Journal of Universal Computer Science, vol. 20, no. 14 (2014), 1926-1950
submitted: 12/1/14, accepted: 6/9/14, appeared: 1/12/14

An Effective Genetic Algorithm for Multi-objective Integrated Process Planning and Scheduling with Various Flexibilities in Process Planning

Xinyu Li
(The State Key Laboratory of Digital Manufacturing Equipment and Technology
Huazhong University of Science and Technology, Wuhan, China
lixinyu@mail.hust.edu.cn)

Xiaoyu Wen
(School of Mechanical and Electrical Engineering, Zhengzhou University of Light Industry
Zhengzhou, China
xiaoyuup@gmail.com)

Liang Gao
(The State Key Laboratory of Digital Manufacturing Equipment and Technology
Huazhong University of Science and Technology, Wuhan, China
Corresponding author: gaoliang@mail.hust.edu.cn)

Abstract: Process planning and scheduling are two of the most important functions in modern manufacturing system. Considering their complementarity, integrating them more tightly can improve the performance and productivity of the whole manufacturing system. Meanwhile, the multi-objective optimization problem is widespread existing in manufacturing system. In this paper, an effective genetic algorithm is proposed to optimize the multi-objective integrated process planning and scheduling (IPPS) problem with various flexibilities in process planning. Three types of flexibilities related to process, sequence and machine are considered. And three objectives including makespan, total machine workload and maximal machine workload are taken into account simultaneously. According to the model and characteristics of multi-objective IPPS, the framework of the proposed algorithm is designed to optimize three objectives simultaneously. Effective genetic operations are employed in the proposed algorithm. Pareto set is set to store and maintain the solutions obtained during the searching procedure, the proposed algorithm could get several Pareto optimal solutions during one searching process. Two experiments are employed to test the performance of the proposed algorithm. The experiment results show that the proposed algorithm can solve multi-objective IPPS problem with various flexibilities in process planning effectively and obtain good solutions.

Keywords: Integrated process planning and scheduling, Multi-objective optimization, Genetic algorithm
Category: J.6

1 Introduction

Process planning and scheduling are two of the most important activities in modern manufacturing system. Process planning transforms the product design into manufacturing instructions [Jung et al. 01]. The aim of process planning is to determinate the appropriate manufacturing resources and operations sequence for jobs. Most jobs may have a large number of process plans due to the flexibilities of machining process and sequences [Li et al. 13]. After the process plans of jobs are determined, the scheduling is to allocate the operations of all these jobs on machines over time by satisfying the precedence constraints in the process plans. It is clearly that there is a close interrelationship between process planning and scheduling. An optimal final schedule not only depends on allocating the machine resource over time but also lies on the results of process planning. Traditionally, process planning and scheduling were studied independently, which may generate following problems to hold back improving the performance and productivity of manufacturing system [Li et al. 10]:

- Firstly, the real-time shop floor situation is not taken into account during the off-line process planning, which may make the process plans invalid at the time of scheduling execution [Phanden et al. 11].
- Secondly, process planning system and scheduling system have different optimized objects. The purpose of process planning system is to provide a suitable process plan for a special job. But the purpose of scheduling is to arrange a group of jobs over time according to shop floor conditions. If process planning system executed separately, the influence from other jobs in workshop could not been taken into consideration. Therefore, the predefined process plans may be not suitable for scheduling system [Wang et al. 06].
- Thirdly, the objectives in process planning system and scheduling system are often conflicting. Process planning system focuses on minimal processing time or manufacturing cost for a job while scheduling system focuses on minimal makespan or other objectives. If one machine has the shortest processing time for all the operations, the optimal process plans with minimal processing time will be arranged in the same machine. As a result, the makespan will be longer.

So, decision maker could not get a satisfactory result for the whole manufacturing system if process planning and scheduling were optimized independently. In fact, integrated process planning and scheduling (IPPS) could overcome these above problems well. IPPS could bring significant improvement to the efficiency of manufacturing through removing resources conflicts, decreasing flow-time and work-in-process, improving production resources utilizing and adapting to irregular shop floor disturbances [Li et al. 12]. Therefore, it is important to integrate process planning and scheduling more closely to achieve the global optimum in manufacturing system.

With the development of market economy, competitions among manufacturers become more and more intense. In order to enhance their competitiveness, manufacturers often need to meet the diverse needs from customers, such as faster processing speed and better quality. Meanwhile, enterprise managers want to reduce manufacturing cost and improve the utilization of machines. Only considering the single objective could not meet the demand from the real-world production [Li et al. 10]. There are many objectives existing in IPPS, such as makespan, total workload of machines, maximal machine workload, lateness etc. Decision makers always need to make a trade-off among different objectives while determining a final schedule.

As both of process planning and job shop scheduling are NP-hard problems, IPPS becomes even more complex with various flexibilities in process planning [Wang et al. 06]. Moreover, multi-objective IPPS is concerned with optimizing multi-objectives simultaneously, which lead the problem much more complicated. Therefore, the research work on multi-objective IPPS is significant both in researches and applications. However, most related works of IPPS in recent years are concerned with single objective [Lian et al. 12] [Wong et al. 12] [Lv et al. 14], there are few research works focusing on multi-objective IPPS problem. [Morad and Zalala 99] used genetic algorithm based on weighted-sum method to optimize multi-objective IPPS. Sequence flexibility and machine flexibility were considered in process planning. [Baykasoğlu et al. 04] utilized multi-objective tabu search to optimize makespan, maximal machine workload and total machine workload in IPPS. [Baykasoglu and Ozbakir 09] proposed a grammatical optimization approach which made use of generic process planning and multi-objective tabu search to optimize multi-objective IPPS problem. The process plans of each job were predefined in [Baykasoğlu et al. 04] and [Baykasoglu and Ozbakir 09]. [Li and McMahon 07] proposed simulated annealing algorithm based on weighted-sum method to optimize multi-objective IPPS. Process plans of each job in this paper were generated dynamically according to the operation flexibility, sequence flexibility and machine flexibility. [Rajkumar et al. 10] proposed GRASP algorithm to solve IPPS problem in flexible job-shop scheduling. [Zhang and Fujimura 10] proposed an improved vector evaluated GA to optimize multi-objective IPPS problem. [Wang et al. 10] presented a PSO-based multi-objective optimization approach to IPPS problem. [Mohammadi et al. 11] presented a slot-based multi-objective MILP model for IPPS and a multi-objective simulate annealing algorithm was proposed to deal with multi-objective IPPS problem. [Mohapatra et al. 13] employed NSGA-II to settle multi-objective IPPS problem in reconfigurable manufacturing settings.

There are some shortcomings existing in above research works. First, the interactions and information sharing scheme between process planning and scheduling was not enough. More effective integration scheme between two functions should be used to optimize multi-objective IPPS. Second, most of the above research works did not consider the flexibilities existing in process planning adequately. This will reduce the solution space of multi-objective IPPS during the optimization procedure. To overcome the above shortcomings, in this paper, an effective genetic algorithm is proposed to optimize multi-objective IPPS problem with various flexibilities in process planning. Three types of flexibilities related to process, sequence and machine and three objective including makespan, total machine workload and maximal machine workload are taken into account simultaneously. According to the model and characteristics of multi-objective IPPS, the framework and operators of the proposed algorithm is designed to optimize three objectives simultaneously.

The remainder of this paper is organized as follows: problem description of multi-objective IPPS is given in [Section 2]. The workflow of the proposed algorithm and the detailed components in the proposed algorithm are described in [Section 3].

Experiments and discussions are given in [Section 4] while [Section 5] is the conclusion and future works.

2 Multi-objective IPPS Description

2.1 IPPS Description

Suppose there are *n* jobs need to be produced on *m* machines. Each job has various operations and alternative manufacturing resources. The aim of IPPS is to select suitable manufacturing resources for each job, determine the operations' processing sequence and the start time of each operation on each machine by satisfying the precedence constraints among operations and achieving several corresponding objectives [Guo et al. 09].

Jobs	**Processing information**				
	Features	**Candidate Operations**	**Candidate Machines**	**Process Time**	**Precedence Constraints**
Job 1	F_1	O_1	M_1,M_2,M_3	4,3,6	Before F_2,F_3
		O_2-O_3	M_2,M_3/M_1,M_2	2,3/3,5	
	F_2	O_4	M_2,M_3,M_4	8,10,9	
		O_5-O_6	M_3,M_5/M_3,M_4	4,3/5,7	
	F_3	O_7	M_1,M_4	8,9	Before F_4
	F_4	O_8-O_9	M_3,M_5/M_1,M_5	7,9/5,8	
		O_{10}	M_4,M_5	14,19	
	F_5	O_{11}	M_1,M_3,M_4,M_5	20,17,19,23	
	F_6	O_{12}	M_1,M_4,M_5	18,13,17	
Job 2	F_1	O_1-O_2	M_2,M_3/M_4	3,6/4	Before F_3
	F_2	O_3	M_1,M_2	3,5	Before F_3
		O_4-O_5	M_1,M_3/M_2,M_5	10,9/7,12	
	F_3	O_6	M_3,M_4	7,12	
		O_7-O_8	M_1,M_3/M_2	5,6/10	
Job 3	F_1	O_1	M_2,M_4	4,7	Before F_3,F_4
	F_2	O_2	M_1,M_5	10,8	
	F_3	O_3-O_4	M_3,M_4/M_1,M_2	14,15/13,16	Before F_4
		O_5	M_4,M_5	12,14	
	F_4	O_6-O_7	M_1,M_3/M_2,M_3	17,19/20,16	

Table 1: Processing Information for 3 Jobs Machined in 5 Machines

In order to describe the mathematical model clearly, the following assumptions should be given at first:

1) Jobs and machines are independent among each other. All jobs have the same priorities.
2) Each machine can only handle one operation at a time.
3) Different operations from one job can't be processed at the same time.
4) One operation can't be interrupted when being processed.
5) All the jobs and machines are available at time zero.
6) The transport time is negligible.
7) The setup time for the operations is independent of the operation sequence and is included in the processing time.

Based on these assumptions, the mathematical model of multi-objective IPPS addressed in this paper is stated as follows, which is referred from [Li et al. 2012].The maximal completion time of machines (makespan), the maximal machine workload (MMW) and total workload of machines (TWM) are taken into account for multi-objective IPPS. In this paper, the aim of multi-objective IPPS is to minimize these three objectives simultaneously.

The notations used to explain the model are described below:

n: total number of jobs

m: total number of machines

g_i: total number of alternative process plans of job i

p_{il}: number of operations in the lth alternative process plan of the job i

o_{ijl}: the jth operation in the lth alternative process plan of job i

k: alternative machine corresponding to o_{ijl}

t_{ijlk}: the processing time of operation o_{ijl} on machine k

c_{ijlk}: the earliest completion time of operation o_{ijl} on machine k

w_k: the workload of machine k

A: a very large positive number

$$X_{il} = \begin{cases} 1 & \text{the } l\text{th flexible process plan of job } i \text{ is selected} \\ 0 & \text{otherwise} \end{cases}$$

$$Y_{ijlpqsk} = \begin{cases} 1 & \text{the operation } o_{ijl} \text{ precedes the operation } o_{pqs} \text{ on machine } k \\ 0 & \text{otherwise} \end{cases}$$

$$Z_{ijlk} = \begin{cases} 1 & \text{if machine } k \text{ is selected for } o_{ijl} \\ 0 & \text{otherwise} \end{cases}$$

The following five objectives are considered to be optimized simultaneously.

(1) f_1: Minimizing the maximal completion time of machines (makespan):

$$\text{Min } f_1 = makespan = Max\ c_{ijlk} \qquad (1)$$
$$i \in [1,n],\ j \in [1, p_{il}],\ l \in [1, g_i],\ k \in [1,m]$$

(2) f_4: Minimizing the maximal machine workload (MMW):

$$\text{Min } f_2 = \text{MMW} = \max w_k \quad k \in [1,m] \qquad (2)$$

(3) f_5: Minimizing the total machine workload (TMW):

$$\text{Min } f_3 = \text{TMW} = \sum_{k=1}^{m} w_k \quad k \in [1,m] \qquad (3)$$

Subject to:

(1) Operation constraint: different operations of one job can't be processed at the same time.

$$(c_{ijlk_0} \times Z_{ijlk_0} \times X_{il}) - (c_{i(j-1)lk_1} \times Z_{i(j-1)lk_1} \times X_{il}) + A \times (1 - X_{il}) \ge (t_{ijlk_0} \times Z_{ijlk_0} \times X_{il}) \qquad (4)$$
$$i \in [1,n],\ j \in [1, p_{il}],\ l \in [1, g_i],\ k_0, k_1 \in [1,m]$$

(2) Machine constraint: Each machine can only handle one operation at a time.

$$
\begin{aligned}
&(c_{pqsk} \times Z_{pqsk} \times X_{ps}) - (c_{ijlk} \times Z_{ijlk} \times X_{il}) + A \times (1 - X_{il}) + A \times (1 - X_{ps}) \\
&+A \times (1 - Y_{ijlpqsk} \times Z_{pqsk} \times X_{ps} \times Z_{ijlk} \times X_{il}) \geq (t_{pqsk} \times Z_{pqsk} \times X_{ps}) \\
&(c_{ijlk} \times Z_{ijlk} \times X_{il}) - (c_{pqsk} \times Z_{pqsk} \times X_{ps}) + A \times (1 - X_{il}) + A \times (1 - X_{ps}) \\
&+A \times (Y_{ijlpqsk} \times Z_{pqsk} \times X_{ps} \times Z_{ijlk} \times X_{il}) \geq (t_{ijlk} \times Z_{ijlk} \times X_{il}) \\
&i, p \in [1, n],\ \ j, q \in [1, p_{il,ps}],\ l, s \in [1, g_{i,p}],\ k \in [1, m]
\end{aligned} \tag{5}
$$

(3) Process plan constraint: Only one alternative process plan can be selected for job *i*.

$$\sum_{l=0}^{g_{il}} X_{il} = 1 \quad l \in [1, g_{il}] \tag{6}$$

Table 1 gives the processing information for 3 jobs machined in 5 machines. Each job has various features, alternative operations and machines. Precedence constraints are existing among different features. The outcome of process planning is the specific process plans for jobs. Then the scheduling system will arrange the jobs overtime according to the specific process plan for each job.

2.2 Multi-objective Optimization

The general multi-objective optimization problem (MOP) is defined as follows:

$$
\begin{aligned}
&\text{Minimize} \quad f(x) = \{f_1(x), f_2(x), ..., f_k(x)\} \\
&\text{Subject to:} \quad g_j(x) \leq 0,\ \ j = 1, 2, ..., m \\
&\qquad\qquad\qquad x \in X,\ \ f(x) \in Y
\end{aligned} \tag{7}
$$

k is the number of objectives, *m* is the number of inequality constraints, *x* is the decision variable, *f*(*x*) is the objective. *X* is the decision space, *Y* is the objectives space. In MOP, for decision variables *a* and *b*, *a* dominates *b*:

$$
\begin{aligned}
&\text{iff} \quad f_i(a) \leq f_i(b) \quad \forall i \in (1, 2, ..., k) \\
&\qquad\ \ f_j(a) < f_j(b) \quad \exists j \in (1, 2, ..., k)
\end{aligned} \tag{8}
$$

a and *b* is non-dominated:

$$\text{iff} \ \ f_i(a) \leq f_j(b) \,\&\, f_i(a) \geq f_j(b) \ \ \forall i, j \in \{1, 2, ..., k\} \tag{9}$$

A solution *x** is called the Pareto optimal solution if no solution in the decision space *X* can dominate *x**. The Pareto optimal set is formed by all the Pareto optimal solutions. The target of MOP is to find a finite number of Pareto optimal solutions instead of a single optimum in single objective optimization problem.

3 Proposed Genetic Algorithm for Multi-objective IPPS

3.1 Workflow of the Proposed Algorithm

An effective genetic algorithm is proposed to solve multi-objective IPPS problem with various flexibilities in process planning effectively. The workflow of the proposed algorithm is given in Figure 1. The main steps of the proposed algorithm are described as follows.

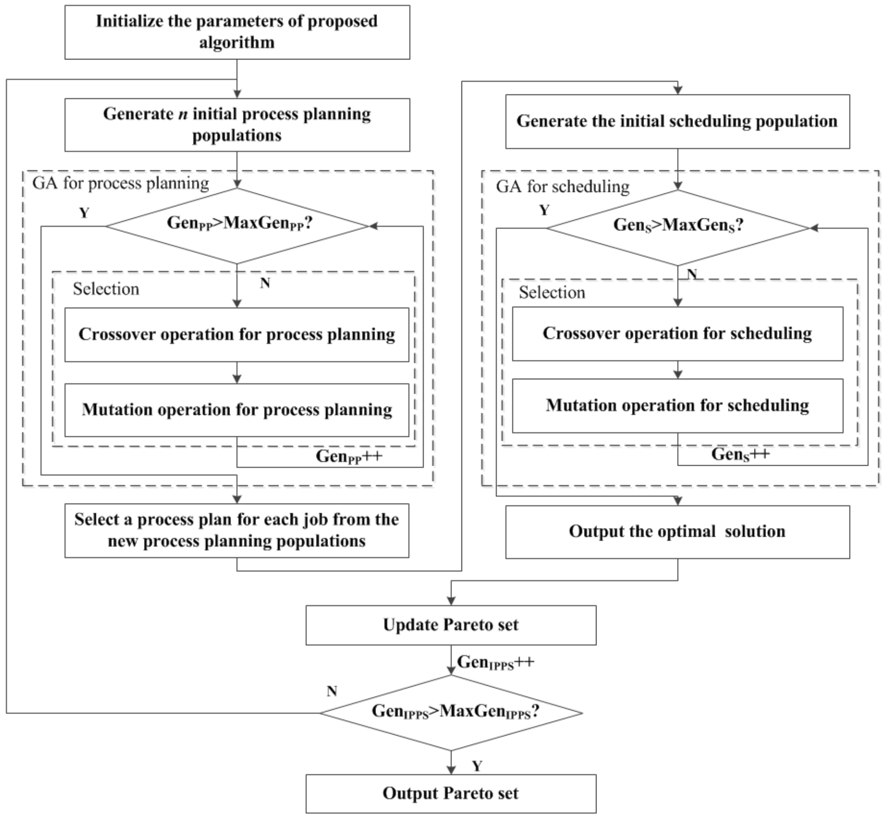

Figure 1: Workflow of the Proposed Algorithm

Step 1: Set the parameters of the proposed algorithm, including the size of the process planning population ($PopSize_{PP}$), the size of the scheduling population ($PopSize_S$), the size of the Pareto set (*ParetoSet*), maximum generations for IPPS($MaxGen_{IPPS}$), maximum generations for process planning ($MaxGen_{PP}$) , maximum generations for scheduling($MaxGen_S$), crossover probability for process planning (PP_c), crossover probability for scheduling (SP_c), mutation probability for process planning (PP_m), mutation probability for scheduling (SP_m).

Step 2: Generate *n* initial populations of flexible process planning for *n* jobs respectively.

Step 3: Generate new population for each job by GA respectively.

Step 4: For each job, select a process plan from the corresponding population randomly.

Step5: According to the determinate process plan for each job, generate the initial population for scheduling.

Step 6: Optimize the scheduling plan by GA. Output the optimal solution in scheduling.

Step 7: Compare the obtained solution with the solutions in the Pareto set, and then use the Pareto set update scheme to update the solutions. The Pareto set update scheme will be given in [Section 3.4].
Step 8: If the terminate criteria is satisfied, output the solutions in the Pareto set. Otherwise, go to Step 2.

The detailed genetic components for process planning and scheduling are given in [Section 3.2] and [Section 3.3].

3.2 Genetic Components for Process Planning

3.2.1 Encoding and Decoding Scheme

The aim of process planning in this research is to provide various near optimal process plans for the scheduling system. To deal with three different kinds of flexibilities in process planning effectively, each individual in process planning population contains of three parts with different length [Li. et al. 2013]. The first part of the individual is the feature sequence, which is the machining sequence of all features for one job. The second part is the selected operations sequence. The element in the *i*th position represents the selected candidate operations of the *i*th feature of this job. The third part is the selected machines sequence. The element in the *j*th position represents the selected candidate machines of the *j*th operation of this job. Figure 2 gives one feasible individual for job 1 in Table 1. In this individual, this job has 6 features and 12 operations. Therefore, the length of feature sequence and candidate operations sequence is 6, the length of candidate machines sequence is 12. From the feature sequence, it is clear that the machining sequence of the features for job 1 is F_1 - F_3 - F_4 - F_2 - F_6 - F_5. For the candidate operations sequence, the second element is 2, it means the second feature (F_2) chooses its second candidate operations (O_5 - O_6). For the candidate machines sequence, the first element is 3, it means the first operation (O_1) chooses its third candidate machines (M_3).

Based on the encoding scheme, this individual could be decoded easily. From the candidate operations sequence, the selected operations for each feature are F_1 (O_1), F_2 (O_5-O_6), F_3 (O_7), F_4 (O_{10}), F_5 (O_{11}), F_6 (O_{12}). From the candidate machine sequence, the selected machines for each selected operations are O_1 (M_3), O_5 (M_5), O_6 (M_3), O_7 (M_1), O_{10} (M_4), O_{11} (M_5), and O_{12} (M_5). So, the process plan determined by this individual is O_1 (M_3) - O_7 (M_1) - O_{10} (M_4) - O_5 (M_5) - O_6 (M_3) - O_{12} (M_5) - O_{11} (M_5).

Feature sequence	1	3	4	2	6	5						
Selected operations sequence	1	2	1	2	1	1						
Selected machine sequence	3	1	2	3	2	1	1	2	1	1	4	3

Figure 2: One Individual in Process Planning Population

3.2.2 Initial Population and Fitness Evaluation

Using the encoding scheme described above, the feature sequence is randomly arranged, and the candidate operations sequence, machines sequence for jobs is randomly determined from the corresponding candidates. Since there are precedence constraints among features, some feature sequences in the initial population may be infeasible. In this paper, the constraint adjustment method proposed by [Li et al. 02] is adopted to regulate the infeasible feature sequence into feasible one.

The processing time of one job is used as the fitness evaluation directly. The processing time is shorter, the individual is better. After the optimization of process planning, the total machine workload and maximal machine workload are determined.

3.2.3 Genetic Operators for Process Planning

Crossover operator: there are three parts in an individual in process planning population, so three crossover operators are developed for feature sequence, candidate operations sequence and candidate machines sequence. The crossover operator of feature sequence works as in Figure 3. First, select two individual P1 and P2 from the current population, initialize two empty offspring O1 and O2. Second, select two crossover points randomly to divide P1 and P2 into three parts. Third, append the element in the middle of P1 and P2 to the same positions in O1 and O2. At the end, delete the existing elements of O1 in P2, and then append the remaining elements of P2 to the rest positions in O1. O2 can be obtained by the same method. This crossover operator can maintain the precedence constraints among features, the new individuals obtained by this operator must be feasible solutions. The crossover operator of candidate operations sequence is shown in Figure 4. First, select two crossover points randomly, and then two offspring O1 and O2 are created by swapping divided middle parts of P1 and P2. The crossover operator of candidate machines sequence has the same procedure with the crossover operator of candidate operations as shown in Figure 5.

Mutation operator: For feature sequence in the individual, the mutation operation is selecting two positions at random and then swapping the elements in these positions. If the new sequence is infeasible, use the constraint adjustment method to regulate the infeasible sequence into feasible one. For candidate operations sequence and candidate machines sequence, the mutation operation is selecting a position randomly and then change the element of this selected position to another alternative operation or machine in the candidate operations or machines set.

Selection operator: The Tournament Selection is used as the selection operation. Select two individual from the population randomly, and then generate a random value between 0 and 1, if the value is less than a given probability, select a better individual, otherwise, select another one. In this paper, this probability is set as 0.8.

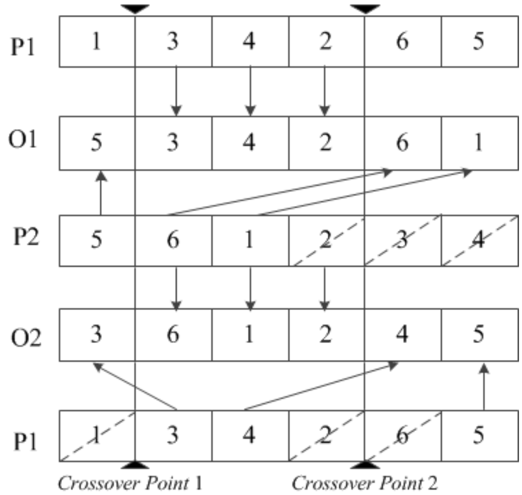

Figure 3: Crossover Operator for Feature Sequence

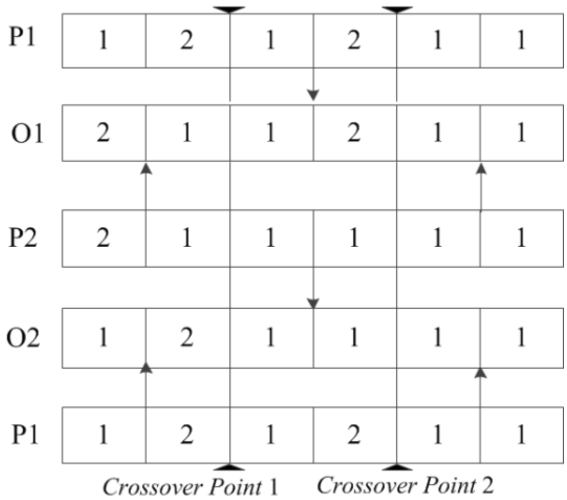

Figure 4: Crossover Operator for Selected Operations Sequence

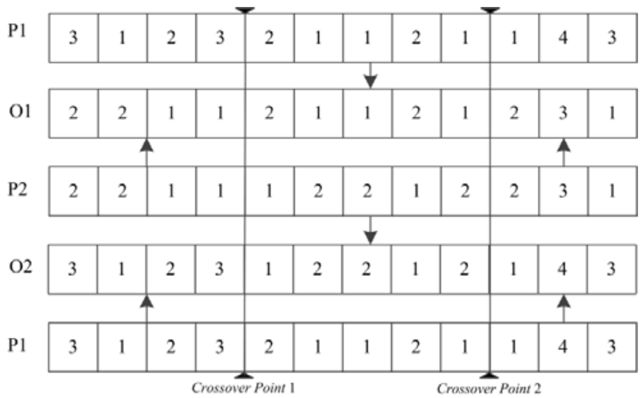

Figure 5: Crossover Operator for Selected Machines Sequence

3.3 Genetic Components for Scheduling

3.3.1 Encoding and Decoding

For each chromosome in the scheduling population, the operation-based encoding method is used as the encoding strategy. As the example described in Section 2, after the optimization of process planning, suppose that job 1 has 6 operations, job 2 has 4 operations and job 3 has 5 operations. One feasible solution in scheduling can be encoded as [1 1 2 3 2 1 1 2 3 2 1 1 2 3 3 3]. The second element in the chromosome is 1, 1 has been repeated twice, so this element represents the second operation of job 1. Each chromosome should be decoded into the active schedules in the decoding procedure [Zhang et al. 05].

3.3.2 Initial Population and Fitness Evaluation

After the optimization of process planning, the number of operations for each job is determined. Each individual in the populations is encoded randomly according to the results of process planning. The maximal machine workload and total workload of machines have been determined after process planning. Therefore, in the scheduling optimization process, makespan is used as the fitness evaluation criterion directly. The makespan can be obtained after decoding the individual into active schedule.

3.3.3 Genetic Operations for Scheduling

Crossover operation: The crossover operation is POX (Precedence Operation Crossover), which could be referred from [Zhang et al. 05]. The crossover operator works as in Figure 6. First, select two individual P1 and P2 from the current population, initialize two empty offspring O1 and O2. Second, 3 jobs are divided into two subsets. Job 2 is included in *JobSet* 1. Job 1 and Job 3 are included in *JobSet* 2. Third, append the elements in *JobSet* 1 of P1 to the same positions in O1. Append the elements in *JobSet* 1 of P2 to the same positions in O2. At the end, append the elements in *JobSet* 2 of P2 to the same positions in O1. Append the elements in *JobSet* 2 of P1 to the same positions in O2.

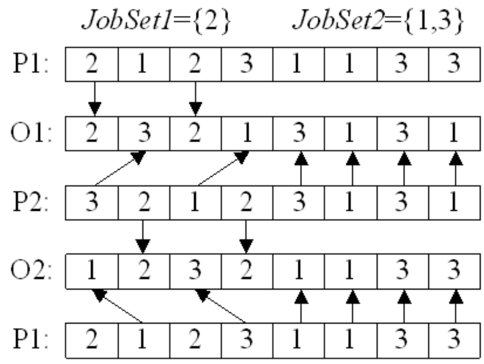

Figure 6: POX crossover operation

Mutation operator: The mutation operator works as in Figure 7. First, select an individual from the population as P randomly. Second, select a pair of different elements in P. At the end, O is obtained by swapping the selected elements.

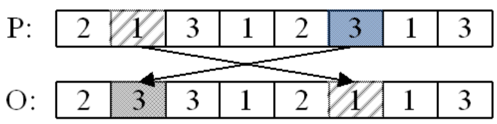

Figure 7: Mutation operations for scheduling

Selection operator: The selection operator for scheduling is the same with the selection operation for process planning.

3.4 Pareto Set Update Scheme

The result of multi-objective optimization problem is not a single solution; it is a Pareto optimal set. Pareto set is utilized to store and maintain the solutions obtained during the optimization procedure. The solutions in Pareto set are non-dominated with each other.

When there is a new solution obtained, the following operations will be applied to update the Pareto set: 1) if the new solution is dominated by any solution in the Pareto set, it will be discarded. 2) If there are solutions in the Pareto set dominated by the new solution, they will be removed from the Pareto set while the new solution will be added into the Pareto set. 3) If the new solution is non-dominated with all the solutions in the Pareto set and the Pareto set is not full, it will be added into the Pareto set. If the Pareto set is full at this time, remove the solution with the minimum crowded distance from the archive and then add the new solution into the archive. The crowded distance for each solution in the Pareto set could be computed by the method in NSGA-II [Deb et al. 02].

4 Experimental results and discussions

To evaluate the performance of the proposed algorithm, two different experiments have been selected in this paper. In order to compare with other algorithms, three different scale instances from literature were selected in Experiment 1. Due to the deficiency of benchmark instances on multi-objective IPPS with various flexibilities in process planning, we present Experiment 2 based on six typical parts with various flexibilities in process planning from previous literature.

The proposed algorithm in this paper was coded in C++ and implemented on a computer with a 2.0GHz Core(TM) 2 Duo CPU. The parameters of the proposed algorithm are selected after a lot of trials and shown in Table 2.

Parameter	Value	Parameter	Value
$PopSize_{PP}$	100	$PopSize_{S}$	200
$MaxGen_{PP}$	10	$MaxGen_{S}$	100
PP_c	0.80	SP_c	0.80
PP_m	0.10	SP_m	0.05
$MaxGen_{IPPS}$	100	*ParetoSet*	10

Table 2: Parameters of the Proposed Algorithm

4.1 Experiment 1

There are three different scale instances in Experiment 1. The first instance obtained from [Baykasoğlu and Özbakır 09] has 5 jobs and 5 machines with 20 operations. The second instance obtained from [Rajkumar et al. 10] has 8 jobs and 8 machines with 37 operations. The third instance has 20 jobs and 5 machine with 80 operations which is also obtained from [Rajkumar et al. 10]. The detailed part data of the three problem instances can be referred from [Baykasoğlu and Özbakır 09] and [Rajkumar et al. 10].

The comparisons among grammatical approach, GRASP and proposed algorithm for the first instance are shown in Table 3. The comparisons between GRASP and proposed algorithm for the second and third instances are shown in Table 4 and Table 5 respectively. It is clearly that the proposed algorithm could obtain several Pareto optimal solutions instead of a single solution obtained by the methods in literature.

The results of proposed algorithm are obtained by running the algorithm 20 times. The results of grammatical approach are obtained from [Baykasoğlu and Özbakır 09]. And the results of GRASP are obtained from [Rajkumar et al. 10]. From the comparisons in Table 3, Table 4 and Table 5, all the Pareto optimal solutions obtained by the proposed algorithm could dominate the solutions obtained by grammatical approach and GRASP algorithm. The detailed process plans of the second Pareto optimal solution for the first, second and third instance are given in Table 6, Table 7 and Table 8 respectively. The correspond Gantt charts are shown in Figure 8, Figure 9 and Figure 10 respectively.

Algorithm	Pareto optimal solutions		
	Makespan	**MMW**	**TWM**
Grammatical approach	394	328	770
GRASP algorithm	242	217	750
Proposed algorithm	***212***	***188***	***721***
	198	***193***	***722***
	207	***187***	***737***
	238	***172***	***730***
	210	***182***	***735***
	211	***199***	***718***
	191	***172***	***745***
	218	***187***	***731***
	233	***197***	***719***
	226	***181***	***739***

Table 3: Comparisons among three algorithms for the first instance

Algorithm	Pareto optimal solutions		
	Makespan	**MMW**	**TWM**
GRASP algorithm	253	237	1189
Proposed algorithm	***234***	***211***	***1146***
	214	***199***	***1163***
	218	***200***	***1159***
	236	***181***	***1164***
	233	***187***	***1153***
	213	***207***	***1149***
	236	***200***	***1142***
	251	***189***	***1137***
	228	***221***	***1139***
	236	***203***	***1135***

Table 4: Comparisons between two algorithms for the second instance

Algorithm	Pareto optimal solutions		
	Makespan	**MMW**	**TWM**
GRASP algorithm	924	889	2963
Proposed algorithm	***806***	***806***	***2836***
	708	***708***	***2960***
	747	***747***	***2923***
	784	***784***	***2856***
	871	***871***	***2835***
	883	***883***	***2830***

Table 5: Comparisons between two algorithms for the third instance

Jobs	Detailed process plans		Jobs	Detailed process plans	
	Operation Sequence	**Machine Sequence**		**Operation Sequence**	**Machine Sequence**
Job1	5-4-2-3	1-4-2-5	Job4	4-3-5-2	4-5-2-4
Job2	1-2-4	3-2-5	Job5	4-3-1	4-5-3
Job3	3-5-1-4	2-2-3-4			

Table 6: Process plans of the Second Pareto Optimal Solution for the First Instance

Jobs	Detailed process plans		Jobs	Detailed process plans	
	Operation Sequence	**Machine Sequence**		**Operation Sequence**	**Machine Sequence**
Job1	4-5-2-3	4-6-2-2	Job5	1-4-3	3-4-5
Job2	1-2-4	6-2-7	Job6	5-3-1-4	5-8-1-4
Job3	1-5-3-4	3-3-8-1	Job7	4-3-5-2	5-4-2-2
Job4	5-3-2-4	2-8-1-5	Job8	4-1-3	7-6-5

Table 7: Process Plans of the Second Pareto Optimal Solution for the Second Instance

Jobs	Detailed process plans		Jobs	Detailed process plans	
	Operation Sequence	**Machine Sequence**		**Operation Sequence**	**Machine Sequence**
Job1	4-5-2-3	4-2-2-1	Job11	3-5-2-4	1-3-2-4
Job2	1-4-2	3-4-2	Job12	2-1-4	2-3-5
Job3	1-5-3-4	3-5-2-5	Job13	5-3-1-4	5-2-1-4
Job4	3-2-5-4	5-4-2-4	Job14	3-2-5-4	5-5-2-4
Job5	4-1-3	4-3-5	Job15	4-3-1	4-5-3
Job6	3-5-2-4	1-2-2-4	Job16	4-5-2-3	4-2-2-4
Job7	4-2-1	5-2-3	Job17	2-1-4	2-3-5
Job8	3-5-1-4	2-2-3-2	Job18	5-3-1-4	5-2-1-3
Job9	3-2-5-4	5-2-4-5	Job19	5-3-2-4	2-5-4-4
Job10	4-1-3	2-3-3	Job20	4-1-3	1-3-5

Table 8: Process Plans of the Second Pareto Optimal Solution for the Third Instance

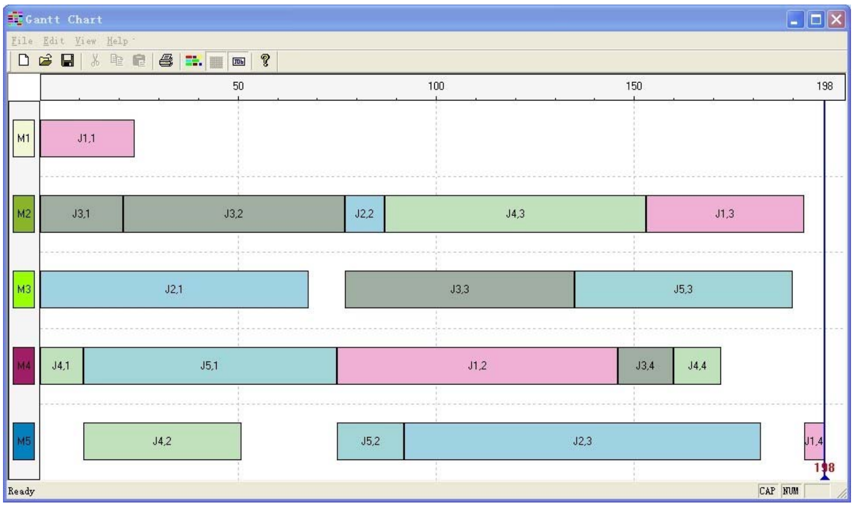

Figure 8: Gantt chart of the second Pareto optimal solution for the first instance

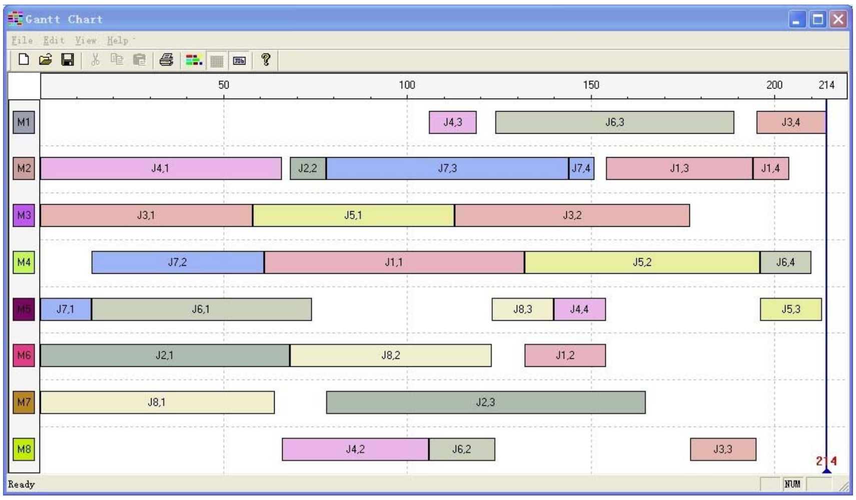

Figure 9: Gantt chart of the second Pareto optimal solution for the second instance

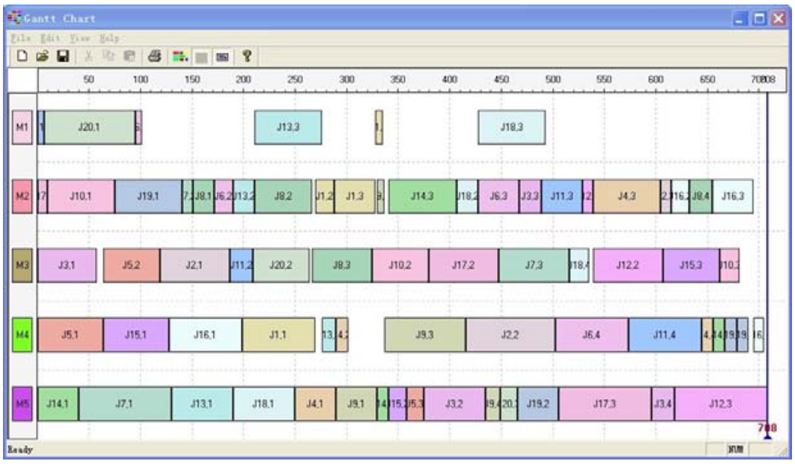

Figure 10: Gantt chart of the second Pareto optimal solution for the third instance

4.2 Experiment 2

Due to the lack of benchmark instances of multi-objective IPPS problem with various flexibilities in process planning, we design a problem based on 6 typical jobs selected from the previous literature. Job 1, Job 2 and Job 3 are acquired from [Li and McMahon 07]. The detailed processing information of Job 1, Job 2 and Job 3 used in this experiment is given in Table 9, Table 10 and Table 11 respectively. Job 4 is obtained from [Ma et al. 00] which contains 9 features and 13 operations. Job 5 is obtained from [Wang et al. 09] which contains 7 features and 9operations. Job 6 is acquired from [Zhang and Nee 01] which contains 14 features and 16 operations. The detailed machining information of Job 4, Job 5 and Job6 can refer to [Li et al. 13]. Suppose that these 6 jobs are machined in 5 machines in workshop.

Features	Candidate Operations	Candidate machines	Processing time	Precedence Constraints
F_1	O_1	M_2, M_3, M_4	40, 40, 30	Before all features
F_2	O_2	M_2, M_3, M_4	40, 40, 30	Before F_{10}, F_{11}
F_3	O_3	M_2, M_3, M_4	20, 20, 15	
F_4	O_4	M_1, M_2, M_3, M_4	12, 10, 10, 8	
F_5	O_5	M_2, M_3, M_4	35, 35, 27	Before F_4, F_7
F_6	O_6	M_2, M_3, M_4	15, 15, 12	Before F_{10}
F_7	O_7	M_2, M_3, M_4	30, 30, 23	Before F_8
F_8	O_8-O_9-O_{10}	M_1, M_2, M_3, M_4	22, 18, 18, 14	
		M_2, M_3, M_4	10, 10, 8	
		M_2, M_3, M_4, M_5	10, 10, 8, 12	
F_9	O_{11}	M_2, M_3, M_4	15, 15, 12	Before F_{10}
F_{10}	O_{12}-O_{13}-O_{14}	M_1, M_2, M_3, M_4	48, 40, 40, 30	Before F_{11}, F_{14}
		M_2, M_3, M_4	25, 25, 19	
		M_2, M_3, M_4, M_5	25, 25, 19, 30	
F_{11}	O_{15}-O_{16}	M_1, M_2, M_3, M_4	27, 22, 22, 17	
		M_2, M_3, M_4	20, 20, 15	
F_{12}	O_{17}	M_2, M_3, M_4	16, 16,12	
F_{13}	O_{18}	M_2, M_3, M_4	35, 35, 27	Before F_4, F_{12}
F_{14}	O_{19}-O_{20}	M_2, M_3, M_4	12, 12, 9	
		M_2, M_3, M_4, M_5	12, 12, 9, 15	

Table 9: Processing information of Job 1 with 14 features and 20 operations

Table 12 shows the Pareto optimal solutions obtained by the proposed algorithm for Experiment 2 by running the algorithm 20 times. The detailed process plans of the second and last Pareto optimal solution for Experiment 2 are given in Table 13 and Table 14 respectively. The correspond Gantt charts are shown in Figure 11 and Figure 12 respectively.

Features	**Candidate operations**	**Candidate machines**	**Processing time**	**Precedence constrains**
F_1	O_1	M_1 M_2 M_3 M_4	12,10,10,8	Before F_2
F_2	O_2	M_2, M_3 M_4	20,20,15	
F_3	O_3	M_2, M_3 M_4	18,18,14	Before F_4
F_4	O_4	M_2, M_3 M_4	16,16,12	
F_5	O_5	M_2, M_3 M_4	15,15,11	
F_6	O_6- O_7	M_1 M_2 M_3 M_4	30,25,25,19	Before F_7
		M_2, M_3 M_4	25,25,19	
F_7	O_8	M_1 M_2 M_3 M_4	14,12,12,9	
F_8	O_9	M_2, M_3 M_4	15,15,11	Before F_7
F_9	O_{10}	M_1 M_2 M_3 M_4	10,8,8,6	
F_{10}	O_{11}	M_2, M_3 M_4	10,10,8	Before F_{11}
F_{11}	O_{12}	M_2, M_3 M_4	10,10,8	Before F_9
F_{12}	O_{13}	M_1 M_2 M_3 M_4	10,8,8,6	
F_{13}	O_{14}	M_2, M_3 M_4	16,16,12	Before F_{14}
F_{14}	O_{15}	M_1 M_2 M_3 M_4	10,8,8,6	
F_{15}	O_{16}	M_1 M_2 M_3 M_4	36,30,30,23	Before all features

Table 10: Processing information of Job 2 with 15 features and 16 operations

Features	**Candidate operations**	**Candidate machines**	**Processing time**	**Precedence constrains**
F_1	O_1	M_2, M_3, M_4	20, 20, 15	Before all features
F_2	O_2	M_2, M_3, M_4	20, 20, 15	Before F_3 - F_{11}
F_3	O_3	M_2, M_3, M_4	15,15,11	Before F_{10}, F_{11}
F_4	O_4	M_1, M_2, M_3, M_4	15,15,11,18	Before F_{10}, F_{11}
F_5	O_5	M_2, M_3, M_4	15,15,11	Before F_{10}, F_{11}
F_6	O_6	M_2, M_3, M_4	15, 15,11	Before F_{10}, F_{11}
F_7	O_7	M_2, M_3, M_4	15,15,11	
F_8	O_8	M_2, M_3, M_4	25,25,19	
F_9	O_9-O_{10}-O_{11}	M_1, M_2, M_3, M_4	30,25,25,19	Before F_7 ,F_8
		M_2, M_3, M_4	20,20,15	
		M_2, M_3, M_4 M_5	20,20,15,24	
F_{10}	O_{12}-O_{13}	M_1, M_2, M_3, M_4	10,8,8,6	
		M_2, M_3, M_4	8,8,6	
F_{11}	O_{14}	M_1, M_2, M_3, M_4	6,5,5,4	

Table 11: Processing information of Job 3 with 11 features and 14 operations

Algorithm	Pareto optimal solutions		
	Makespan	MMW	TWM
Proposed algorithm	617	617	1522
	599	599	1528
	540	520	1534
	520	511	1541
	511	511	1551
	494	484	1552
	494	469	1562
	502	453	1568
	460	439	1569
	459	432	1579

Table 12: Pareto Optimal Solutions Obtained by Proposed Algorithm for Experiment 2

Jobs	Detailed process plans	
	Operation Sequence	Machine Sequence
Job 1	1-11-5-7-6-18-4-3-2-12-13 -14-19-20-8-9-10-15-16-17	4-4-4-4-2-4-2-4-4-4-4 -4-2-4-1-4-4-3-4-4
Job 2	16-5-13-9-14-6-7-11 -3-15-12-8-1-2-4-10	4-4-4-4-4-4-4-2 -4-4-3-2-2-4-4-2
Job 3	1-2-5-3-6-9-10 -11-8-4-14-7-12-13	2-4-3-4-2-4-4 -4-4-1-4-2-4-2
Job 4	4-5-6-13-2-3-10 -11-1-7-8-9-12	1-1-2-1-2-1-1 -2-5-5-1-2-1
Job 5	1-3-5-6-7-8-4-2-9	2-2-3-5-5-4-5-4-5
Job 6	2-12-3-14-1-5-6-7-11 -18-15-13-4-16-17-8-9-10	4-1-1-1-2-2-5-1-1 -2-3-5-1-1-1-1-1-2

Table 13: Process plans for the second Pareto optimal solution in Experiment 2

Jobs	Detailed process plans	
	Operation Sequence	**Machine Sequence**
Job 1	1-3-18-6-17-11-5-7-4-2-12-13-14-15-16-8-9-10-19-20	4-3-2-3-3-2-4-3-2-4-4-2-2-2-4-2-3-4-4-4
Job 2	16-6-7-14-1-2-3-13-9-8-5-15-11-12-10-4	4-4-3-4-2-4-4-3-2-4-4-4-3-3-4-3-2
Job 3	1-2-5-6-9-10-11-7-8-3-4-12-13-14	4-4-4-4-4-2-2-4-4-4-3-2-4-1
Job 4	2-4-13-1-7-8-9-3-5-6-12-10-11	2-1-4-1-2-1-5-1-1-3-1-1-1
Job 5	1-5-6-4-2-7-8-9-3	5-2-1-5-4-2-1-1-2
Job 6	14-15-8-9-10-13-16-3-4-12-18-17-2-5-6-7-11-1	1-3-5-3-2-1-1-2-1-2-1-1-5-3-5-2-1-1

Table 14: Process plans for the last Pareto optimal solution in Experiment 2

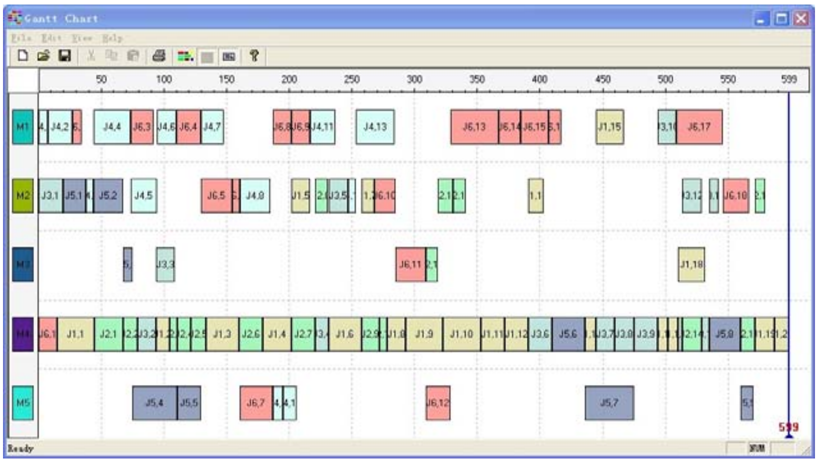

Figure 11: Gantt chart of the second Pareto optimal solution for Experiment 2

4.3 Discussions

From all of above experimental results, it is clear that three objectives of IPPS problem considered in this paper are conflicting. In Experiment 2, Table 9, Table 10 and Table 11 show that almost all the operations in job 1, job 2 and job 3 have a shorter processing time in machine 4. If all the operations supposed to be machined in machine 4, TWM will be smaller. In this case, makespan will be longer. For example, the second Pareto optimal solution has a shorter MMW compared with the last Pareto optimal solution in Experiment 2. It is obvious that more operations need to be machined in machine 4 in the second solution compared with the last solution from Figure 11 and Figure 12. More operations are waiting to be processed in machine 4. But the other machines have a lot of free time in Figure 11. As a result, the makespan of the second solution is longer than the last solution. The proposed algorithm in this paper could optimize these conflicting objectives simultaneously and help decision makers to make a trade-off among these objectives while determining a final schedule.

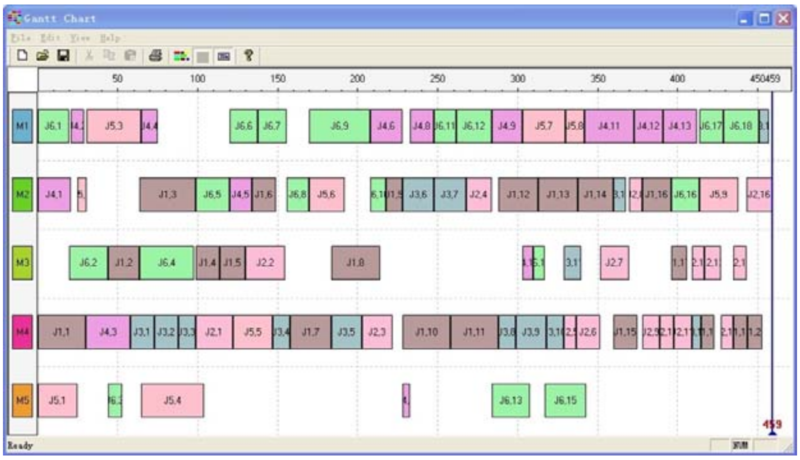

Figure 12: Gantt chart of the last Pareto optimal solution for Experiment 2

Based on results of Experiment 1, the proposed algorithm could obtain more and better Pareto optimal solutions compared with grammatical approach and GRASP algorithm. It shows that the proposed algorithm has achieved satisfactory improvement compared with previous research works.

The problem presented in Experiment 2 considers various flexibilities in process planning simultaneously during the whole optimization procedure. Each job has many different process plans according to the processing information. As a result, this problem is much more complex than instances in Experiment 1. On the other hand, this problem is much closer to realistic production process compared with Experiment 1. The proposed algorithm can also obtain good solutions of Experiment 2 effectively.

The reasons of the proposed algorithm's superior performance in solving multi-objective IPPS problem are as follows:

Firstly, effective genetic operations based on the characteristics of IPPS are employed in the proposed algorithm. This can make the proposed algorithm suitable for solving multi-objective IPPS problem.

Secondly, from the framework of the proposed algorithm, process planning system provides many different process plans of jobs dynamically based on various flexibilities in process planning for scheduling system, which ensure that the algorithm explore IPPS solution space fully.

Finally, the Pareto set could store and maintain the solutions obtained during the searching procedure, the proposed algorithm could get several Pareto optimal solutions during one searching process.

5 Conclusion and Future Works

This paper has presented an effective genetic algorithm for solving multi-objective IPPS problem with various flexibilities in process planning. Makespan, maximal machine workload and total workload of machines are considered as optimization objectives simultaneously. To compare with the other algorithms, three different scale instances have been employed to test the performance of the proposed algorithm. The experiment results show that the proposed algorithm has achieved satisfactory improvement. Due to the lack of benchmark instances of multi-objective IPPS problem with various flexibilities in process planning, a problem was presented based on six typical jobs with various flexibilities in process planning in literature. The proposed algorithm could also settle this problem effectively.

There are also some limitations in the proposed algorithm. Only three objectives are optimized in this study, more objectives of IPPS can be taken into account in future works. Exploring more effective algorithms to solve multi-objective IPPS problem is another future work.

Acknowledgements

This research work is supported by the National Natural Science Foundation of China (NSFC) under Grant no. 51375004 and 51435009.

References

[Baykasoğlu et al. 04] Baykasoğlu, A., Özbakir, L., Sönmez, A.: "Using Multiple Objective Tabu Search and Grammars to Model and Solve Multi-objective Flexible Job Shop Scheduling Problems"; Journal of Intelligent Manufacturing, 15 (2004), 777-785.

[Baykasoğlu and Özbakır 09] Baykasoğlu, A., Özbakır, L.: "A Grammatical Optimization Approach for Integrated Process Planning and Scheduling"; Journal of Intelligent Manufacturing, 20 (2009), 211-221.

[Deb et al. 02] Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II"; IEEE Transactions on Evolutionary Computation, 6 (2002), 182-197.

[Fang et al. 93] Fang, H. L., Ross, P., Corne, D.: "A Promising Genetic Algorithm Approach to Job-Shop Scheduling, Rescheduling, and Open-Shop Scheduling Problems"; Proceedings of the Fifth International Conference on Genetic Algorithms, San Mateo, California (1993), 375-382.

[Guo et al. 09] Guo, Y. W., Li, W. D., Mileham, A. R., Owen, G. W.: "Applications of Particle Swarm Optimisation in Integrated Process Planning and Scheduling"; Robotics and Computer-Integrated Manufacturing, 25 (2009), 280-288.

[Jung et al. 01] Jung, H. H., Inho, H., Eunseok, L., Juneho, Y.: "Manufacturing Feature Recognition Toward Integration with Process Planning"; IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 31 (2001), 373-380.

[Kumar et al. 03] Kumar, M., Rajotia, S.: "Integration of Scheduling with Computer Aided Process Planning"; Journal of Materials Processing Technology, 138 (2003), 297-300.

[Li et al. 12] Li, X. Y., Gao, L., Li, W. D.: "Application of Game Theory based Hybrid Algorithm for Multi-objective Integrated Process Planning and Scheduling"; Expert Systems with Applications, 39 (2012), 288-297.

[Li et al. 02] Li, W. D., Ong, S. K., Nee, A. Y. C.: "Hybrid Genetic Algorithm and Simulated Annealing Approach for the Optimization of Process Plans for Prismatic Parts"; International Journal of Production Research, 40 (2002), 1899-1922.

[Li and McMahon 07] Li, W. D., McMahon, C. A.: "A Simulated Annealing-based Optimization Approach for Integrated Process Planning and Scheduling"; International Journal of Computer Integrated Manufacturing, 20 (2007), 80-95.

[Li et al. 10] Li, X. Y., Gao, L., Zhang, C. Y., Shao, X. Y.: "A Review on Integrated Process Planning and Scheduling"; International Journal of Manufacturing Research, 5 (2010), 161-180.

[Li et al. 13] Li, X. Y., Gao, L., Wen, X. Y.: "Application of an Efficient Modified Particle Swarm Optimization Algorithm for Process Planning"; International Journal of Advanced Manufacturing Technology, 67 (2013), 1355-1369.

[Lian et al. 12] Lian, K. L., Zhang, C. Y., Gao, L., Li, X. Y.: "Integrated Process Planning and Scheduling Using an Imperialist Competitive Algorithm"; International Journal of Production Research, 50 (2012), 4326–4343.

[Lv et al. 14] Lv, S. P., Qiao, L. H.: "Process Planning and Scheduling Integration With Optimal Rescheduling Strategies". International Journal of Computer Integrated Manufacturing, 27 (2014), 638-655.

[Ma et al. 00] Ma, G. H., Zhang, Y. F., Nee, A. Y. C.: "A Simulated Annealing-based Optimization Algorithm for Process Planning"; International Journal of Production Research, 38 (2000), 2671-2687.

[Mohammadi et al. 11] Mohammadi, G., Karampourhaghghi, A., Samaei, F.: "A Multi-objective Optimisation Model to Integrating Flexible Process Planning and Scheduling based on Hybrid Multi-objective Simulated Annealing"; International Journal of Production Research, 50 (2011), 5063-5076.

[Mohapatra et al. 13] Mohapatra, P., Benyoucef, L., Tiwari, M. K.: "Integration of Process Planning and Scheduling Through Adaptive Setup Planning: a Multi-objective Approach"; International Journal of Production Research, 51 (2013), 7190-7208.

[Marler et al. 04] Marler, R. T., Arora, J. S.: "Survey of Multi-objective Optimization Methods for Engineering"; Structural and Multidisciplinary Optimization, 26 (2004), 369-395.

[Morad and Zalala 99] Morad, N., Zalala, A.: "Genetic Algorithms in Integrated Process Planning and Scheduling"; Journal of Intelligent Manufacturing, 10 (1999), 169-179.

[Phanden et al. 11] Phanden, R. K., Jain, A., Verma, R.: "Integration of Process Planning and Scheduling: a State-of-the-art Review"; International Journal of Computer Integrated Manufacturing, 24 (2011), 517-534.

[Rajkumar et al. 10] Rajkumar, M., Asokan, P., Page, T., Arunachalam, S.: "A GRASP Algorithm for the Integration of Process Planning and Scheduling in a Flexible Job-shop"; International Journal of Manufacturing Research, 5 (2010), 230-251.

[Wang et al. 09] Wang, Y. F., Zhang, Y. F., Fuh, J. Y. H.: "Using Hybrid Particle Swarm Optimization for Process Planning Problem"; Proceedings of the Computational Sciences and Optimization, Sanya (2009), 304-308.

[Wang et al. 10] Wang, Y. F., Zhang, Y. F., Fuh, J. Y. H.: "A PSO-based Multi-objective Optimization Approach to the Integration of Process Planning and Scheduling"; Proceedings of the 2010 8th IEEE International Conference on Control and Automation (ICCA), Xiamen (2010), 614-619.

[Wang et al. 06] Wang, L., Shen, W., Hao, Q.: "An Overview of Distributed Process Planning and Its Integration with Scheduling"; International Journal of Computer Applications in Technology, 26 (2006), 3-14.

[Wong et al. 12] Wong, T. N., Zhang, S., Wang, G., Zhang, L. P.: "Integrated Process Planning and Scheduling – Multi-agent System with Two-stage Ant Colony Optimization Algorithm"; International Journal of Production Research, 50 (2012), 6188-6201.

[Zhang et al. 05] Zhang, C., Li, P., Rao, Y., Li, S.: "A New Hybrid GA/SA Algorithm for the Job Shop Scheduling Problem"; Lecture Notes in Computer Science, 3448 (2005), 246-259.

[Zhang and Nee 01] Zhang, Y. F., Nee, A. Y. C.: "Applications of Genetic Algorithms and Simulated Annealing in Process Planning Optimization"; Computational Intelligence in Manufacturing Handbook, Wang and Kusiak (Eds), CRC Press, Boca Raton, (2001).

[Zhang and Fujimura 10] Zhang, W., Fujimura, S.: "Improved Vector Evaluated Genetic Algorithm with Archive for Solving Multiobjective PPS Problem"; Proceeding of the 2010 International Conference on E-Product E-Service and E-Entertainment (ICEEE), Henan (2010), 1-4.