Hindawi
Mathematical Problems in Engineering
Volume 2017, Article ID 6758147, 9 pages
https://doi.org/10.1155/2017/6758147

Research Article

Integrating Preventive Maintenance Planning and Production Scheduling under Reentrant Job Shop

Ruiqiu Li[1] **and Huimin Ma**[2]

[1] *School of Management, University of Shanghai for Science and Technology, Shanghai 200093, China*
[2] *Business School, Shanghai Dianji University, Shanghai 201306, China*

Correspondence should be addressed to Huimin Ma; mahm81@163.com

Received 18 October 2016; Revised 2 February 2017; Accepted 7 February 2017; Published 19 March 2017

Academic Editor: Alberto Borboni

This paper focuses on a preventive maintenance plan and production scheduling problem under reentrant Job Shop in semiconductor production. Previous researches discussed production scheduling and preventive maintenance plan independently, especially on reentrant Job Shop. Due to reentrancy, reentrant Job Shop scheduling is more complex than the standard Job Shop which belongs to NP-hard problems. Reentrancy is a typical characteristic of semiconductor production. What is more, the equipment of semiconductor production is very expensive. Equipment failure will affect the normal production plan. It is necessary to maintain it regularly. So, we establish an integrated and optimal mathematical model. In this paper, we use the hybrid particle swarm optimization algorithm to solve the problem for it is highly nonlinear and discrete. The proposed model is evaluated through some simple simulation experiments and the results show that the model works better than the independent decision-making model in terms of minimizing maximum completion time.

1. Introduction

In 1993, Kumar [1] proposed the third mode of production, reentrant production, which was different from common Job Shop or Flow Shop when he studied the semiconductor manufacturing process. Reentrancy means that the same job, at various processing stages, may wait for processing before the same device or the workpiece, at different stages of processing, visits the same device repeatedly. This feature makes the process even more intense when different jobs are processed in the same machine, which will undoubtedly make scheduling more difficult. With the rapid development of semiconductor industry, the reentrant Job Shop scheduling problem (RJSP) has gotten more and more attention recently. What is more, semiconductor manufacturing equipment is extremely expensive. In the production process, an indeterminate fault might happen to these machines which will affect normal production planning. To improve the efficiency of the whole production system, equipment maintenance should also be considered at the same time. Combining production scheduling with preventive maintenance is worth discussing on RJSP. We propose an integrated and optimal mathematical model in this paper and use the hybrid particle swarm optimization algorithm to solve the problem. The proposed model is evaluated through simulation experiments and the results show that the model works better than the independent decision-making model in terms of minimizing maximum completion time.

RJSP is more complex than JSP because the same job will visit the same machine more than once at different stages. The main solving method is the heuristic algorithm. An approach of artificial neural network is proposed to solve multidecision scheduling problems of semiconductor wafer fabrication [2]. Topaloglu and Kilincli [3] proposed a modified shifting bottleneck heuristic for the reentrant Job Shop scheduling problem with makespan minimization objective. Zoghby et al. [4] built a model for reentrant Job Shop scheduling problem with sequence-dependent setup times and investigated the feasibility conditions for metaheuristic searches. Danping and Lee [5] gave a full picture of the research methodology for the reentrant scheduling problem. Two-level hierarchical production planning based on linear programming was proposed for semiconductor wafer fabrication in [6], which was proved to be better than an existing algorithm by simulation experiments. Gupta and Sivakumar [7] made a review on Job Shop scheduling techniques in semiconductor manufacturing. Mönch et al. [8] discussed a complex Job Shop environment, which contains parallel batching machines, machines with sequence-dependent setup times, and reentrant process flows, and a modified shifting bottleneck heuristic was used to solve this complicated problem.

Although many achievements have been made in semiconductor manufacturing production scheduling, they are studied only from the perspective of the production department to optimize production scheduling. They ignore the fact that the semiconductor manufacturing workshop equipment maintenance department arranges a large number of preventive maintenance tasks so that they will affect the equipment production. What is more, the random failure of machines will interrupt production scheduling. It will generate a production scheduling conflict when we discuss production scheduling and equipment maintenance separately. Obviously, the related theoretical study of production scheduling does not reflect the actual status of the reentrant shop job scheduling very well at present.

Equipment preventive maintenance is also important to improve production efficiency. Ma et al. [9] built an equipment maintenance scheduling model of semiconductor manufacturing and proposed a method which is called particle swarm optimization algorithm to solve the problem. Mosley et al. [10] compared different dispatching maintenance scheduling policies which can significantly affect system performance by using a discrete event simulation model of a wafer fabrication. Davenport et al. [11] presented a goal programming approach that incorporates both constraint programming and mixed-integer programming solution technologies. Yao et al. [12] proposed a two-level hierarchical modeling framework to solve preventive maintenance scheduling. At the higher level, there is a model for long-term planning, and, at the lower level, there is a model for short-term PM scheduling. Based on the above research, we can see that equipment maintenance can greatly improve production efficiency. However, connected equipment maintenance and production scheduling have more realistic significance.

Although joint optimization is rarely studied on semiconductor production, many researches have been carried out in traditional production scheduling and equipment maintenance. Cassady and Kutanoglu [13] proposed an integrated model to solve the problem of production scheduling and equipment maintenance on a single machine. Da and Wang [14] built an integrated model about production scheduling and equipment maintenance under the Flow Shop and proposed a heuristic approach to solve this problem. Da et al. [15] have done some researches on batch scheduling and maintenance under the Flow Shop and established a joint optimization model. Kaihara et al. [16] proposed a new approach, which is based on Lagrangian decomposition coordination, to solve the problem of maintenance scheduling. As for the complex workshop environment such as JSP, studies on joint optimization are very few. It will consider multiobjective optimization measures when optimizing processing sequence and maintenance time at the same time. Some papers are always introducing other objective functions such as system availability in order to reflect the influence of maintenance decision-making on the reliability of the equipment [17–19]. On the whole, the existing research on joint optimization is preliminary; most of the researches focus on basic workshop types and mode of production. They do not consider the reentrancy of complex production workshop.

Figure 1: Processing diagram of JSP.

Judging from the current research results, although joint optimization of production and maintenance has aroused more and more attention, studies on solving joint research of reentrant manufacturing production and equipment maintenance operations are very rare. In this paper, we combine production scheduling with preventive maintenance planning and establish an integrated optimization model of scheduling and preventive maintenance with the goal of minimizing completion time.

The paper is organized as follows. Section 2 is devoted to representing the integrated problem and building a mathematical model. Section 3 describes the process of hybrid particle swarm optimization algorithm. Section 4 presents the simulation results and comparison result. Section 5 discusses the conclusion and future research direction.

2. Problem Formulation

Standard JSP contains two subproblems: the machine selection problem and the operation sequencing problem. During the production, each job can only be processed on different machines only once. And each machine can process a job at the same time. The processing diagram is shown as Figure 1. The processing sequence on machines is 1→2→3→5. However, RJSP is more complicated and challenging than the standard JSP because it exhibits a reentrant situation, where different operations of the same job can be processed on the same machine. The processing diagram of RJSP is shown as Figure 2. The processing sequence on machines is 1→2→3→4→2→5. The job is processed on machine 2 twice. At the same time, if the machines fail during production, this will affect the whole production plan. So, we develop a joint mathematical optimization model to solve this problem in this paper. The objective of this problem is to minimize the maximal completion time of all the operations.

2.1. Assumption. RJSP and equipment preventive maintenance planning problem can be represented as follows: there are n machines in a workshop and m different jobs should be processed. It has been arranged which operation will be processed on which machine before production. Different operations of the same job may be processed on the same machine not only once. Each machine may have different

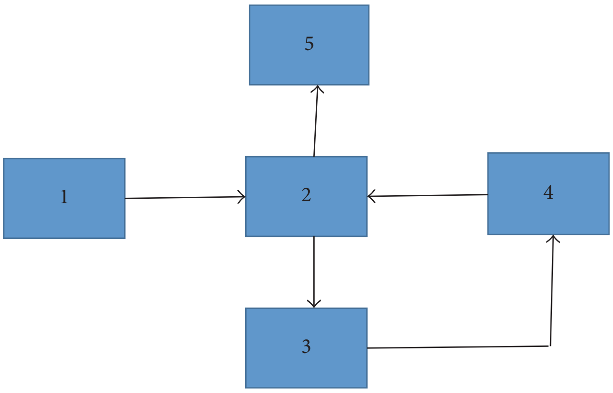


Figure 2: Processing diagram of RJSP.

abilities. Some hypotheses considered in this paper are summarized as follows:

(1) The processing time for each operation in a particular machine is defined.

(2) A machine can only process one operation at a certain time (resources constraint).

(3) Each job must be processed on one machine at a given time, so jumping the queue is not allowed.

(4) All the jobs have the same probability to be scheduled at the beginning, which means different jobs have the same priority.

(5) Each operation of each job is not allowed to be interrupted in processing time; that is, preventive maintenance operations must be before or after production.

(6) All machines are new at the beginning of the production process.

(7) Each machine recovers as new after the implementation of preventive maintenance.

(8) The time of small repair and machine setup is ignored during the processing.

2.2. Definition of Notations. The indices and decision variables used in the model are as follows.

Indices

i, h: job, $i, h = 1, \ldots, m$, where m is the total number of jobs.

j, k: machine, $j, k = 1, \ldots, n$, where n is the total number of machines.

l': sequence of assigned operation on machine j $(1, \ldots, d_j)$.

l_i: reentrance times $(l_i = 1, 2, \ldots)$.

$O_{ij}^{l_i}$: the operation of i job processed l_i times on machine j.

p_{ij}: processing time of $O_{ij}^{l_i}$.
s_{0j}: the initial age of machine k.

q_j: preventive maintenance time on machine j (if the maintenance interval iswithin the fixed time T', then $q_j = \theta$; if not, $q_j = \theta + \alpha\Delta t$, where Δt is the over time and α is a parameter).

$C_{ij}^{l_i}$: completion time of operation $O_{ij}^{l_i}$.

$r_{l'j}$: processing time of l'th operation processed on machine j.

$t_{ij}^{l_i}$: starting time of $O_{ij}^{l_i}$.

$T_{jl'}$: starting time of l'th operation processed on machine j.

s_{lk}: the age of machine j starting to process l'th job.

Variables

$$x_{ijl_il'} = \begin{cases} 1, & \text{if } O_{ij}^{l_i} \text{ is performed on machine } j \text{ in priority } l' \\ 0, & \text{otherwise} \end{cases}$$
$$y_{jl'} = \begin{cases} 1, & \text{machine } j \text{ conducts preventive maintenance after finishing } l'\text{th procession} \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

2.3. Maintenance Strategies. In the process of production, equipment maintenance is a basic work. Timely maintenance can restore equipment performance, fix the trouble, and effectively prolong the service life of equipment. So, it is one of the most important ways to ensure smooth production. For preventive maintenance, it can be roughly divided into two categories. One is preventive maintenance which is based on machine reliability, and the other is periodic preventive maintenance.

For preventive maintenance based on the reliability of the machine, when the reliability of the machine reaches the threshold set, it is time for equipment maintenance. Some scholars suppose the machines will return to the new state after maintenance in their studies. In fact, equipment performance is gradually degraded with decreasing machine age. Under the same maintenance conditions and maintenance time, the equipment degree of maintenance decreases with the increase of the maintenance times. So, it will take

Table 1: Job processing time on machines for the example.

		Machine 1	Machine 2
Job 1	Operation 1	10	5
	Operation 2	5	10

less and less time to get the reliability threshold. So, lots of scholars assume that the reliability threshold will reduce after maintenance. Periodic preventive maintenance is the most common method in industrial production. It means that when each machine achieves the maximum maintenance interval, it must be maintained in time. When there is small equipment failure, we assume that the maintenance time is negligible. So, we do not consider the failure rate of the machine. Thus, this article mainly uses the periodic preventive maintenance strategy.

In practice, many companies perform PM at a fixed interval. This kind of fixed interval is according to the processing time when starting production. If we calculate time according to processing time, the machines will be maintained excessively because some machines may not process any job during the maintenance cycle. So, we adopt other methods to determine PM intervals. When we calculate time, we focus on the age of the machine. This method is used in many researches [17, 18, 20]. For example, one job is processed on 2 machines and it has 4 operations. The processing time is as Table 1 shows. We assume preliminary maintenance cycle $T = 15$.

If we calculate time according to machine age, each machine just needs only onetime maintenance as depicted in Figure 3. However, each machine should be maintained twice if we calculate time according to processing time described in Figure 4. The increasing maintenance times will increase maintenance cost and delay production scheduling. So, the maintenance strategy based on machine age is more reasonable for our study.

2.4. Mathematical Model

2.4.1. Objective. The main goal in this paper is to minimize the makespan ($C_{\max}$), which is the cumulative time to complete all operations on all machines with maintenance.

$$C_{\max} = \min_{i} \max_{j} \max_{l_i} C_{ij}^{l_i}. \tag{2}$$

2.4.2. Jobs' Starting Production Time on Machines. The starting production time of the ith job, which is produced on the jth machine l_i times, is dependent on many factors. They are as follows:

(1) completion time of previous jobs on jth machine:

$$t_{ij}^{l_i} + p_{ij}^{l_i} \le t_{ik}^{l_i+1} \quad \forall i,\ \forall l_i,\ \forall j, k = 1, 2, \ldots, n. \tag{3}$$

(2) completion time of the previous operation of the ith job on another machine

$$t_{ij}^{l_i} + p_{ij}^{l_i} \le t_{ij+1}^{l_i} \quad \forall i,\ \forall j = 1, 2, \ldots, n - 1. \tag{4}$$

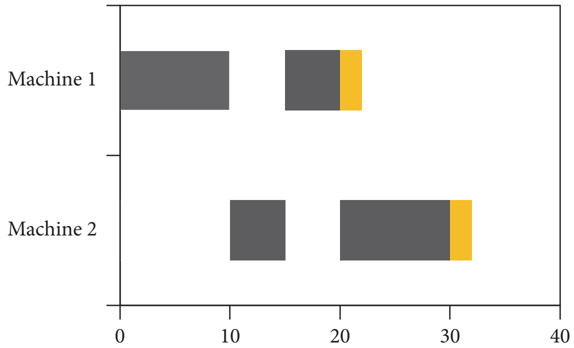


Figure 3: Maintenance based on machine age.

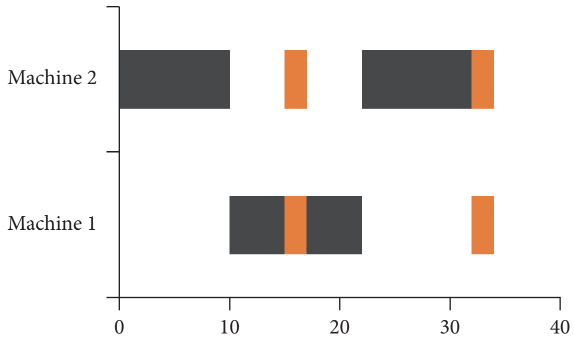


Figure 4: Maintenance based on processing time.

(3) time for implementing PM. In our model, PM is regarded as perfect, which means the machine will be renewed to an as-good-as-new status. We assume that the initial age of the machines is zero. Then, according to the machines' age and fixed maintenance cycle, we can decide whether PM should be conducted. Hence, the completion time of the current job can be calculated as follows:

$$\begin{aligned}
&T_{jl'} + p_{ij}^{l_i} * x_{ijl_il'} + y_{jl'} * q_j \le T_{jl'+1} \quad \forall i, j, l_i,\ \forall l' = 1, 2, \ldots, d_j - 1 \\
&T_{jl'} \le t_{ij}^{l_i} + \left(1 - x_{ijl_il'}\right) * M \quad \forall i, j, l_i, l' \\
&T_{jl'} + \left(1 - x_{ijl_il'}\right) * M \ge t_{ij}^{l_i} \quad \forall i, j, l_i, l' \\
&s_{l'j} = s_{0j}\left(1 - y_{jl'-1}\right) + y_{jl'-1}\left(s_{l'-1j} + r_{l'-1j}\right) \\
&\sum_{i} x_{ijl_il'} = 1 \quad \forall j, l'.
\end{aligned} \tag{5}$$

The completion production time of the ith job, which is produced on the jth machine l_i times, is calculated as follows:

$$C_{ij}^{l_i} = t_{ij}^{l_i} + p_{ij}^{l_i} \quad \forall i, j, l_i. \tag{6}$$

Due to reentrancy, the same kind of job in different stages will wait to be produced on one machine. So, we should decide which one will be produced firstly. The calculation is as follows:

$$\begin{gathered}
t_{ij}^{l_i} + p_{ij}^{l_i} \geq t_{hj}^{l_h} \\
\text{or} t_{hj}^{l_h} + p_{hj}^{l_h} \geq t_{ij}^{l_i} \\
\forall i, h, \ \forall l_i, l_h, \ \forall j, k = 1, 2, \ldots, n.
\end{gathered} \tag{7}$$

3. Hybrid Particle Swarm Optimization Algorithm

3.1. Workflow of the Proposed HPSOA. With high nonlinearity and discreteness, the reentrant Job Shop scheduling problem is a strongly NP-hard problem. It is very hard to be solved by traditional optimization methods within an acceptable time. In recent years, many scholars have been committed to using intelligent optimization algorithms to solve this problem, such as genetic algorithm, ant colony algorithm, and particle swarm optimization algorithm. General particle swarm optimization algorithm is to find the optimum value by tracking individual best value and group best value [21]. Although this algorithm is simple and its convergence rate is very fast, it will be trapped in local optimum and cannot jump out with increasing iteration times. Hybrid particle swarm algorithm discards the method of tracking the extreme to update the particle's location in the traditional particle swarm optimization algorithm. It introduces crossover and mutation in the genetic algorithm. The way of searching the optimal value is to find the best particle after crossing operation and mutation operation. So, the hybrid particle swarm optimization algorithm (HPSOA) [21] is mainly applied to solve the integrated problem in this paper.

Workflow of the proposed HPSOA is shown in Figure 5. All the details of the proposed HPSOA will be described in the following subsections. The overall procedure of the proposed approach is described as follows:

Step 1. Initialize the number of particles, the maximum number of iterations, learning factors, and so on.

Step 2. Initialize the particle's initial position and velocity and calculate the fitness value.

Step 3. Initialize the particle's personal best and global best.

Step 4. Calculate the value of inertia weight ω and update the particle's velocity and position.

Step 5. Calculate the fitness value of particles and then sort order according to the fitness value.

Step 6. Improve particles by crossover and mutation operation.

Step 7. Recalculate the fitness value.

Step 8. Obtain updated personal best and global best.

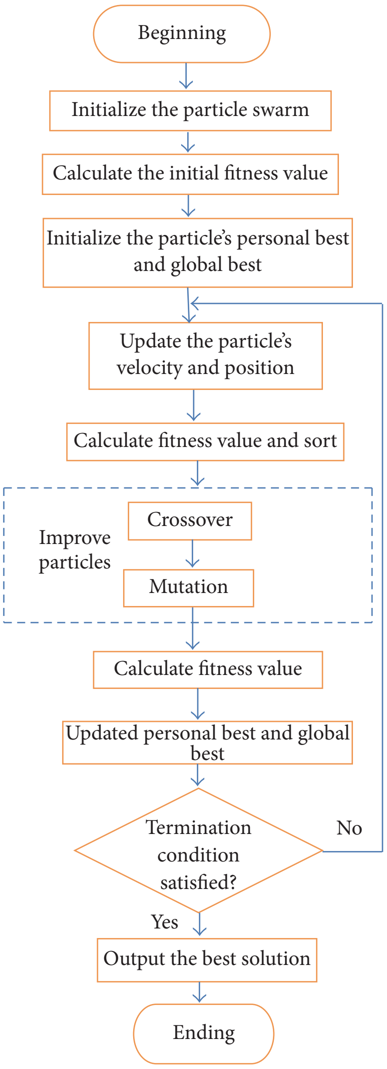

Beginning
Initialize the particle swarm
Calculate the initial fitness value
Initialize the particle's personal best and global best
Update the particle's velocity and position
Calculate fitness value and sort
Improve particles
Crossover
Mutation
Calculate fitness value
Updated personal best and global best
Termination condition satisfied?
No
Yes
Output the best solution
Ending

FIGURE 5: The workflow of HPOSA.

Step 9. If the termination condition is met, output the optimal solution. Otherwise, execute the loop body Steps 4 to 8.

3.2. Encoding. In our encoding, the particle is composed of two-dimensional vectors. The job sequence in the first dimension decides the job's processing sequence. The second dimension is a set of random numbers which can decide particles' velocity and position. The range of those random numbers is $[0, 1]$. It has been arranged which operation of each job will be processed on which machine before production. If job m has N operations, it will generate Nm. For example, if there are 2 jobs, each will have 3 operations. The job sequence is [1 1 1 2 2 2]. And generate a set of random numbers ranging from 0 to 1. The primary particle is [1 1 1 2 2 2; 0.8147 0.9058 0.1270 0.9134 0.6324 0.0975]. This set of random numbers is sorted in an ascending order. And the job sequence will change accordingly. The real particle is [2 1 2 1 1 2;

Table 2: Job processing time on each machine (time units).

	$O1$	$O2$	$O3$	$O4$	$O5$	$O6$	$O7$	$O8$	$O9$
$J1$	(1,20)	(3,10)	(2,39)	(5,24)	(4,33)	(2,17)	(3,26)	(1,22)	(2,19)
$J2$	(1,20)	(3,10)	(2,39)	(5,24)	(4,33)	(2,17)	(3,26)	(1,22)	(2,19)
$J3$	(1,18)	(2,37)	(3,36)	(4,45)	(2,30)	(3,18)	(4,31)	(1,24)	(5,33)
$J4$	(3,21)	(2,33)	(1,40)	(4,35)	(3,15)	(2,34)	(1,18)	(5,25)	(4,37)

0.0975 0.1270 0.6324 0.8147 0.9058 0.9134]. This means that the first operation of job 2 is processed firstly and then the first operation of job 1 is processed and so on.

3.3. Crossover. If the particle has reached the condition of crossover, we choose the particles to cross with the best personal particle and best group particle. The method of crossover is integer crossover. At first, we choose the position of crossover randomly. And then the selected one crosses with the personal best or group best. For example, if we choose the second position and fourth position to cross, the operation will be depicted as follows.

Individual

2	1	1	1	2	2
0.1576	0.2785	0.5469	0.9575	0.9649	0.9706

Best Personal Particle

2	2	1	1	2	1
0.1419	0.4218	0.4854	0.8003	0.9157	0.9572

New Particle

2	2	1	1	2	2
0.1576	0.4218	0.4854	0.8003	0.9649	0.9706

After crossing, the amount of processing of some jobs will increase or decrease. In this situation, we will take necessary actions to make sure the amount of processing meets the conditions. In the crossover process, we will choose the best individual to update the particle swarm according to the fitness value.

3.4. Mutation. In the mutation operation, two positions are changed with each other in the same individual. At first, we choose the mutation position, position 1 and position 2, and then change the mutual position. We just change the position of the job not the particle's position. For example, if we change the second position and fourth position, it will be described as follows:

2	2	1	1	2	1
0.1419	0.4218	0.4854	0.8003	0.915	0.9572

After Mutation

2	1	1	2	2	1
0.1419	0.4218	0.4854	0.8003	0.915	0.9572

4. Simulation and Discussion

4.1. Simulation. In real semiconductor manufacturing scheduling, the production process is very complex. Each job needs to go through hundreds of operations. In order to validate the superiority of joint optimization model and the effectiveness of hybrid particle swarm algorithm, we compare it with the independent decision-making model. In the example, there are 4 jobs that are processed on 5 machines in the system. Some machines will produce the same job more than once. In order to reflect the competition of similar products at different stages on the same machine, we assume that job 1 and job 2 are the same products. Job operation information is depicted in Table 2. In the array, the first number presents processing machine and the second number presents processing time.

Maintenance has degraded effect in the whole production process. Kubzin and Strusevich proposed a linear degradation function to calculate PM time [22]. In this example, we also use $q_j = \theta + \alpha\Delta t$ to calculate PM time. We suppose $\alpha = 0.5$. In this paper, the hybrid particle swarm optimization algorithm is used to minimize makespan and MATLAB 2010b is used to realize the algorithm. The algorithm parameters are set as follows: the particles' initial number is 200 and the number of loop iterations is 500. P_c is the crossover probability in g_n.

$$P_c = P_{c\max} - \frac{(P_{c\max} - P_{c\min}) \times g_n}{g_{n\max}}. \quad (8)$$

$P_{c\max}$ equals 1 which is the maximum crossover probability. $P_{c\min}$ equals 0.5 which is the minimum crossover probability. $g_{n\max}$ which represents the number of generations equals 200. P_m is the mutation probability in g_n.

$$P_m = P_{m\max} - \frac{(P_{m\max} - P_{m\min}) \times g_n}{g_{n\max}}. \quad (9)$$

$P_{m\max}$ equals 0.1 which is the maximum mutation probability. $P_{m\min}$ equals 0.01 which is the minimum mutation probability.

4.1.1. Integrated Optimization Strategy. By using hybrid particle swarm optimization algorithm, the optimal solution is 392.5. The Gantt chart of integrated optimization model is shown in Figure 6. In the figure, the horizontal axis is the time axis, the vertical axis is the machine order, the same color box in the figure represents the same job, and the red box denotes equipment maintenance.

Table 3: Comparison of independent decision making and integrated optimization.

Jobs	Machines	Reentrancy times	independent decision making	integrated optimization model	Improvement
4	5	3	411.5	390.5	4.62%
4	5	5	669	646	3.44%
5	7	3	696	650.5	6.54%
5	7	5	920.5	876.5	4.78%

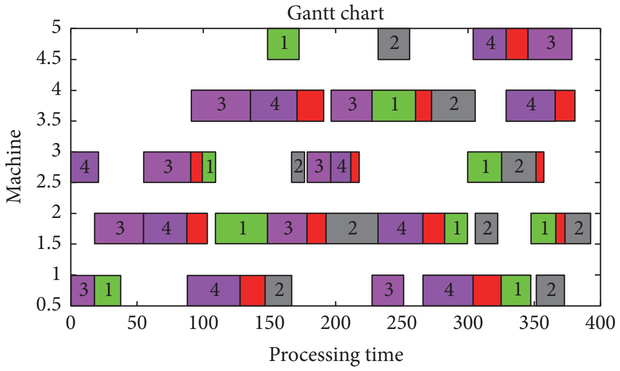


Figure 6: The Gantt chart of integrated optimization model.

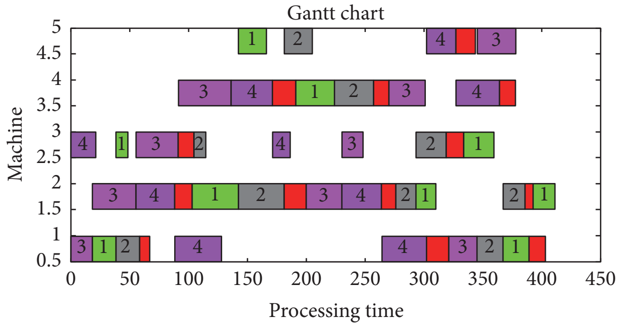


Figure 7: The Gantt chart of independent decision-making.

4.1.2. Independent Decision-Making Strategy. In actual production, the production department is responsible for the production scheduling, and the equipment maintenance department just makes equipment maintenance plan. So, their work is separated. The maintenance department calculates the best maintenance cycle, and the production department calculates the best production scheduling, and then the production department arranges a production plan. In the study, we assume that the production sequence is fixed. And if the machine's cumulative processing time is more than maintenance cycle, the machine needs to be maintained right now. Used by the same algorithm, the optimal solution is 411.5. Its scheduling Gantt chart is shown in Figure 7.

4.2. Discussion of Result. Compared with independent decision-making, the result is improved by 4.62%. From the Gantt chart, we also can get that the maintenance time of joint decision-making is less than of independent decision-making. So, this will decrease the maintenance cost in some ways. In order to prove the superiority of the combined optimization model, we simulate a few more complex numerical experiments, and the simulation data can be checked in Supplementary Materials available online at https://doi.org/10.1155/2017/6758147. The final results are shown in Table 3. On the whole, the joint decision-making model performs better than the independent decision-making model in solving the problem of production scheduling and maintenance plans.

4.3. Evaluation of Hybrid Particle Swarm Optimization Algorithm (HPSOA). Genetic algorithm (GA) is a stochastic search and optimization method which is based on natural selection and genetic mechanism [23]. In recent years, the genetic algorithm had great potential in solving complex optimization problems and it has successful application in the industrial engineering field. In the area of production scheduling, this algorithm is also very important. There are many scholars using this algorithm [24–27].

To verify the validity of HPSOA, we make some tests on different examples compared with GA. In the testing, we make comparison with GA which can also be used to solve this problem. And the results indicated that HPSOA is better than GA to solve this problem.

At first, we analyze the example of 4 jobs and 5 machines we talked about above. The final result is 403.5 which is worse than HPSOA. Its scheduling Gantt chart is shown in Figure 8.

The machines' reentrant times will affect dispatching, so reasonable arrangements for these machines play a big role in improving production efficiency. So, we give two more complex examples to verify the superiority of our method by increasing reentrancy times. One is 5 jobs/7 machines with 3 times reentrancy. The other is 5 jobs/7 machines with 5 times reentrancy. They are all solved using GA and HPSOA. The results are shown in Table 4. Although the running speed of HPSOA is slower than GA, the results are better especially with the number of operations and reentrancy times increasing. So, HPSOA is preferable to be used to solve our problems.

5. Conclusions and Future Research

In actual production, production scheduling and preventive maintenance are closely linked and interactive. In this paper, we build an optimal model of reentrant manufacturing scheduling and equipment maintenance. We adopt fixed maintenance cycle with penalty function strategy and solve the problem effectively by the hybrid particle swarm optimization algorithm. By comparison with GA, this method is preferable to solve the combined optimization problem.

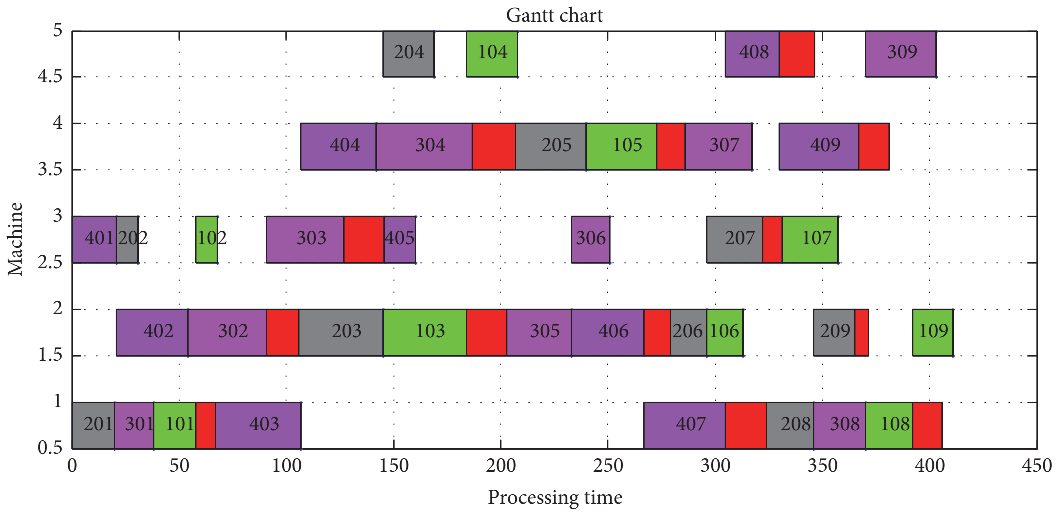


Figure 8: Job processing and machine maintenance by GA.

Table 4: Comparison of GA and HPSOA for complex problems.

Jobs	Machines	Reentrancy times	GA	HPSOA	Improvement
4	5	3	403.5	390.5	3.22%
4	5	5	700.5	646	7.78%
5	7	3	711.5	650.5	8.57%
5	7	5	1065.5	876.5	18.69%

We make lots of assumptions to simplify the complex problem for our study. However, they will not exist in practical production. Firstly, equipment failure is difficult to avoid, so the machines' minor repair time should not be ignored. What is more, equipment age is changeable after maintenance in reality. At last, we focus on only one single objective in the study. However, cost is an important factor to consider which includes production cost, preventive maintenance cost, minimal repair cost for unexpected failures, and tardiness cost. As a consequence, future efforts are still needed to solve this composite problem in reentrant Job Shop production scheduling.

Conflicts of Interest

The authors declare that they have no competing interests.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (no. 71472125).

References

[1] P. R. Kumar, "Re-entrant lines," *Queueing Systems. Theory and Applications*, vol. 13, no. 1-3, pp. 87–110, 1993.

[2] O. B. Senties, C. Azzaro-Pantel, L. Pibouleau, and S. Domenech, "Multi-objective scheduling for semiconductor manufacturing plants," *Computers and Chemical Engineering*, vol. 34, no. 4, pp. 555–566, 2010.

[3] S. Topaloglu and G. Kilincli, "A modified shifting bottleneck heuristic for the reentrant job shop scheduling problem with makespan minimization," *International Journal of Advanced Manufacturing Technology*, vol. 44, no. 7-8, pp. 781–794, 2009.

[4] J. Zoghby, J. W. Barnes, and J. J. Hasenbein, "Modeling the reentrant job shop scheduling problem with setups for metaheuristic searches," *European Journal of Operational Research*, vol. 167, no. 2, pp. 336–348, 2005.

[5] L. Danping and C. K. M. Lee, "A review of the research methodology for the re-entrant scheduling problem," *International Journal of Production Research*, vol. 49, no. 8, pp. 2221–2242, 2011.

[6] J.-Y. Bang and Y.-D. Kim, "Hierarchical production planning for semiconductor wafer fabrication based on linear programming and discrete-event simulation," *IEEE Transactions on Automation Science and Engineering*, vol. 7, no. 2, pp. 326–336, 2010.

[7] A. K. Gupta and A. I. Sivakumar, "Job shop scheduling techniques in semiconductor manufacturing," *International Journal of Advanced Manufacturing Technology*, vol. 27, no. 11, pp. 1163–1169, 2006.

[8] L. Mönch, R. Schabacker, D. Pabst, and J. W. Fowler, "Genetic algorithm-based subproblem solution procedures for a modified shifting bottleneck heuristic for complex job shops," *European Journal of Operational Research*, vol. 177, no. 3, pp. 2100–2118, 2007.

[9] H. M. Ma, S. H. L. Xu, and Ch. M. Ye, "Maintenance of semiconductor manufacturing equipment based on Particle Swarm algorithm," *Industrial Engineering and Management*, vol. 6, pp. 66–69, 2008.

[10] S. A. Mosley, T. Teyner, and R. M. Uzsoy, "Maintenance scheduling and staffing policies in a wafer fabrication facility,"

IEEE Transactions on Semiconductor Manufacturing, vol. 11, no. 2, pp. 316–323, 1998.

[11] A. Davenport, "Integrated maintenance scheduling for semiconductor manufacturing," in *Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems*, A. Lodi, M. Milano, and P. Toth, Eds., vol. 6140 of *Lecture Notes in Computer Science*, pp. 92–96, Springer, Berlin, Germany, 2010.

[12] X. D. Yao, E. Fernández-Gaucherand, M. C. Fu, and S. I. Marcus, "Optimal preventive maintenance scheduling in semiconductor manufacturing," *IEEE Transactions on Semiconductor Manufacturing*, vol. 17, no. 3, pp. 345–356, 2004.

[13] C. R. Cassady and E. Kutanoglu, "Integrating preventive maintenance planning and production scheduling for a single machine," *IEEE Transactions on Reliability*, vol. 54, no. 2, pp. 304–309, 2005.

[14] J. M. Da and L. Y. Wang, "Integrating preventive maintenance planning and production schedulingunder the flow shop," *Machinery*, vol. 50, no. 575, pp. 86–88, 2012.

[15] J. Da, L. Wang, S. Xu, and S. Zhao, "The joint optimization of the batch scheduling and the maintenance under the Flow Shop," *Science Technology and Engineering*, vol. 13, no. 10, pp. 2700–2703, 2013.

[16] T. Kaihara, N. Fujii, A. Tsujibe, and Y. Nonaka, "Proactive maintenance scheduling in a re-entrant flow shop using Lagrangian decomposition coordination method," *CIRP Annals—Manufacturing Technology*, vol. 59, no. 1, pp. 453–456, 2010.

[17] A. Berrichi, F. Yalaoui, L. Amodeo, and M. Mezghiche, "Bi-Objective Ant Colony Optimization approach to optimize production and maintenance scheduling," *Computers and Operations Research*, vol. 37, no. 9, pp. 1584–1596, 2010.

[18] E. Moradi, S. M. T. Fatemi Ghomi, and M. Zandieh, "Bi-objective optimization research on integrated fixed time interval preventive maintenance and production for scheduling flexible job-shop problem," *Expert Systems with Applications*, vol. 38, no. 6, pp. 7169–7178, 2011.

[19] A. Berrichi and F. Yalaoui, "Bi objective artificial immune algorithms to the joint production scheduling and maintenance planning," in *Proceedings of the International Conference on Control, Decision and Information Technologies (CoDIT '13)*, pp. 810–814, IEEE, Hammamet, Tunisia, May 2013.

[20] O. Roux, D. Duvivier, G. Quesnel, and E. Ramat, "Optimization of preventive maintenance through a combined maintenance-production simulation model," *International Journal of Production Economics*, vol. 143, no. 1, pp. 3–12, 2013.

[21] S. Feng, W. Hui, Y. Lei, and H. Fei, *30 Cases Analysis by Intelligent Algorithm with MATLAB*, Beijing University of Aeronautics and Astronautics Press, Beijing, China, 2011.

[22] M. A. Kubzin and V. A. Strusevich, "Planning machine maintenance in two-machine shop scheduling," *Operations Research*, vol. 54, no. 4, pp. 789–800, 2006.

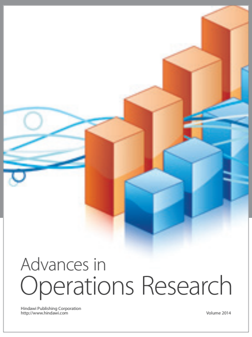
[23] G. Ji, "A survey of Genetic Algorithm," *Computer Applications and Software*, vol. 21, no. 2, pp. 69–73, 2004.

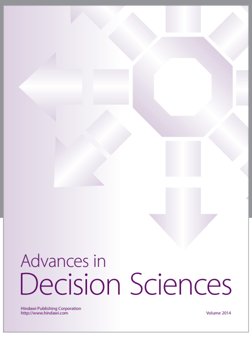
[24] W. Lu and Y. Dang, "Scheduling re-entrant lines based on GA and integrated rules," *Computer Engineering*, vol. 31, no. 13, pp. 186–188, 2005.

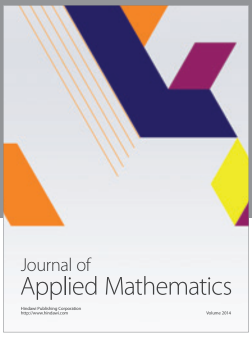
[25] W.-Y. Lu and Y.-Z. Dang, "Scheduling Re-entrant lines based on GA and integrated rules," *Journal of System Simulation*, vol. 17, no. 4, pp. 993–996, 2005.

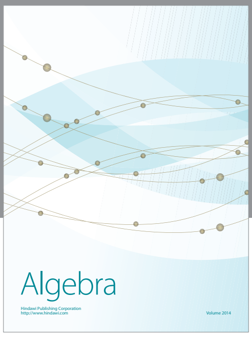
[26] Y. Wang, Z. Wu, and S. Yi, "Scheduling re-entrant manufacturing lines of semiconductor based on GA," *Computer Simulation*, vol. 24, no. 12, pp. 247–252, 2007.

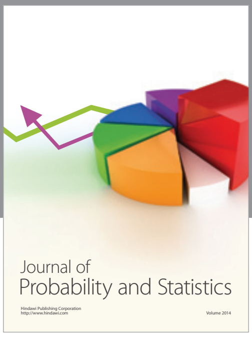
[27] B. Feng, F. Qiao, and J. Wang, "Research on scheduling approach for semiconductor wafer fab based on genetic algorithms," *Computer Engineering*, vol. 31, no. 13, pp. 145–148, 2005.


Advances in
Operations Research


Advances in
Decision Sciences

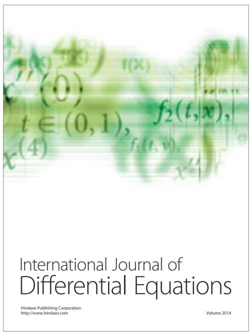

Journal of
Applied Mathematics

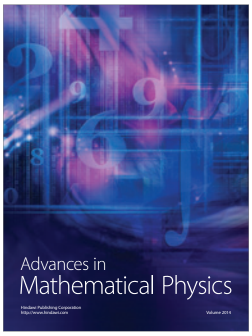

Algebra

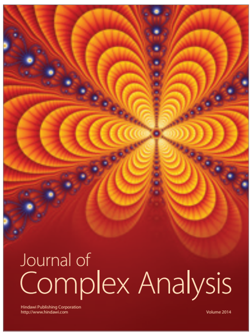

Journal of
Probability and Statistics

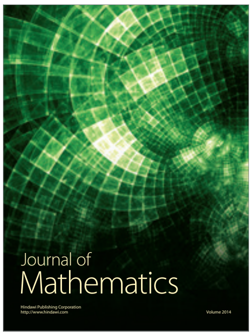
The Scientific
World Journal

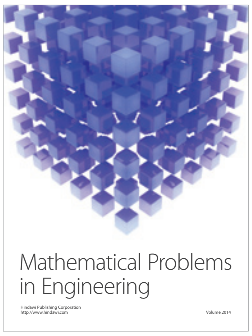
International Journal of
Combinatorics

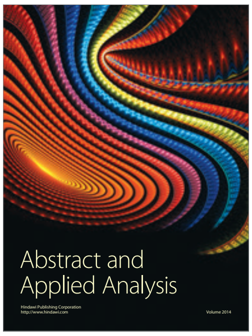

International Journal of
Differential Equations

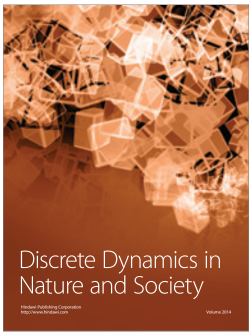

Advances in
Mathematical Physics

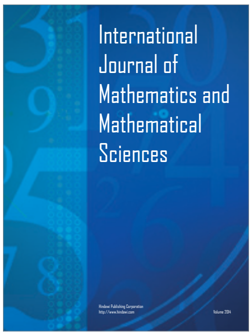

Journal of
Complex Analysis


Journal of
Mathematics


Mathematical Problems
in Engineering

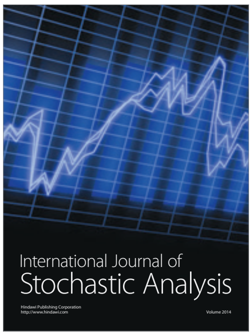

Abstract and
Applied Analysis


Discrete Dynamics in
Nature and Society


International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Function Spaces


International Journal of
Stochastic Analysis

Journal of
Optimization