denotes that examination *j* of person *q* precedes examination *j* of person *i* on the machine *k*; (5) denotes the opposite processing sequence; (6) denotes that on each stage, each person can be checked on at most one room at any time; (7) is to ensure the completing time of any examination of person *i*.

Computation of examination period

The computation formulas of completing time for NWFS (no-wait flow shop) are given in[8-9]. But the examination period in this paper can't be computed with above formulas, because there are unrelated parallel examination rooms. The comparison graphs between NWFS and NWHFSP-UPM are given in Figure 1 and Figure 2. In NWFS, where d_{ij} is the difference between finishing time of job *i* and starting time of its adjacent job *j*, and are the parallel examination rooms in the same stage. Because of the parallel machines, the period of NWHFSP-UPM is bit shorter than that of NWFS.

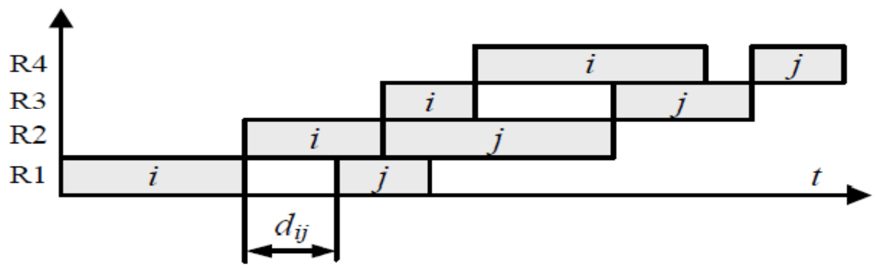

Figure 1: The scheduling of NWFS

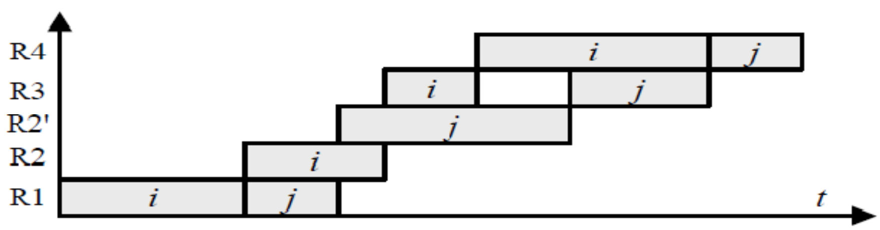

Figure 2: The scheduling of NWHFSP-UPM

Due to NWHFSP-UPM is a special case in hybrid flow shop problem with unrelated parallel machines (HFSP-UPM), the examination period can be computed through HFSP-UPM.

The computation of examination period:

Step 1 In stage 1, select the first starting person, if there are the parallel persons, then choose the person with short examination time.

Step 2 After selecting the person, then in the last stage, searching for the ending time of the person.

Step 3 Set i=1

Step 4 Computing the starting time of the person reversely push from the end time of last stage.

Step 4.1 If $c_{i(j-1)k} \le s_{ijg}$;

then $c_{i(j-1)k} = s_{ijg}$; $(j = p, p-1, \cdots, 1)$

Repeat step 4.1, compute the starting time of person *i* from stage *p* to stage 1.

Step 4.2 If $c_{i(j-1)k} > s_{ijg}$ when $j = h; (2 \le h \le p)$ then $s_{ijg} = c_{i(j-1)k}; (j = h, \cdots, p)$

Repeat step 4.2 until s_{ihg} to s_{ipg} are all replaced, then go to step 4.1.

Step 5 Update i=i+1, if $i > n$, go to step 6; otherwise, in stage 1, select the next starting person, if there are the parallel persons, then choose the person with short examination time, and go to step 2.

Step 6 Stop and output the final sequence and starting time and examination period.

where *k* and *g* are the parallel examination rooms or the same rooms in the same stage.

GASA

There are a lot of studies in the literature which use genetic algorithm and simulated annealing algorithm for the job shop and flow shop scheduling problems[2,10-11]. In theory, GA and SA are both based on probability distributions. SA is a generic probabilistic meta-heuristic for global optimization that locates a good approximation to the global maximum/minimum of a given function in a large search space. It is usually much more efficient than the exhaustive brute-force search for such search. SA reaches an almost zero probability jumping phenomenon through endowing a key time variety in searching, and avoiding converging to a local optimum and attaining total optimization. GA realizes optimization through population generic operation based on survival of the fittest. GASA hybrid algorithm is good for searching for an optimum process, enhancing whole and local search ability and efficiency; it has excellent search ability, efficiency and high reliability. The flow chart of GASA is given is Figure 3.

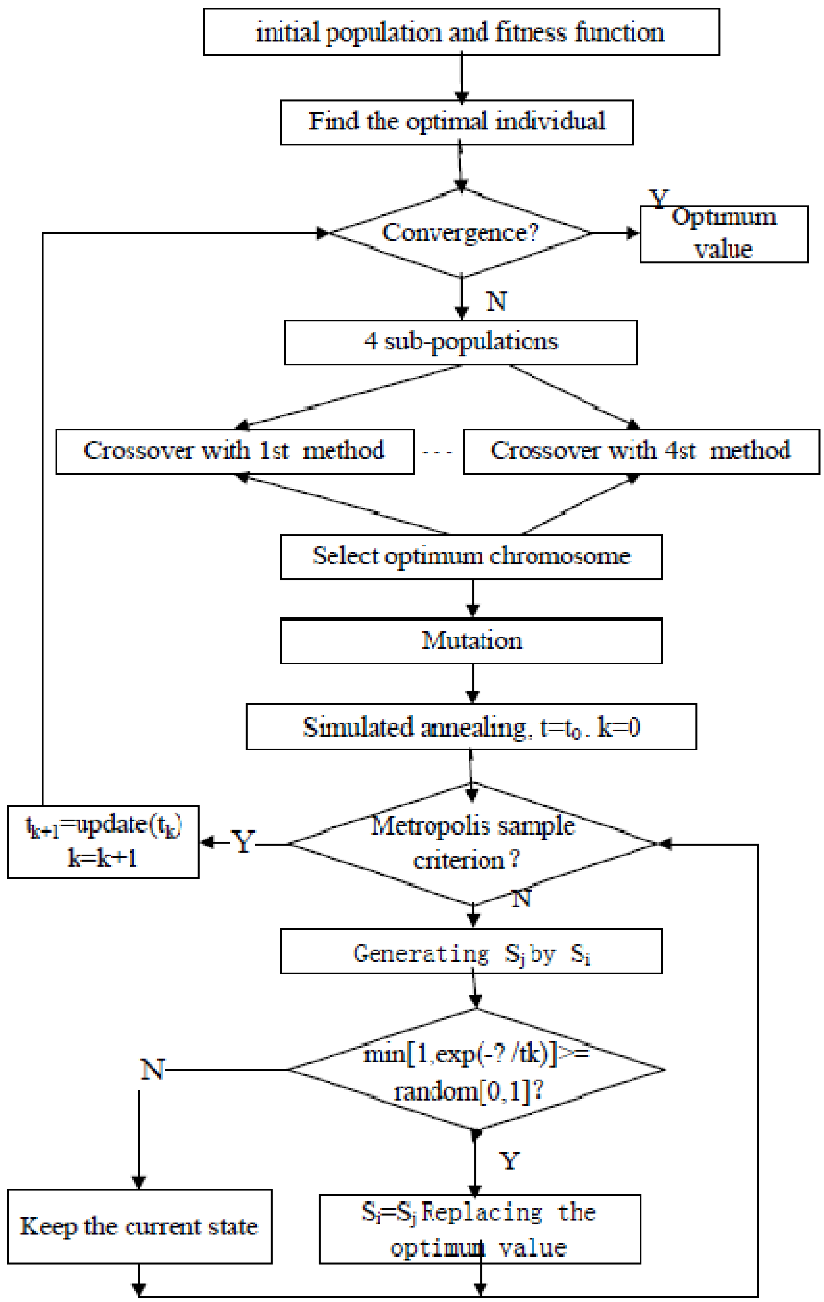

Figure 3

The details of GASA are as follows:

Permutation representation represents a solution of a problem as chromosome, and the permutation representation is given based on the job process. The order of the jobs in the permutation denotes the processing order of the jobs on the machines. Its length is equal to the summation of each job process.

Create initial population and fitness function. The initial population is generated at random. The larger is the population size, the easier optimal result can be found. The computation time of computer is proportional to the population size.

Crossover operation: The crossover operation is the most important operation in genetic algorithm. It produces new individuals from the selected parents, in order to obtain better results. The crossover probability is the basic parameter in the crossover procedure. It is a parameter deciding how many chromosomes will be performed crossover. Before the operation, divide population into K sub-populations (4 sub-populations in this paper): select the optimum individual in each sub-population and crossover with other individuals until producing K new populations, then select the best individual. Adopt MPPX, MGOX, MGPMX1, and MGPMX2[12] in simulation, which can make the population have obvious diversity.

Selection. Select optimum chromosome between offspring chromosome produced by crossover in different sub-population and parent. Too much selection may induce algorithm finishing prematurely, and not enough selection may induce slow convergence of the algorithm.

Mutation: This operation helps the algorithm to escape from a local optimum by the process of diversification. It introduces new genetic structures in an individual by a random modification of its string. INV mutation is used to produce small perturbations on chromosomes.

Metropolis sample process. The stochastic rule, in which the probabilistic decision is made to prevent the optimization process from sticking at a possible local minimum, is the main idea behind simulated annealing. The probability of accepting a new individual under the circumstances: min {1, exp $(-\Delta/t)\}>$random[0,1] (here t denotes the temperature, Δ denotes difference of aim value between new and old). Initially, the temperature t is high enough permitting many deteriorative moves to be accepted, and then it is lowered at a low speed of rate to the point which the inferior moves are approximately rejected. This algorithm sequentially and slowly investigates the possible neighbors in each temperature in order to find the best solution.

Enhancing memory ability: In order to avoid losing the current optimum solution in the search process, save the current optimum solution through adding memory.

Cooling scheme: Exponential cooling, $t_k = \lambda t_{k-1}$, and λ =0.9 in simulation.

Criterion of temperature changing and algorithm termination: In order to adapt dynamic variety of algorithm performance, pay attention to optimum and time performance, adopt two criteria of temperature changing and algorithm termination. In the optimizing process, if the optimum value remains constant through 30 eras then start cooling; if the optimum value remains constant through cooling 30 iterations then stop searching, the optimum value is the solution.

Due to there are parallel examination rooms, the examination room which has earlier ending time is selected optimally.

CASE STUDY

Case 1

In order to test the validity and feasibility of GASA and the method proposed in this paper, one case in[13] is applied. There are three stages, in each stage, the quantity of parallel machines is 3, 2, 4, respectively. The performance of parallel machines is unrelated. The detailed information of jobs and machines are given in TABLE 1. Firstly, the GASA algorithm is tested and compared with GA[14], SFLA[15], and EDA[13]. The scheduling results are obtained based on GASA with minimizing the scheduling period as function objective. The algorithm parameters are that population size is 100, crossover rate is 0.8, mutation rate is 0.01. The results are given in TABLE 2. The results correspond to the HFSP-UPM.

TABLE 1: Processing information of jobs

Job	Stage 1		Stage 2			Stage 3			
	M1	M2	M3	M4	M5	M6	M7	M8	M9
1	2	2	3	4	5	2	3	2	3
2	4	5	4	3	4	3	4	5	4
3	6	5	4	4	2	3	4	2	5
4	4	3	4	6	5	3	6	5	8
5	4	5	3	3	1	3	4	6	5
6	6	5	4	2	3	4	3	9	5
7	5	2	4	4	6	3	4	3	5
8	3	5	4	7	5	3	3	6	4
9	2	5	4	1	2	7	8	6	5
10	3	6	4	3	4	4	8	6	7
11	5	2	4	3	5	6	7	6	5
12	6	5	4	5	4	3	4	7	5

TABLE 2: Scheduling results and comparisons

No.	1	2	3	4	5	6	7	8	9	10
GA	30	27	26	27	29	27	26	27	26	28
SFLA	24	24	24	24	24	24	24	24	24	24
EDA	23	24	23	23	23	23	24	24	23	24
GASA	23	23	23	24	24	23	24	24	23	23

From TABLE 2, it shows that not only GASA has better solutions than conventional GA and SFLA, but also the stability of GASA is much better than that of GA, the optimal solution can be found Z=23. Compared with EDA, the results performance of GASA is roughly equivalent to that of EDA. The validity and feasibility of GASA has been verified. And then the scheduling of NWHFSP-UPM is tested based on the math model and the computation method of period proposed in this paper. The case is the same as above example. The comparison Gantt graphs between HFSP-UPM and NWHFSP-UPM are given in Figure 4 and Figure 5, respectively. The optimal result of NWHFSP-UPM is 23. Due to the operations of a job have to be processed continuously from start to end without interruption, usually, the idle time of machines is longer than that of HFSP-UPM in Figure 4. From Figure 4, it shows that stage 2 is the bottle neck during the processing. Due to the processing sequence of jobs on stage 2 in Figure 5 is the same as that in Figure 4, the production period are both 23, there is no idle time on stage 2 once the operation starting. One of the suboptimal results of NWHFSP-UPM is given in Figure 6, the result is 24. The result is bit longer than that in Figure 5, the reason is that there is idle time on bottle neck stage 2, that is, there is idle time on machine 4 and machine 5.

The unrelated parallel machines lead to the different results in Figure 5 and in Figure 6. For example, for job 8, the processing time are 3 and 4 on machine 1 and machine 3, respectively. Although the processing time is short on machine 1, the starting time and finishing time of job 8 is later in Figure 6 than that in Figure 5. Except job 8, the processing sequences of other jobs are same. Due to the operations of all jobs have to be processed continuously from start to end without interruption, the result in Figure 6 is bit longer than that in Figure 5. The reason is the scheduling of job 8 on stage 1.

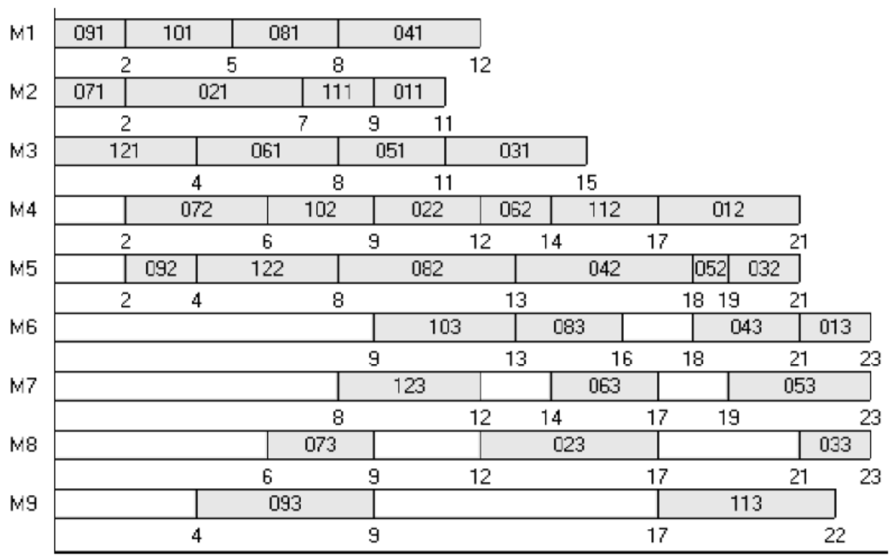

Figure 4: Gantt graph of HFSP-UPM

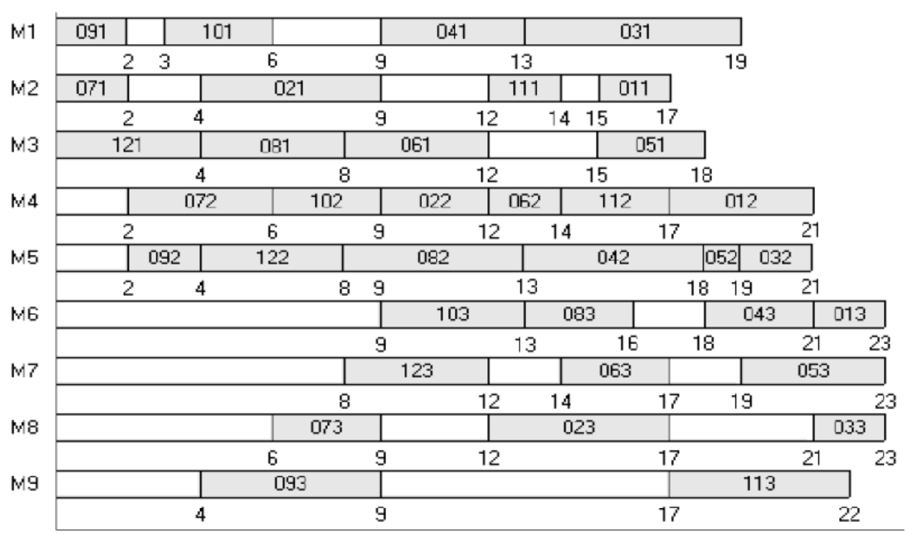

Figure 5: Gantt graph of NWHFSP-UPM

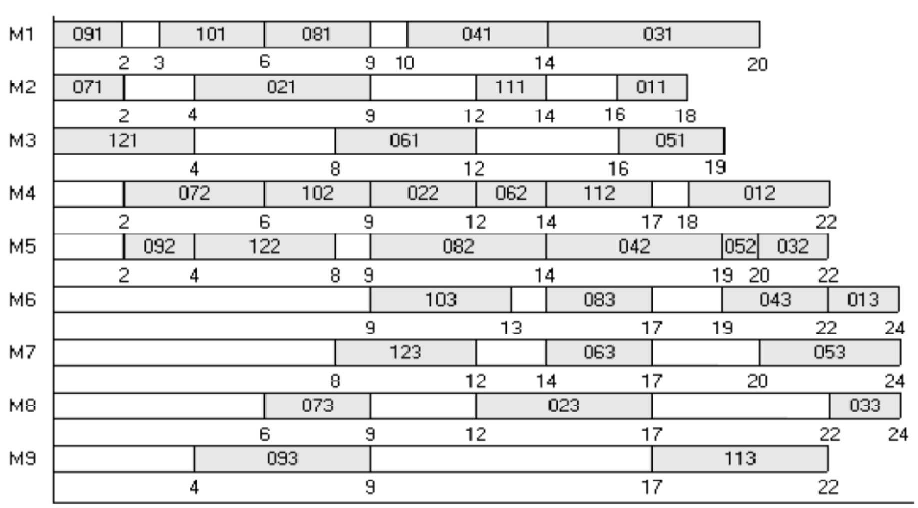

Figure 6: Gantt graph of NWHFSP-UPM

Case 2

In order to further test the effectiveness of the proposed approach, another case of examination center is tested in this section. There are three stages, in each stage, the parallel examination rooms is 2, 3, 2, respectively. The examination time of each person in different room is different. The examination time is unrelated among the parallel rooms. It related with person and doctors. The results can be represented as a Gantt chart, which depicts the starting and ending time of all examinations on each examination room. The data and schedule are provided in TABLE 3.

TABLE 3: The examination information of person in different stages

person	Stage 1		Stage 2			Stage 3	
	room1	room 2	room 3	room 4	room 5	room 6	room 7
1	4	6	3	5	6	4	5
2	2	3	6	5	4	3	4
3	5	5	3	4	4	5	5
4	8	6	3	3	3	3	3
5	2	4	5	7	5	5	6
6	4	4	6	5	6	6	7
7	7	5	4	3	3	7	5
8	3	4	2	3	2	4	3
9	5	4	5	5	3	2	2

According to the math model and the computation method of examination period, the scheduling results are obtained based on GASA with minimizing the examination period as function objective. The algorithm parameters are that population size is 100, crossover rate is 0.8, mutation rate is 0.01. The results of NWHFSP-UPM and HFSP-UPM are given in Figure 7 and Figure 8.

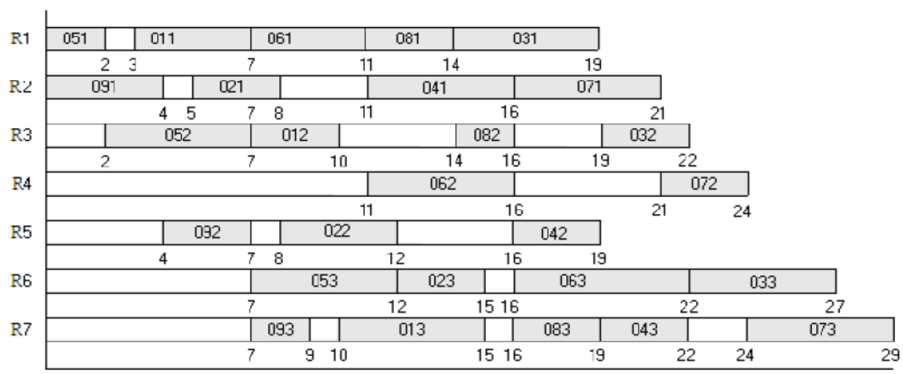

Figure 7: Gantt graph of NWHFSP-UPM

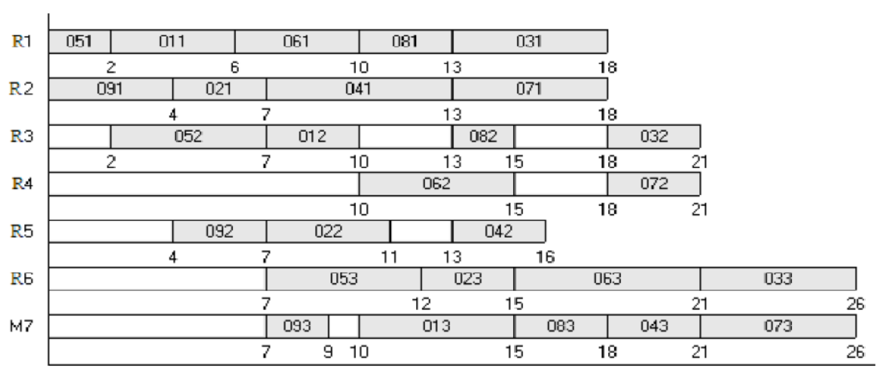

Figure 8: Gantt graph of NWHFSP-UPM

In Figure 7 and Figure 8, the basic data of examination centre and persons is same, but the examination period in Figure 8 is shorter than that in Figure 7. The examination sequences of persons are all same in Figure 7 and Figure 8, but due to the examinations of a person have to be examined continuously from start to end without interruption in Figure 7, the idle time in Figure 7 is longer than that in Figure 8. From Gantt graph 7 and 8, it shows that the bottle neck is stage 1.

CONCLUSION

In this paper, according to the practical examination demand of examination centre, a no-wait hybrid flow shop scheduling problem with multi-stage unrelated parallel machines (NWHFSP-UPM) is studied. The math model considered the examination continuity is constructed. The examining time in parallel rooms is different according to the conditions of persons and doctors. The evolution of examination sequence decision can be obtained using a GASA with examination period as the criterion based on the computation method of examination period. When these factors are considered during examination scheduling, generally, a more effective schedule can be obtained. Testing and analyzing the performance of scheduling results. Moreover, the approach was shown to be effective and feasible in practical case.

ACKNOWLEDGMENT

Project supported by the education department of Liaoning Province (L2012067).

CONFLICT OF INTERESTS

The authors declare that there is no conflict of interests regarding the publication of this article.

REFERENCE

[1] J.J.Liang, Quan-KePan, ChenTiejun, LingWang; Solving the blocking flow shop scheduling problem by a dynamic multi-swarm particle swarm optimizer, The International Journal of Advanced Manufacturing Technology, **55**, 755-762 **(2011)**.

[2] E.Rashidi, M.Jahandar, M.Zandieh; An improved hybrid multi-objective parallel genetic algorithm for hybrid flow shop scheduling with unrelated parallel machines, The International Journal of Advanced Manufacturing Technology, **49**, 1129-1139 **(2010)**.

[3] M.Akhshabi, R.Tavakkoli-Moghaddam, F.Rahnamay-Roodposhti; A hybrid particle swarm optimization algorithm for a no-wait flow shop scheduling problem with the total flow time, The International Journal of Advanced Manufacturing Technology, **70**, 181-1188 **(2014)**.

[4] Sina Hakimzadeh Abyaneh, M.Zandieh; Bi-objective hybrid flow shop scheduling with sequence-dependent setup times and limited buffers, The International Journal of Advanced Manufacturing Technology, **58**, 309-325 **(2012)**.

[5] Kai zhou Gao, Quan ke Pan, P.N.Suganthan, Junqing Li; Effective heuristics for the no-wait flow shop scheduling problem with total flow time minimization, The International Journal of Advanced Manufacturing Technology, **66**, 1563-1572 **(2013)**.

[6] Pan Quanxke, Zhao Baoxhua, Qu Yuxgui, Bi Yuxhui; Ant colony heuristic algorithm for no-wait flow shop problem with makespan criterion, Computer Integrated Manufacturing Systems, **13(9)**, 1801-1815 **(2007)**.

[7] Song Ji-wei, Tang Jia-fu; No-wait hybrid flow shop scheduling method based on discrete particle swarm optimization, Journal of System Simulation, **22(10)**, 2257-2261 **(2010)**.

[8] J.Grabowski, J.Pempera; Some local search algorithms for no-wait flow-shop problem with makespan criterion, Computers＆ Operations Research, **32(8)**, 2197-2122 **(2005)**.

[9] Pan Quan-Ke, Zhao Bao-hua, Qu Yu-Gui; Heuristics for the No-Wait Flow Shop Problem with Makespan Criterion, Chinese Journal of Computers, **31(7)**, 1147-1154 **(2008)**.

[10] Bassem Jarboui, Mansour Eddaly, Patrick Siarry; A hybrid genetic algorithm for solving no-wait flow shop scheduling problems, The International Journal of Advanced Manufacturing Technology, **54**, 1129-1143 **(2011)**.

[11] S.M.Mousavi, M.Zandieh, M.Yazdani; A simulated annealing/local search to minimize the makespan and total tardiness on a hybrid flowshop, The International Journal of Advanced Manufacturing Technology, **64**, 369-388 **(2011)**.

[12] J.Byung, R.Hyung., S.Hyun; A hybrid genetic algorithm for the job shop scheduling problems, Computers & Industrial Engineering, **45**, 597-613 **(2003)**.

[13] Wang Sheng-Yao, Wang Ling, Xuye, Zhou Gang; An Estimation of distribution algorithm for solving hybrid flow-shop scheduling problem, Acta Automatic Sinica, **38(3)**, 437-443 **(2012)**.

[14] Zhou Hui-Ren, Tang Wan-Sheng, Wei Ying-Hui; Optimize flexible flow-shop scheduling using genetic algorithm, Compute Engineering and Applications, **45(30)**, 224226 **(2009)**.

[15] Y.Xu, L.Wang, G.Zhou, S.Y.Wang; An effective shuffled frog leaping algorithm for solving hybrid flow-shop scheduling problem, In:Proceedings of the International Conference on Intelligent Computing. Zhengzhou, China: Springer, 560-567 **(2011)**.