Int j simul model 15 (2016) 1, 157-169
ISSN 1726-4529 Original scientific paper

MODELLING AND SIMULATION OF A MULTI-RESOURCE FLEXIBLE JOB-SHOP SCHEDULING

Ba, L.; Li, Y.; Yang, M. S.; Gao, X. Q. & Liu, Y.
School of Mechanical and Precision Instrument Engineering, Xi'an University of Technology, Xi'an, 710048, Shanxi, China
E-Mail: xautbali@163.com, yangmingshun@xaut.edu.cn

Abstract
Flexible job-shop scheduling problem (FJSP) in the field of production scheduling presents a quite difficult combinatorial optimization problem. Machines are mostly considered to be the only resource in many research projects dealing with FJSP. In actual production, there are many other factors which influence production scheduling, such as transportation, storage and detection. If machines are considered to be the only resource, the problem may not be in accord with the actual production. Thus, in order to make FJSP more in line with the real production situation, machines, warehouses, vehicles and detection equipment are all considered to be the scheduled resources simultaneously due to the shortage of flexible job shop scheduling problem in resources. A new mathematical model for a multi-resource flexible job-shop scheduling problem (MRFJSP) is proposed. The constraints of the model are presented. The makespan is the main target which will be minimized. A genetic algorithm which includes elitist strategy is proposed to solve the MRFJSP. Due to the complexity of MRFJSP, each key module of the genetic algorithm is redesigned. Finally, the model and algorithm are proved through an application case.
(Received, processed and accepted by the Chinese Representative Office.)

Key Words: Multi-Resource, Scheduling, Genetic Algorithm

1. INTRODUCTION

Job-shop scheduling problem (JSP), a subject of intense study in recent research in the field of production scheduling, is known to be NP-Hard [1]. Flexible job-shop scheduling problem (FJSP), proposed by Brucker and Schlie in 1990 [2], is an extension of JSP that eliminates the one-to-one correspondence relationship between process and machine. Therefore, compared to JSP, FJSP is more complex and has been proved to be NP-Hard [3]. Since FJSP better fits the actual production environment, research on FJSP is of significant theoretical and practical value [4].

In recent years, many research projects have mainly focus on the improvement of objective and algorithm in FJSP. As for the scheduled resources, much research concentrates only on machines. Machines are the most important resource which should be considered primarily in FJSP, but in the actual production environment, other resources also need to be scheduled, such as warehouses, vehicles, etc. These resources will also affect the production process and cannot be ignored.

Therefore, in order to make considerations of FJSP more in line with practical production conditions, this paper considers four various kinds of resources simultaneously. The scheduled resources include machines, warehouses, vehicles and detection equipment. A multi-resource flexible job-shop scheduling problem (MRFJSP) is proposed in this paper. The makespan is the objective which will be minimized, a mathematical model for MRFJSP is created and an elitist selection genetic algorithm is designed to solve the problem. Finally, an instance of verification is proposed for the model and algorithm.

This paper initially provides brief literature related to FJSP in Section 2. Section 3 presents a description for MRFJSP. A model for MRFJSP is proposed in Section 4. The design for an elitist selection genetic algorithm is presented in Section 5. Computational results are discussed in Section 6. The paper ends with some conclusions in Section7.

2. LITERATURE REVIEW

The flexible job-shop scheduling problem can be defined as follows [5]. Assume there are n workpieces $J=\{J_1, J_2, \ldots, J_n\}$ that need to be processed on m machines $M=\{M_1, M_2, \ldots, M_m\}$, each process of each workpiece can be operated on several available machines with various capabilities. In this situation, the most appropriate machine for each process is selected in order to minimize makespan.

Recently, much research has been done on FJSP from various aspects.

2.1 Flexible Job-shop Scheduling Problem (FJSP)

Karimi et al. [5] set makespan as an objective, proposing a knowledge-based variable neighbourhood search (KBVNS) to solve FJSP. Gutierrez and Garcia-Magarino [6] set makespan as an objective, proposing a hybrid genetic algorithm based on genetic algorithm (GA) and repair heuristics to solve FJSP. Li et al. [7] set makespan as an objective, proposing a hybrid bee colony algorithm based on a simulated annealing algorithm (SAA) and an artificial bee colony (ABC) for solving FJSP, with the algorithm being proved through some benchmarks. He and Zeng [8] set makespan as an objective, designing a Bayesian statistical inference-based estimation of distribution algorithm and adopting some benchmarks to prove the algorithm. Yazdani et al. [9] proposed a parallel variable neighbourhood search (PVNS) to solve FJSP in an effort to minimize makespan.

2.2 Multi-objective Flexible Job-shop Scheduling Problem (MOFJSP)

Wang and Feng [4] set makespan and cost as well as penalties for earliness and tardiness (E/T) as objectives, establishing a model for MOFJSP and proposed a multi-objective particle swarm optimization algorithm (PSO) for solving MOFJSP. Zhang et al. [10] took production capacity into constraints, setting makespan and maximum machine workload as objectives, and proposed a model for MOFJSP and designed an improved GA. The model and algorithm were proved through some benchmarks. Moslehi and Mahnam [11] set makespan, total workload, as well as maximum machine workload as objectives, proposing an intelligent algorithm based on (PSO) and local search (LS). Peng et al. [12] proposed a cloud model evolutionary MOFJSP based on improved non-dominated sorting for solving MOFJSP. Chen and Xing [13] proposed a bi-layer structured intelligent algorithm based on GA and ant colony optimization (ACO). The GA is used to solve the machine assignment problem, and the ACO is used to solve the operation sequencing problem. Shahsavari-Pour and Ghasemishabankareh [14] set different weights for different objectives, combining multiple objectives into a single objective, and then proposed a hybrid algorithm based on GA and SAA. Shi et al. [15] took total workload, penalties for earliness and tardiness, product qualification rate and cost into MOFJSP, establishing a model for MOFJSP and proposed a hybrid algorithm based on tabu search (TS) and ACO. Liu et al. [16] obtained the customer satisfaction through fuzzy due date and set makespan and customer satisfaction as objectives, establishing a model for MOFJSP and proposed a multi-population GA for solving the problem.

2.3 Flexible Job-shop Scheduling Problem with consideration of other factors

Aiming at FJSP with uncertain processing times, Palacios et al. [17] set fuzzy makespan as an objective and proposed a genetic tabu search based on GA, TS and heuristic seeding. Finally, the proposed algorithm was compared with several intelligent algorithms through some test instances. Xu et al. [18] set maximum fuzzy completion time as an objective and proposed a teaching-learning-based optimization algorithm for solving FJSP with uncertain processing times, which was then compared with several intelligent algorithms through some test instances. Özgüven and Yavuz [19] proposed two mixed-integer goal programming models for FJSP with separable and non-separable sequence-dependent setup times. The models were proved through some sample tests.

The literature related to FJSP mainly focus on MOFJSP and improvements of algorithms which only consider machine as the scheduled resource, while in the actual production environment, makespan is affected by various factors. Due to limited resources, some processes of workpieces may not able to be processed in only one enterprise. Instead, workpieces need to be transported to other related manufactures for processing, and because manufactures are usually in different geographic locations, the workpieces have to be transported to various locations. Due to the actual production situation changing in real time, the related machines may still be in working condition when the transported workpieces have arrived. Therefore, the workpieces will likely be transported to a warehouse nearby. For some workpieces, inspection is required after the completion of a certain process. These factors also affect makespan, and they should not be ignored in the actual processing situation.

Therefore, in this paper, machines, warehouses, vehicles and detection equipment are considered simultaneously in FJSP, with minimal makespan as the objective. A MRFJSP model is established and an elitist selection genetic algorithm is designed correspondently. Some computational results are discussed, and some conclusions are obtained at the end of this paper. Details are shown in Sections 3 to 7.

3. PROBLEM DESCRIPTION

Enterprise X has an order which requires processing a variety of workpieces. Due to the limited resources of enterprise X, there are several continuous processes of some workpieces that need to be processed on some machines belonging to some external enterprises. Each process of each workpiece can be processed on several available machines. Simultaneously, some warehouses, vehicles and detection equipment used for storage, transportation and detection. The problem is choosing the appropriate resources for each process of each workpiece in order to minimize the makespan and obtain the optimal solution.

In order to establish the model for MRFJSP, the following parameters are introduced:

n: The number of workpieces;

$p_{(i)}$: The number of processes of a workpiece i, $i = 1, 2, \ldots, n$;

m: The number of machines;

q: The number of enterprises which provide machines;

x: The number of warehouses;

y: The number of vehicles;

z: The number of pieces of detection equipment;

$T_{M(i,v)}$: The processing time of process v of workpiece i, $v = 1, 2, \ldots, p_{(i)}$;

$T_{S(i,v)}$: The storage time of process v of workpiece i;

$T_{Y(i,v)}$: For workpiece i, when $v = 1$, $T_{Y(i,v)}$ refers to the transport time between enterprise X and the enterprise which processes the first process of workpiece i; when $1 < v \leq p_{(i)}$, $T_{Y(i,v)}$ refers to the transport time between process $v - 1$ and process v; when $v = p_{(i)} + 1$, $T_{Y(i,v)}$ refers to the transport time between the enterprise which processes the last process of workpiece i and enterprise X;

$T_{YW(i,v)}$: The full car wait time of process v of workpiece i;

$T_{C(i,v)}$: The detection time of process v of workpiece i;

$T_{GM(i,v,j)}$: The processing time on machine j of process v of workpiece i, $j = 1, 2, \ldots, m$;
$T_{GS(i,v,w)}$: The storage time in warehouse w of process v of workpiece i, $w = 1, 2, \ldots, x$;
$T_{GY(i,v,r)}$: The transport time on vehicle r of process v of workpiece i, $r = 1, 2, \ldots, y$;
$T_{GYW(i,v,r)}$: The wait time of a full car on vehicle r of process v of workpiece i;
$T_{GC(i,v,d)}$: The detection time of detection equipment d of process v of workpiece i, $d = 1, 2, \ldots, z$;
$T_{EN(a,b,r)}$: The transport time on vehicle r between enterprise a and enterprise b, $a, b \in [1, q]$;
$T_{GST(i,v)}$: The starting time of process v of workpiece i;
$T_{MS(j)}$: The starting time of machine j;
$T_{AM(i)}$: The processing time of workpiece i;
$T_{AS(i)}$: The storage time of workpiece i;
$T_{AY(i)}$: The transport time of workpiece i;
$T_{AYW(i)}$: The wait time of a full car of workpiece i;
$T_{AC(i)}$: The detection time of workpiece i;
$R_{M(i,v,j)}$: Boolean variable, if process v of workpiece i is processed on machine j, the value is 1, otherwise it is 0;
$R_{S(i,v,w)}$: Boolean variable, if process v of workpiece i is stored in warehouse w after it is finished, the value is 1, otherwise it is 0;
$R_{Y(i,v,r)}$: Boolean variable, if process v of workpiece i is transported by vehicle r after it is finished, the value is 1, otherwise it is 0;
$R_{C(i,v,d)}$: Boolean variable, if process v of workpiece i is inspected by detection equipment d after it is finished, the value is 1, otherwise it is 0;
$R_{E(i,v,a)}$: Boolean variable, if process v of workpiece i is processed in enterprise a, the value is 1, otherwise it is 0;
$R_{ME(j,a)}$: Boolean variable, if machine j belongs to enterprise a, the value is 1, otherwise it is 0;
$R'_{SE(w,a)}$: Boolean variable, if warehouse w is near enterprise a, the value is 1, otherwise it is 0;
$R'_{YE(r,a)}$: Boolean variable, if vehicle r is near enterprise a, the value is 1, otherwise it is 0;
$R'_{CE(d,z)}$: Boolean variable, if detection equipment d is near enterprise a, the value is 1, otherwise it is 0;
$R_{SE(i,v,w)}$: Boolean variable, for workpiece i, if the location of warehouse w is near the affiliated enterprise of the machine that processes the process v, the value is 1, otherwise it is 0;
$R_{YE(i,v,r)}$: Boolean variable, for workpiece i, if the location of vehicle r is near the affiliated enterprise of the machine that processes the process v, the value is 1, otherwise it is 0;
$R_{CE(i,v,d)}$: Boolean variable, for workpiece i, if the location of detection equipment is near the affiliated enterprise of the machine that processes the process v, the value is 1, otherwise it is 0.

4. MATHEMATICAL MODEL FOR MRFJSP

Based on the above arrangement, the objective of each factor is established as follows:

$$T_{AM(i)} = \sum_{v=1}^{p(i)} T_{M(i,v)} \tag{1}$$

$$T_{AS(i)} = \sum_{v=1}^{p_{(i)}} T_{S(i,v)} \tag{2}$$

$$T_{AY(i)} = \sum_{v=1}^{p_{(i)}+1} T_{Y(i,v)} \tag{3}$$

$$T_{AYW(i)} = \sum_{v=1}^{p_{(i)}} T_{YW(i,v)} \tag{4}$$

$$T_{AC(i)} = \sum_{v=1}^{p_{(i)}} T_{C(i,v)} \tag{5}$$

Among them:

$$T_{M(i,v)} = \sum_{j=1}^{m} [T_{GM(i,v,j)} \cdot R_{M(i,v,j)}] \tag{6}$$

$$T_{S(i,v)} = \sum_{w=1}^{x} [T_{GS(i,v,w)} \cdot R_{S(i,v,w)} \cdot R_{SE(i,v,w)}] \tag{7}$$

$$T_{Y(i,v)} = \sum_{r=1}^{y} [T_{GY(i,v,r)} \cdot R_{Y(i,v,r)} \cdot R_{YE(i,v,r)}] \tag{8}$$

$$T_{YW(i,v)} = \sum_{r=1}^{y} [T_{GYW(i,v,r)} \cdot R_{Y(i,v,r)} \cdot R_{YE(i,v,r)}] \tag{9}$$

$$T_{C(i,v)} = \sum_{d=1}^{z} [T_{GC(i,v,d)} \cdot R_{C(i,v,d)} \cdot R_{CE(i,v,d)}] \tag{10}$$

$i = 1, 2, \ldots, n$, $v = 1, 2, \ldots, p_{(i)}$, $j = 1, 2, \ldots, m$, $w = 1, 2, \ldots, x$, $r = 1, 2, \ldots, y$, $d = 1, 2, \ldots, z$.

The makespan is minimized by Eq. (11):

$$\text{Min}\{\text{Max}[T_{AM(i)} + T_{AS(i)} + T_{AY(i)} + T_{AYW(i)} + T_{AC(i)}]\} \tag{11}$$

The following constraints are considered for the above model:

① One machine can only process one process of a workpiece at a time.

During time span $[T_{GST(i',v')},\ \ T_{GST(i',v')} + T_{GM(i',v',j')}]$:

$$R_{M(i',v',j')} = 1 \tag{12}$$

Then:

$$\sum_{i=1}^{n} \sum_{v=1}^{p^{(i)}} R_{M(i,v,j')} = 1 \tag{13}$$

$i' \in [1, n]$, $v' \in [1, p_{(i)}]$, $j' \in [1, m]$.

② One process can only be processed by one machine.

During time span $[T_{GST(i',v')},\ \ T_{GST(i',v')} + T_{GM(i',v',j')}]$:

$$R_{M(i',v',j')} = 1 \tag{14}$$

Then:

$$\sum_{j=1}^{m} R_{M(i',v',j)} = 1 \tag{15}$$

③ For workpiece i, if a succession of $u + 1$ processes between process v and process $v + u$ are processed in the same enterprise, warehouse w stores workpiece i after process v is completed. During the $u + 1$ processes, workpiece i is stored in the same warehouse, and the transport time between $u + 1$ processes is ignored.

If:

$$R_{E(i,v,a)} = R_{E(i,v+1,a)} = \cdots = R_{E(i,v+u,a)} = 1 \tag{16}$$

then:

$$R_{S(i,v,w)} = R_{S(i,v+1,w)} = \cdots = R_{S(i,v+u,w)} = 1 \tag{17}$$

$$T_{Y(i,v+1)} = \cdots T_{Y(i,v+u)} = 0 \tag{18}$$

$i \in [1, n]$, $v \in [1, p_{(i)}]$, $a \in [1, q]$, $w \in [1, x]$.

④ For workpiece i, process v is processed on machine j', process $v+1$ is processed on machine j''. Machine j' and machine j'' belong to the same enterprise a. After completion of process v, $T'_{(i,v)}$ is generated by adding together the accumulated flow time of workpiece i and the detection time $T_{C(i,v)}$. If $T'_{(i,v)}$ has reached the starting time $T_{MS(j'')}$ of process $v+1$, process $v+1$ can be processed; otherwise, workpiece i will be stored in the corresponding warehouse until the machine which processes process $v+1$ of workpiece i is idle, then the next process can be processed.

If:

$$R_{M(i,v,j')} = R_{M(i,v+1,j'')} = 1 \tag{19}$$

$$R_{ME(j',a)} = R_{ME(j'',a)} = 1 \tag{20}$$

then:

$$T_{S(i,v+1)} = \begin{matrix} T_{MS(j'')} \quad T'_{(i,v)} & , & T_{MS(j'')} \quad T'_{(i,v)} > 0 \\ 0 & , & T_{MS(j'')} \quad T'_{(i,v)} \leqslant 0 \end{matrix} \tag{21}$$

$j', j'' \in [1, m]$.

⑤ For workpiece i, process v is processed on machine j', process $v+1$ is processed on machine j''. Machine j' belongs to enterprise a'. Machine j'' belongs to enterprise a''. After completion of process v, $T'_{(i,v)}$ is generated by adding together the accumulated flow time of workpiece i, the detection time $T_{C(i,v)}$ and the transport time $T_{Y(i,v+1)}$. If $T'_{(i,v)}$ has reached the starting time $T_{MS(j'')}$ of process $v+1$, process $v+1$ can be processed; otherwise, the time difference between the starting time of machine j'' and the accumulated time before process $v+1$ is calculated. If the time difference is less than a threshold value MT, the car which transports workpiece i waits until machine j'' is idle; otherwise, workpiece i will be stored in the corresponding warehouse.

$$R_{M(i,v,j')} = R_{M(i,v+1,j'')} = 1 \tag{22}$$

$$R_{ME(j',a')} = R_{ME(j'',a'')} = 1 \tag{23}$$

Then:

$$T_{S(i,v+1)} = \begin{matrix} 0 & , & (T_{MS(j'')} \quad T'_{(i,v)}) \leqslant \mathrm{MT} \\ T_{MS(i+1)} \quad T'_{(v,i)} & , & (T_{MS(j'')} \quad T'_{(i,v)}) > \mathrm{MT} \end{matrix} \tag{24}$$

$$T_{YW(i,v+1)} = \begin{cases} T_{MS(j'')} - T'_{(i,v)} & , \quad (T_{MS(j'')} - T'_{(i,v)}) \le \mathrm{MT} \\ 0 & , \quad (T_{MS(j'')} - T'_{(i,v)}) > \mathrm{MT} \end{cases} \tag{25}$$

$a', a'' \in [1, q]$.

5. ELITIST SELECTION GENETIC ALGORITHM

Based on the description and model of MRFJSP, the problem obviously belongs to NP-hard. Elitist selection genetic algorithm (ESGA) is an advanced algorithm based on basic GA. In ESGA, the best solution will be saved in each iteration. Due to the complexity of MRFJSP, an ESGA is proposed to solve the problem. Each module will be introduced as follows.

5.1 Encoding

The feasible solution for MRFJSP is composed of seven chains, including workpiece, processing sequence, machine, enterprise, warehouse, vehicle, and detection equipment chains. Table I shows a sample coding.

Table I: Sample Coding of MRFJSP.

Workpiece	1	2	3
Processing sequence	2, 6, 8	4, 5, 7	**1**, 3, 9
Machine	6, 1, 3	2, 4, 1	4, 1, 5
Enterprise	2, 2, 1	1, 3, 2	3, 2, 3
Warehouse	5, 1, 3	3, 2, 4	5, 1, 4
Vehicle	7, 2, 3, 1	5, 4, 1, 3	2, 2, 6, 8
Detection equipment	3, 1, 2	6, 5, 1	6, 3, 1

The solution will be decoded according to the processing sequence chain. For instance, the smallest code 1 of the processing sequence is in the first position of the processing sequence of workpiece 3; thus the first process of workpiece 3 should be processed firstly, and so on.

5.2 Population

The population size is M. According to the above encoding method, M chromosomes will be generated randomly, and the initial population of the problem is generated. The makespan is set as the fitness value for MRFJSP and calculated by Eq. (11).

5.3 Selection operator

The method of roulette wheel selection and elitist strategy are combined. Population size is M. The best and the second best individuals are copied to the next generation. Fitness value of chromosome i is $f_{(i)}$, and selective probability is $P_{S(i)}$:

$$P_{S(i)} = [(\sum_{k=1}^{M} f_{(k)}) / f_{(i)}] / [\sum_{j=1}^{M} (\sum_{k=1}^{M} f_{(k)}) / f_{(j)}] \tag{26}$$

5.4 Crossover operator

The crossover operation includes two steps. Firstly, assuming the processing sequence chain is crossed, there are 3 crossover positions, and the codes 2, 6, 7 in the processing sequence will be operated. The crossover operation of the processing sequence chain is shown in Fig. 1.

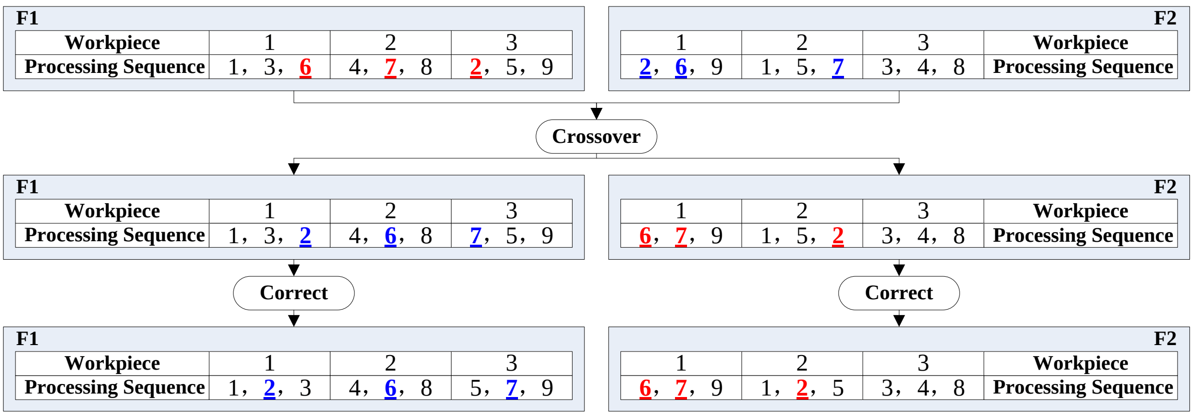

Figure 1: Crossover operation of the chain of the processing sequence.

Secondly, assuming all the other chains are crossed, workpiece 3 will be operated. The crossover operation of the other chains is shown in Fig. 2.

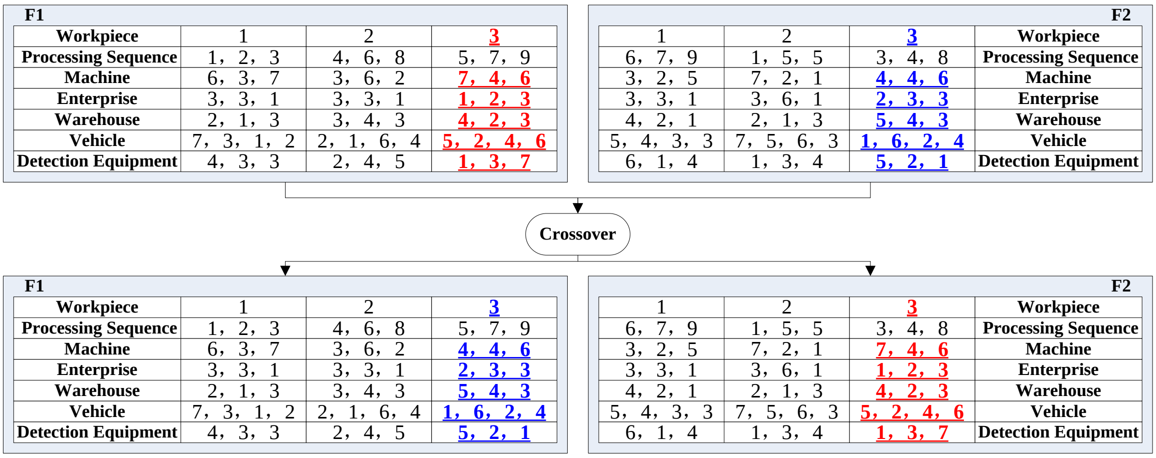

Figure 2: Crossover operation of other chains.

5.5 Mutation operator

Mutation operation includes two steps. Processing sequence chain is mutated firstly. An even number of codes in processing sequence will be generated randomly. The codes are switched back and forth in pairs, as shown in Fig. 3. Secondly, the multi-point mutation is adopted to mutate other chains. For example, the mutation operation of machine chain is shown in Fig. 4. After machine chain is mutated, it is necessary to rematch machines to enterprises.

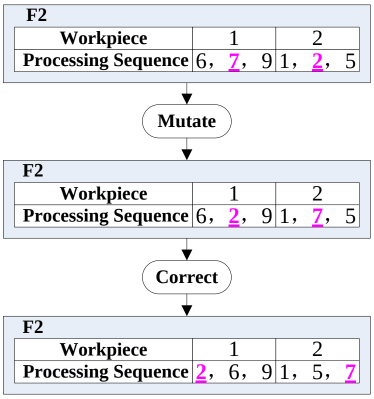

Figure 3: Mutation of the chain of the processing sequence.

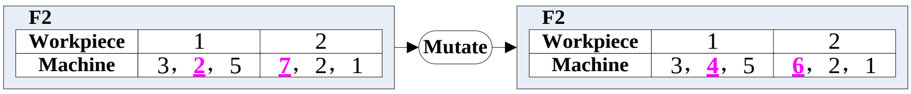

Figure 4: Mutation of the chain of machines.

5.6 Parameters of the genetic algorithm

M is population size, controlling the number of chromosomes; D is iteration, controlling the number of iterations of the algorithm, and the algorithm stops if the number of iterations reaches D; $P_{s(i)}$ is selection probability of each individual, $i = 1, 2, \ldots, M$; P_c is crossover probability, decimal in (0, 1) interval; P_m is mutation probability, decimal in (0, 1) interval; $f_{(i)}$ is fitness value of each individual; P_{s1}, P_{s2} is temporary selection probability, decimal in (0, 1) interval; P_{ctem} is temporary crossover probability, decimal in (0, 1) interval; F_{tem} is temporary population, and temporary chromosome is stored.

Based on the description of modules and parameters of ESGA, flowchart of the algorithm is shown in Fig. 5.

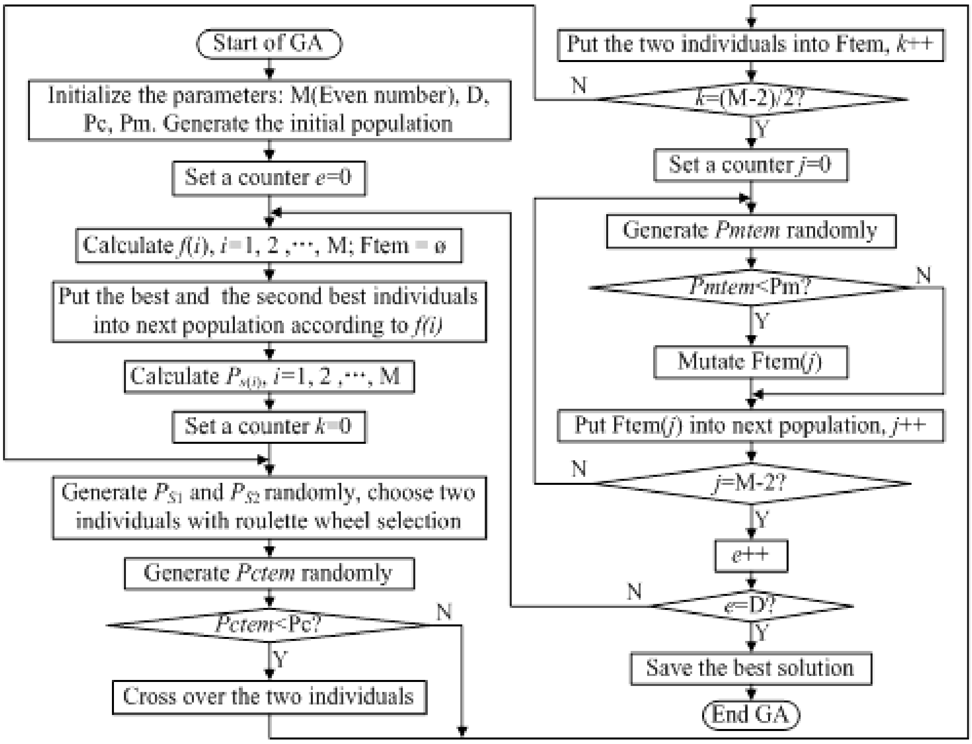

Figure 5: Flowchart of the Elitist Selection Genetic Algorithm for MRFJSP.

6. EXPERIMENT RESULTS AND ANALYSIS

Enterprise X has an order which requires processing four kinds of workpieces. Due to the limited resources of enterprise X, there are four continuous processes for each workpiece which need to be processed on twelve machines (M1-M12). They belong to four enterprises (A, B, C, D). There are ten warehouses (S1-S10), eight vehicles (Y1-Y8) and ten pieces of detection equipment (C1-C10). Assuming they are adequate and available anytime, the appropriate resources are chosen for each process of each workpiece in order to minimize the makespan.

6.1 Sample

Subordinate relationships between machines and enterprises, distance relationships between warehouses and enterprises, distance relationships between vehicles and enterprises, distance relationships between detection equipment and enterprises, available starting time of each machine, processing time of each machine, transport time of each vehicle, detection time of each piece of detection equipment are shown in Table II to IX.

Table II: Subordinate relations between machines and enterprises.

Machine	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12
Enterprise	C	D	A	D	A	C	A	D	B	B	B	C

Table III: Distance relationships between warehouses and enterprises.

Warehouse	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10
Enterprise	A	C	D	C	C	B	B	D	A	A

Table IV: Distance relationships between vehicles and enterprises.

Vehicle	Y1	Y2	Y3	Y4	Y5	Y6	Y7	Y8
Enterprise	C	A	B	A	D	C	X	B

Table V: Distance relationships between detection equipment and enterprises.

Detection equipment	Y1	Y2	Y3	Y4	Y5	Y6	Y7	Y8	Y9	Y10
Enterprise	A	D	B	C	D	C	D	B	C	A

Table VI: Available starting time of each machine (hour).

M1	**M2**	**M3**	**M4**	**M5**	**M6**	**M7**	**M8**	**M9**	**M10**	**M11**	**M12**
42	32	0	18	39	7	15	55	62	12	13	26

Table VII: Processing time of every machine (hour).

Workpiece	1				2				3				4			
Sequence	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
M1						5.4			7.0		8.1					5.7
M2					5.3	5.8	6.6					6.2				
M3	6.5				5.4					7.8				5.5	6.2	5.0
M4		7.7		6.5							7.6		4.4			
M5							6.8	5.5	7.3					5.1		
M6	6.2		7.8			6.5						6.4			6.3	5.5
M7	6.3			5.6						8.4	8.0					
M8						5.6	6.6		7.3							5.2
M9			8.1					6.1		7.8			4.9			
M10	6.9	7.8			5.1							6.1			6.1	
M11		7.5	8.0	5.8					7.6					5.6		
M12								5.6			7.8		5.0	5.2		

Table VIII: Transport time of each vehicle (hour).

	Y2(A)	**Y4(A)**	**Y3(B)**	**Y8(B)**	**Y1(C)**	**Y6(C)**	**Y5(D)**	**Y7(X)**
A	—	—	0.8	0.6	0.5	0.8	0.6	0.6
B	0.5	0.8	—	—	0.7	0.5	0.5	0.5
C	0.4	0.5	0.7	0.6	—	—	0.6	0.4
D	0.6	0.4	0.4	0.5	0.7	0.5	—	0.4
X	0.5	0.6	0.6	0.7	0.7	0.5	0.8	—

Table IX: Detection time of each piece of detection equipment (hour).

Workpiece	1				2				3				4			
Sequence	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
C1	0.5	0.3	0.4	0.6	0.4	0.4	0.5	0.4	0.3	0.7	0.4	0.4	0.8	0.4	0.5	0.7
C2	0.6	0.7	0.7	0.4	0.3	0.9	0.4	0.4	0.5	0.8	0.4	0.5	0.9	0.4	0.6	0.4
C3	0.3	0.4	0.6	0.5	0.7	0.7	0.3	0.6	0.6	0.6	0.6	0.6	0.7	0.3	0.5	0.5
C4	0.5	0.5	0.5	0.3	0.5	0.5	0.4	0.6	0.7	0.4	0.4	0.5	0.4	0.5	0.4	0.5
C5	0.4	0.6	0.4	0.3	0.5	0.6	0.3	0.7	0.8	0.4	0.4	0.4	0.5	0.5	0.7	0.6
C6	0.6	0.4	0.4	0.4	0.6	0.8	0.5	0.3	0.9	0.6	0.5	0.6	0.5	0.6	0.3	0.5
C7	0.7	0.4	0.5	0.4	0.5	0.5	0.6	0.3	0.8	0.5	0.3	0.3	0.6	0.7	0.5	0.5
C8	0.6	0.6	0.7	0.6	0.5	0.5	0.6	0.5	0.5	0.7	0.5	0.4	0.8	0.4	0.6	0.4
C9	0.3	0.5	0.6	0.5	0.4	0.5	0.4	0.3	0.5	0.6	0.6	0.6	0.6	0.7	0.7	0.4
C10	0.4	0.6	0.5	0.7	0.5	0.5	0.6	0.5	0.4	0.4	0.5	0.6	0.6	0.9	0.7	0.5

6.2 Assumptions

Four assumptions are considered in this problem:

- All equipment except for machines is available anytime and adequate;
- Workpieces are processed according to processing sequence;
- Waiting between processes is allowed;
- Machines will never break down.

6.3 Results and analysis

Set $M = 30$, $D = 100$, $P_c = 0.9$, $P_m = 0.2$, $MT = 0.5$ hours. The detected processes of workpieces 1, 4 are processes 2, 3. The detected processes of workpieces 2, 3 are processes 3, 4. The C # programming language is used for performing of the algorithm. The obtained optimal makespan is 45.7 hours. The best solution is shown in Table X. The time period of each resource is shown in Fig. 6. The curve of makespan of best solutions in each generation is shown in Fig. 7.

Table X: Best solution.

Workpiece	1	2	3	4
Processing sequence	1, 3, 6, 15	4, 7, 10, 12	2, 8, 11, 16	5, 9, 13, 14
Machine	6, 10, 11, 7	3, 6, 2, 5	11, 7, 4, 6	4, 3, 3, 3
Enterprise	3, 2, 2, 1	1, 3, 4, 1	2, 1, 4, 3	4, 1, 1, 1
Warehouse	2, 0, 0, 0	0, 4, 3, 0	7, 0, 0, 0	3, 0, 0, 0
Vehicle	7, 6, 0, 2, 2	2, 2, 6, 4, 2	7, 2, 4, 6, 7	7, 4, 0, 0, 2
Detection equipment	0, 3, 3, 0	0, 0, 5, 1	0, 0, 7, 4	0, 1, 1, 0

Workpiece1

X	Y7	S2	M6	—	Y6	—	M10	C3	—	—	M11	C3	Y2	—	M7	—	Y2	X
	0~0.4	0.4~7	7~13.2	—	13.2~13.7	—	13.7~21.5	21.5~21.9	—	—	21.9~29.9	29.9~30.5	30.5~31	—	31~36.6	—	36.6~37.1	

Workpiece2

X	Y2	—	M3	—	Y2	S4	M6	—	Y6	S3	M2	C5	Y4	—	M5	C1	Y2	X
	0~0.5	—	0.5~5.9	—	5.9~6.3	6.3~13.2	13.2~19.7	—	19.7~20.2	20.2~32	32~38.6	38.6~38.9	38.9~39.3	—	39.3~44.8	44.8~45.2	45.2~45.7	

Workpiece3

X	Y7	S7	M11	—	Y2	—	M7	—	Y4	—	M4	C7	Y6	—	M6	C4	Y7	X
	0~0.5	0.5~13	13~20.6	—	20.6~21.1	—	21.1~29.5	—	29.5~29.9	—	29.9~37.5	37.5~37.8	37.8~38.3	—	38.3~44.7	44.7~45.2	45.2~45.6	

Workpiece4

X	Y7	S3	M4	—	Y4	—	M3	C1	—	—	M3	C1	—	—	M3	—	Y2	X
	0~0.4	0.4~18	18~22.4	—	22.4~22.8	—	22.8~28.3	28.3~28.7	—	—	28.7~34.9	34.9~35.4	—	—	35.4~40.4	—	40.4~40.9	

Figure 6: Time period of each resource.

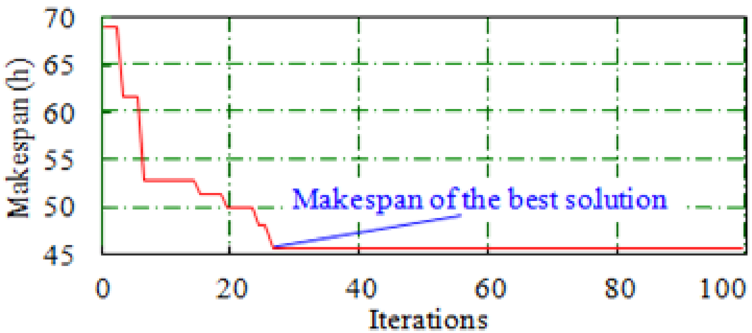

Figure 7: Curve of makespan of best solutions in each generation.

The result will be analysed from three aspects which include vehicles and detection equipment, warehouses and machines.

① Vehicle and detection equipment

The result shows that although each vehicle and piece of detection equipment utilizes a small period of time, they still affected the production schedule. It can be inferred that if vehicles and detection equipment are not considered, the obtained makespan will be less than the value of the current best solution. But in the actual production environment, transportation and detection are indispensable factors in the production process. Obviously, vehicles and detection equipment are taken into account in FJSP, which will better accommodate the actual production environment.

② Warehouse

In Fig. 6, workpieces are stored in some warehouses after some processes are completed. In actual production, all processes of workpieces are not processed uninterruptedly. When the machine that processes the next process of a workpiece is still working, the workpiece must be temporarily stored in the corresponding warehouse. So, it is necessary to take warehouse into account in the overall scheduling.

③ Machine

The result shows that, on the one hand, machines of the best solution are dispersed, and therefore, workloads of these machines are reduced. Delays in available starting times of some machines may be shorted. On the other hand, the reasonable configuration toward each workpiece has reduced the time of storage and transport to some extent.

7. CONCLUSION

In order to make FJSP more in line with the actual production environment, machines, warehouses, vehicles and detection equipment are all considered to be the scheduled resources. The makespan is the objective which will be minimized. A MRFJSP is proposed and a mathematical model for the problem is established. An elitist selection genetic algorithm is proposed to solve the problem. The result shows that all the considered resources can affect the production schedule. Although processing is an important factor in production scheduling, in actual practice, storage, transport and detection are also factors that cannot be disregarded. The makespan is affected by these factors. Obviously, taking these resources into account is a more practical approach. The optimal solution of MRFJSP has better reference and guidance functions when making a real processing solution.

ACKNOWLEDGEMENT

This research is supported by Nature Science Foundation of China (Grant No. 61402361, Grant No. 51575443).

REFERENCES

[1] Garey, M. R.; Johnson, D. S.; Sethi, R. (1976). The complexity of flowshop and jobshop scheduling, *Mathematics of Operations Research*, Vol. 1, No. 2, 117-129, doi:10.1287/moor.1.2.117

[2] Brucker, P.; Schlie, R. (1990). Job-shop scheduling with multi-purpose machines, *Computing*, Vol. 45, No. 4, 369-375, doi:10.1007/BF02238804

[3] Ho, N. B.; Tay, J. C.; Lai, E. M.-K. (2007). An effective architecture for learning and evolving flexible job-shop schedules, *European Journal of Operational Research*, Vol. 179, No. 2, 316-333, doi:10.1016/j.ejor.2006.04.007

[4] Wang, Y.; Feng, Y.-X.; Tan, J.-R.; Li, Z.-K. (2011). Optimization method of flexible job-shop scheduling based on multiobjective particle swarm optimization algorithm, *Transactions of the Chinese Society for Agricultural Machinery*, Vol. 42, No. 2, 190-196

[5] Karimi, H.; Rahmati, S. H. A.; Zandieh, M. (2012). An efficient knowledge-based algorithm for the flexible job shop scheduling problem, *Knowledge-Based Systems*, Vol. 36, 236-244, doi:10.1016/j.knosys.2012.04.001

[6] Gutiérrez, C.; García-Magariño, I. (2011). Modular design of a hybrid genetic algorithm for a flexible job-shop scheduling problem, *Knowledge-Based Systems*, Vol. 24, No. 1, 102-112, doi:10.1016/j.knosys.2010.07.010

[7] Li, X.-L.; Lu, J.-S.; Chai, G.-Z.; Tang, H.-T. (2011). Hybrid bee colony algorithm for flexible job shop scheduling problem, *Computer Integrated Manufacturing Systems*, Vol. 11, No. 7, 1495-1500

[8] He, X.-J.; Zeng, J.-C. (2012). Solving flexible job-shop scheduling problems with Bayesian statistical inference-based estimation of distribution algorithm, *Systems Engineering – Theory & Practice*, Vol. 32, No. 2, 380-388

[9] Yazdani, M.; Amiri, M.; Zandieh, M. (2010). Flexible job-shop scheduling with parallel variable neighbourhood search algorithm, *Expert Systems with Applications*, Vol. 37, No. 1, 678-687, doi:10.1016/j.eswa.2009.06.007

[10] Zhang, T.-N.; Han, B.; Yu, B. (2011). Flexible job-shop scheduling optimization based on improved genetic algorithm, *Systems Engineering – Theory & Practice*, Vol. 31, No. 3, 505-511

[11] Moslehi, G.; Mahnam, M. (2011). A Pareto approach to multi-objective flexible job-shop scheduling problem using particle swarm optimization and local search, *International Journal of Production Economics*, Vol. 129, No. 1, 14-22, doi:10.1016/j.ijpe.2010.08.004

[12] Peng, J.-G.; Liu, M.-Z.; Zhang, M.-X.; Zhang, X.; Ge, M.-G. (2014). Cloud model evolutionary multi-objective flexible job-shop scheduling based on improved non-dominated sorting, *Journal of Mechanical Engineering*, Vol. 50, No. 12, 198-205

[13] Chen, C.; Xing, L.-N. (2011). GA-ACO for solving flexible job shop scheduling problem, *Computer Integrated Manufacturing Systems*, Vol. 17, No. 3, 615-621

[14] Shahsavari-Pour, N.; Ghasemishabankareh, B. (2013). A novel hybrid meta-heuristic algorithm for solving multi objective flexible job shop scheduling, *Journal of Manufacturing Systems*, Vol. 32, No. 4, 771-780, doi:10.1016/j.jmsy.2013.04.015

[15] Shi, J.-F.; Jiao, H.-J.; Chen, T. (2012). Multi-objective Pareto optimization on flexible job-shop scheduling problem about due punishment, *Journal of Mechanical Engineering*, Vol. 48, No. 12, 184-192

[16] Liu, A. J.; Yang, Y.; Xing, Q. S.; Lu, H.; Zhang, Z. D. (2011). Multi-population genetic algorithm in multiobjective fuzzy and flexible job shop scheduling, *Computer Integrated Manufacturing Systems*, Vol. 17, No. 9, 1954-1961

[17] Palacios, J. J.; González, M. A.; Vela, C. R.; González-Rodriguez, I.; Puente, J. (2015). Genetic tabu search for the fuzzy flexible job shop problem, *Computers & Operations Research*, Vol. 54, 74-89, doi:10.1016/j.cor.2014.08.023

[18] Xu, Y.; Wang, L.; Wang, S.-Y.; Liu, M. (2015) An effective teaching-learning-based optimization algorithm for the flexible job-shop scheduling problem with fuzzy processing time, *Neurocomputing*, Vol. 148, 260-268, doi:10.1016/j.neucom.2013.10.042

[19] Özgüven, C.; Yavuz, Y.; Özbakır, L. (2012). Mixed integer goal programming models for the flexible job-shop scheduling problems with separable and non-separable sequence dependent setup times, *Applied Mathematical Modelling*, Vol. 36, No. 2, 846-858, doi:10.1016/j.apm.2011.07.037