

# Application of Hybrid Particle Swarm Optimization Algorithm in Workshop Scheduling Problem

Wang Guitang<sup>1,a</sup>, Chen Zhisheng<sup>1,b</sup>, Liang WenJie<sup>1,c</sup>, Yang ChaoQiong<sup>1,d</sup>

<sup>1</sup>Guangdong University of Technology, Guangzhou, 510006, China

<sup>a</sup>[745171013@qq.com](mailto:745171013@qq.com), <sup>b</sup>[1289161413@qq.com](mailto:1289161413@qq.com), <sup>c</sup>[1297498625@qq.com](mailto:1297498625@qq.com), <sup>d</sup>[1421235476@qq.com](mailto:1421235476@qq.com)

**Key words:** Particle Swarm Optimization, Heuristic algorithm, Crossover Operator, workshop problem, Area Search

**Abstract:** Because of the traditional genetic algorithm is difficult to consider both the quality of solution and the efficiency of convergence in solving the problem of constrained optimization in workshop scheduling problem. In this paper, we use the method of working procedure coding to generate feasible scheduling and learn from the single point crossover method of genetic algorithm to generate crossover operator which based on workpiece as the particle update mode. Then the improved Particle Swarm Optimization Algorithm will adopt in solving workshop scheduling problem, and enhance the particle convergence efficiency of particle swarm by using local search in the algorithm. We proved that the improved hybrid particle swarm optimization algorithm has a good performance in solving the workshop scheduling problem through the simulation experiments on the typical scheduling problems.

## Introduction

With the sustained and rapid development of the economy, the enterprises were forced to adjust their production and management strategy. As the core of manufacturing system, the development of efficient scheduling method is in line with the current economic situation and the needs of users. Under the known technical route and processing time, using limited workshop resources to satisfy the constraints of tasks and achieving the optimization of some production targets which can obviously improve the production efficiency of workshop.

Workshop scheduling problem is considered as the most difficult constrained combinatorial optimization problem and typical NP-hard problem. Its characteristic is that no efficient algorithm can find its optimal solution in polynomial time<sup>[1]</sup>. At present, the research on the workshop scheduling has made splendid achievements, a large number of intelligent optimization algorithms are widely used in this field. Liu Xiaobing, Jiao Xuan etc proposed a double chain quantum genetic algorithm to solve the flexible workshop problem<sup>[2]</sup>. Zhou Xin, Ma Yue etc proposed hybrid genetic simulated annealing algorithm for workshop scheduling problem<sup>[3]</sup>. Zhou Anyang etc proposed an improved artificial immune algorithm is used to solve the workshop scheduling problem<sup>[4]</sup>. Paulus Mudjihartono etc proposed the application of workshop scheduling based on genetic algorithm and particle swarm optimization<sup>[5]</sup>. Huang Yingjie etc used the hybrid particle swarm optimization algorithm based on entropy to apply in multi-objective flexible workshop scheduling<sup>[6]</sup>.

In this paper, we applied the improved hybrid particle swarm optimization algorithm in production scheduling. The simulation test has proved that the improved algorithm is effective and has good performance.

**Description of workshop scheduling problem and mathematical modeling**

**Problem description**

Workshop scheduling problem is a kind of resource allocation problem. It needs to configure the resources and tasks under the condition of satisfying various constraints. It is generally stated that in a manufacturing system, n workpiece are processed on m devices, the technical route of each piece is known and the time required for each process is clear. The task of scheduling is to arrange the processing sequence of the workpiece on each device reasonably under the limit condition in the workshop. Workshop scheduling is based on the following assumptions and constraints:

- (1) Workpiece processing sequence is not allowed to change;
- (2) One workpiece can only be processed by a single process in the same time;
- (3) Once the process begins, it cannot be interrupted;
- (4) The workpiece has no preemptive privileges;
- (5) The preparation time of the workpiece is negligible;

As shown in Table 1 is a typical workshop scheduling problem with two workpiece and three machines.

Table 1: worksheet list

workpiece	machine number, processing time		
	process1	process2	3rocess3
J <sub>1</sub>	(3, 5)	(2, 4)	(1, 8)
J <sub>2</sub>	(2, 1)	(3, 3)	(1, 5)

The corresponding workshop scheduling problem usually describe based on the technical route construction disjunctive graph. For the workshop scheduling problem with n workpieces and m machines, the corresponding disjunctive graph  $G = (V, A, E)$ . V is the vertex set of all operations; A is the solid line subset which represents a workpiece based on technical route of constraint condition; E is dotted subset which represents the sequence of the processes that are processed on the same device. Therefore, in the above two jobs, three equipment scheduling problem  $A = \{(1,2),(2,3),(4,5),(5,6)\}$  corresponding the process constraint of the same workpiece. The arc set  $E1 = \{(1,5)\}$  represents the process sequence of device 1. The arc set  $E2 = \{(2,4)\}$  represents represents the process sequence of device 2. The arc set  $E3 = \{(3,6)\}$  represents represents the process sequence of device 3. The corresponding disjunctive graph model as shown in Figure 1.

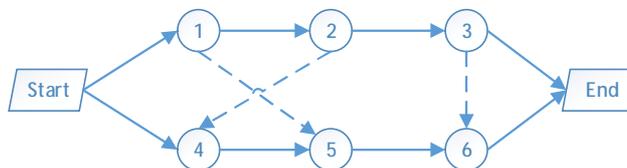


Figure 1: disjunctive graph model of workshop scheduling

**Mathematical model**

First define the meaning of the variables used:

$J = \{1, 2, \dots, n\}$ : represent the workpiece set of n workpieces;

$M = \{1, 2, \dots, m\}$ : represent the device set of m devices;

$O = \{1, 2, \dots, n * m\}$ : represent the set of all process;

$t_j$ : represent the processing time of process j;

$F_j$ : represent the process finish time;

$B_{jm}$ : represent the process j which processing in the device m, its value is 0 or 1;

$A(t)$ : represent the process set which processing on time t;

When solving the workshop scheduling problem, the most commonly used method that evaluate the advantages and disadvantages of the scheduling schemes is to transform the objective function into the minimization of the makespan. Based on this, the objective function of minimizing the maximum completion time is:

$$\min(t(JM)) = \min\{\max[t(1), t(2), \dots, t(i), \dots, t(m)]\} \tag{1.1}$$

The final finish time of all workpieces is expressed as t(JM). The final finish time of all workpieces in device i is expressed as t(i). Therefore, we can use the following models to describe the workshop scheduling problem:

$$\min(t(JM)) = \min\{\max[t(1), t(2), \dots, t(i), \dots, t(m)]\} \tag{1.2}$$

$$s.t F_k \leq F_j - t_j, j = 1, 2, \dots, n * m \tag{1.3}$$

$$\sum_{j \in A(t)} e_{jm} \leq 1, m \in M; t \geq 0 \tag{1.4}$$

$$F_j \geq 0, j = 1, 2, \dots, n * m \tag{1.5}$$

**Improvement and application of the algorithm**

**Particle swarm initialization**

In order to cover the entire solution space with a relatively large probability, the initial population must have a certain distribution. So we adopt heuristic HEH algorithm which can improve the search speed and reduce the probability of blind search to initialize the particle. It can not only ensure the distribution of the initial population, but also improve its quality.

**Encoding and decoding**

Using particle swarm optimization algorithm to solve the scheduling problem of n workpieces and m devices, each particle represents a potential solution to the problem. In this paper, the sequencing of all processes is used as the encoding method, so each particle in the particle swarm has n\*m gene which represent process to structure, which is a permutation of all the processes. Each workpiece appear m times. The i times appearance of the workpiece number from left to right represent the i procedure in the technical constraint of the workpiece. It ensures the processing sequences satisfy the technical route constraint after decoding and the effectiveness of the solution. This coding method can make any gene string substitution represent a feasible scheduling, and avoid deadlock when decoding.

Consider a scheduling problem with three workpieces and three devices, as shown in table 2.

Table 2: workpiece queue parameter table

workpiece	machine number, processing time		
	process1	process2	3rocess3
J <sub>1</sub>	(2, 2)	(1, 3)	(3, 2)
J <sub>2</sub>	(3, 1)	(2, 4)	(1, 2)
J <sub>3</sub>	(1, 4)	(3, 2)	(2, 3)

Assuming that there is an example of a=(1,2,3,2,3,1,3,2,1), the scheduling sequence is represented as follows:

1	2	3	2	3	1	3	2	1
---	---	---	---	---	---	---	---	---

Figure 2: scheduling sequence of particle a

Its decoding process is as follows:

(1) Marking the processing sequence of each workpiece. In the figure 3, it means the  $j$  times process of the  $i$  times workpiece;

1	2	3	2	3	1	3	2	1
$J_{11}$	$J_{21}$	$J_{31}$	$J_{22}$	$J_{32}$	$J_{12}$	$J_{33}$	$J_{23}$	$J_{13}$

Figure 3: processing sequence of particle a

(2) Marking the corresponding device number with the schedule table. In the figure 4, it means the device  $i$ ;

1	2	3	2	3	1	3	2	1
$M_{21}$	$M_{31}$	$M_{11}$	$M_{22}$	$M_{32}$	$M_{12}$	$M_{23}$	$M_{13}$	$M_{33}$

Figure 4: corresponding device number of processing sequence of particle a

(3) Marking the processing sequence of each equipment. In the figure 5, it means the  $j$  times processing sequence on the device  $i$ .

1	2	3	2	3	1	3	2	1
$M_{21}$	$M_{31}$	$M_{11}$	$M_{22}$	$M_{32}$	$M_{12}$	$M_{23}$	$M_{13}$	$M_{33}$

Figure 5: processing sequence of particle a on each device

(4) Generating activity scheduling Gantt chart.

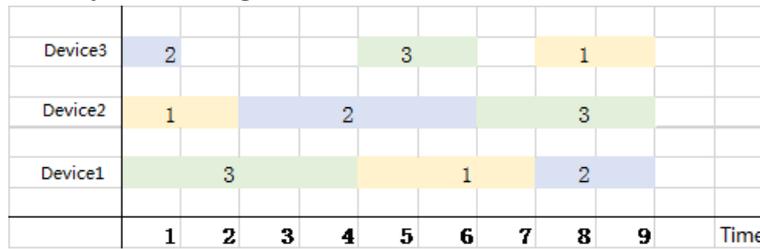


Figure 6: activity Gantt chart of particle a

After decoding, the maximum makespan is minimized by 9, i.e., makespan is 9 unit time.

**Secondary Encode**

In solving the job shop scheduling problem, we can assign the particle which have same feasible activity scheduling to the same particle. It is noted that the particle is not a one-to-one correspondence with the activity scheduling, and an activity schedule may correspond to a number of identical particles. Definitions such as particle  $a=(1,3,2,2,1,3)$  and particle  $b=(3,2,1,3,1,2)$  correspond to the same activity scheduling as shown in the figure. Therefore, the search space of particles is redundant.

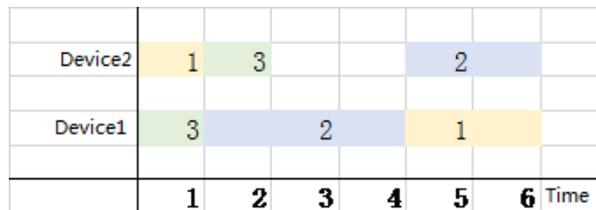


Figure 7: activity Gantt chart of particle a and b

We can get the order of the work on the device in sequence by scanning all the genes from the left to the right. Repeat the above operation for each device, we will obtain a sequence of  $S$ . When the  $S$  sequence of the two particle is the same, exchange one of the two genes randomly. This operation aim at reducing the redundancy of particle search space, so as to improve the quality of particle search.

**The update method of particle swarm**

In this paper, a crossover operator based on workpiece is used to be the update method. It achieves the update of particle by grouping, splitting and restructuring method. For a workshop scheduling problem with 4 workpieces and 3 devices, assuming particle  $a=(2,3,1,4,3,1,2,4,3,4,1,2)$  and  $b=(3,4,2,1,3,1,4,2,4,2,1,3)$ , its specific implementation is as follows:

(1) Grouping. Dividing the workpieces into two groups. Assuming the groups are  $J1=\{1,2\}$  and  $J2=\{3,4\}$ ;

(2) Splitting. Splitting particle a according to the workpieces contain in J1 workpiece set, so as to obtain new particle a1 and a2; Splitting particle b according to the workpieces contain in J2 workpiece set, so as to obtain new particle b1 and b2. As shown in figure 8:

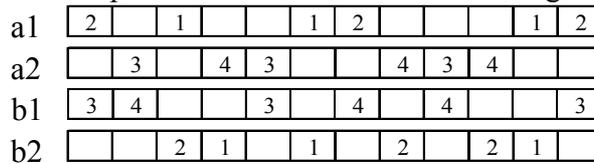


Figure 8: split of particle a and b

(3) Restructuring. Filling the gene of particle b1 from left to right in the blank of a1, so as to get the updated particle a'; Filling the gene of particle b2 from left to right in the blank of a2, so as to get the updated particle b'. As shown in figure 9:

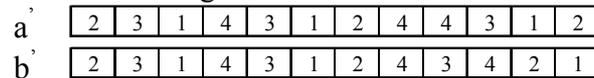


Figure 9: new particle a' and b'

**Local search**

In order to accelerate the convergence efficiency of particle swarm, the particle is updated by updating the particle swarm system. Using k bit adjustment to search the best solution from each particle randomly, in which bit is the upper limit of adjustment. The k bit adjustment using k position of the selected particles and exchange the selected genes randomly to finish the operation. Finally, the new particles are replaced by the adjusted operation to replace the worst particles in the original population.

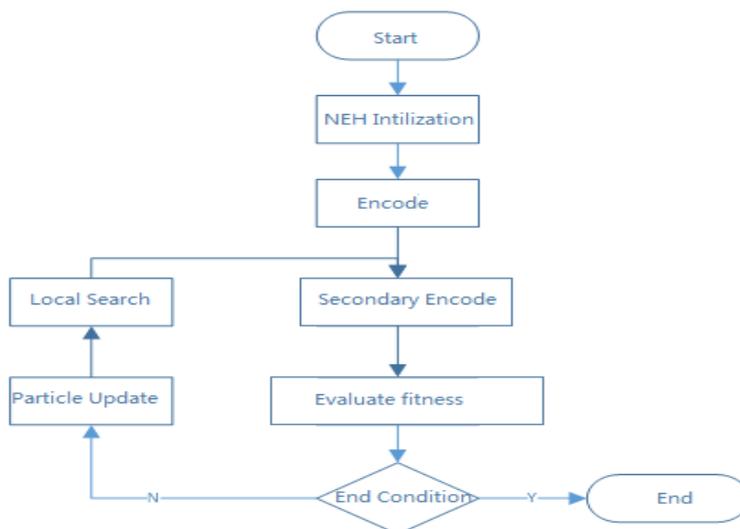


Figure 10: flow chart of improved particle swarm optimization algorithm

## Numerical simulation experiments and results

In order to prove the applicability of the improved algorithm for the workshop scheduling problem and the performance of the algorithm after improve, this paper selects several scheduling test functions to test the performance of the algorithm. It includes FT class mt06, mt10, MT20, LA class la04, la08, la12, la16, la20, La24, la28 a total of 10 test functions. The following formula was used to evaluate the fitness:

$$f_i = \frac{C_{best}}{T_i(JM)} \quad (3.1)$$

The fitness of the particle  $i$  is expressed as  $f_i$ , the known solution of the problem is expressed as  $C_{best}$ , and the minimized maximum completion time is expressed as  $T_i(JM)$ . The simulation experiment is implemented on the PC with Intel I7-4770HQ, 2.5GHz processor and 12G memory of 1600MHz frequency. It can be seen from the below experimental data that the improved hybrid particle swarm optimization algorithm to solve the workshop scheduling problem and the optimal solution obtained by the relative deviation of the problem is less than 2%. The simulation results are shown in figure 11:

Test problem	$C_{best}$	$C_{cur}$	Offset / %	Cpu /ms
mt06	55	55	0	5
mt10	930	935	0.54%	311
mt20	1165	1180	1.29%	632
la04	590	600	1.69%	3
la08	863	875	1.39%	30
la12	1039	1047	0.77%	564
la16	945	947	0.21%	734
la20	902	910	0.89%	458
la24	935	940	0.53%	882
la28	1216	1220	0.33%	1027

Figure 11: simulation result

## Conclusion

This paper describes the workshop scheduling problem and constructs a disjunctive graph model and corresponding mathematical model. Then we proposed a hybrid particle swarm optimization algorithm to solve the typical scheduling problem and given the corresponding solving process. Finally, the simulation results show that the algorithm has good performance in solving the workshop scheduling problem.

## Acknowledgement

This work was financially supported by the Guangdong Science and Technology Foundation( 2016A010119143, 201604010011, 2017A090905047).

## Reference

- [1] Blazewicz J, Ecker K H, Schmidt G, etc. Scheduling in Computer and Manufacturing Systems [C]. Springer Publishing Company, Incorporated, 1993: 1-2.
- [2] Liu Xiaobing, et al.Flexible job shop scheduling based on double-stranded quantum genetic algorithm [J]. Computer Integrated Manufacturing Systems, 2015,21 (2): 495-502
- Zhou Xin, et al.Hybrid genetic simulated annealing algorithm for solving job shop scheduling problem [J]. Microcomputer Systems, 2015, (2): 370-374

- Zhou Anyang, et al. An improved immune algorithm in shop scheduling [J]. Proceedings of the CSEE, 2012, 41 (19): 16-20 (in Chinese).
- [5] Paulus Mudjihartono, Rachsuda Jiamthapthaksin. Parallelized GA-PSO algorithm for solving Job Shop Scheduling Problem [C]. Thitipong Tanprasert: 2016 2nd International Conference on Science in Information Technology (ICSITech), 2016. 103-108
- [6] Huang Yingjie, et al. Application of entropy-based hybrid particle swarm optimization in flexible scheduling [J]. Journal of Hunan University of Technology, 2012, 39 (3): 48-52.
- [7] Zhang Qiliang, et al. Mixed particle swarm optimization algorithm based on NEH to solve the problem of no-wait flexible flowshop scheduling [J]. Systems Engineering - Theory & Practice, 2014 (3): 802-809.
- [8] Sheng Ligang, et al. Application of hybrid genetic algorithm NEH in flow shop scheduling [J]. Control Engineering, 2010, 17 (4): 94-97.