

Juan LIU, Fei QIAO, Yumin MA, Weichang KONG

Novel slack-based robust scheduling rule for a semiconductor manufacturing system with uncertain processing time

© The Author(s) 2018. Published by Higher Education Press. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0>)

Abstract The NP-hard scheduling problems of semiconductor manufacturing systems (SMSs) are further complicated by stochastic uncertainties. Reactive scheduling is a common dynamic scheduling approach where the scheduling scheme is refreshed in response to real-time uncertainties. The scheduling scheme is overly sensitive to the emergence of uncertainties because the optimization of performance (such as minimum make-span) and the system robustness cannot be achieved simultaneously by conventional reactive scheduling methods. To improve the robustness of the scheduling scheme, we propose a novel slack-based robust scheduling rule (SR) based on the analysis of robustness measurement for SMS with uncertain processing time. The decision in the SR is made in real time given the robustness. The proposed SR is verified under different scenarios, and the results are compared with the existing heuristic rules. Simulation results show that the proposed SR can effectively improve the robustness of the scheduling scheme with a slight performance loss.

Keywords semiconductor manufacturing system, uncertain processing time, dynamic scheduling, slack-based robust scheduling rule

1 Introduction

The research on production scheduling started in the 1950s with the development of operational research (Pinedo and Hadavi, 1992). Most conventional scheduling methods

only consider the deterministic environment and emphasize the precise modeling of scheduling problems. However, real manufacturing systems, which can be modeled by Petri nets (Murata, 1989; Wang et al., 2017), are usually in such a dynamic and indeterministic environment where several types of uncertainties may disturb the execution of production schedules. Examples of such uncertainties include machine failure, rush order arrival, due-time change, uncertain processing time, and so on. New technology revolutions, such as Internet of Things and Big Data, give rise to the development of smart manufacturing (Zhang et al., 2017). Smart factories, as a key approach to smart manufacturing, are now in a dynamic, uncertain, and unpredictable environment both internally and externally.

Uncertainties that affect the production scheduling can be divided into two categories:

(1) In the internal environment:

- Resource-related: Machine failure, load limitation, tool unavailable or damaged, delayed material arrival, material defect, etc.

- Job-related: Rush order arrival, due-time change, uncertain processing times, etc.

(2) In the external environment: Product demand, product price, energy, raw materials supply, etc.

In response to these challenges brought by uncertainties, the anti-interference and quick response capability of the scheduling scheme must be improved, and an efficient production scheduling system must be established. The scheduling problem in the presence of uncertain and dynamic environment is referred to as dynamic scheduling.

A semiconductor manufacturing system (SMS) is a complicated manufacturing system characterized by re-entrant flows, uncertain production interruption, urgent orders, rework, scrap, etc. (Chiang, 2013). The research on semiconductor manufacturing scheduling problems started in 1988 (Wein, 1988). This study assesses the influence of scheduling on the performance of an SMS. The concept of the SMS as a re-entrant manufacturing system was

Received May 3, 2018; accepted July 14, 2018

Juan LIU, Fei QIAO (✉), Yumin MA, Weichang KONG
College of Electronics and Information Engineering, Tongji University,
Shanghai 201804, China
E-mail: fqiao@tongji.edu.cn

This work was funded by the National Natural Science Foundation of China (Grant NO: 71690234)

proposed by Kumar (1993). Since then, vast amounts of research efforts have been devoted to developing scheduling approaches for SMS (Yugma et al., 2015). In most research efforts, the production parameters were assumed to be deterministic in SMS scheduling. Few studies investigated semiconductor manufacturing scheduling in an indeterministic environment (Jamrus et al., 2017). Regardless of the business background, dynamic scheduling has three approaches: Completely reactive, predictive–reactive, and robust proactive (Ouelhadj and Petrovic, 2009).

In completely reactive scheduling, no decided schedule is available in advance, and the decisions are made in real time. Heuristic SR is a completely reactive dynamic scheduling method where the order in which the operation is processed is determined by the priority of jobs. The job with the highest priority is selected to be processed next from a set of jobs. The priority of a job is determined based on job and machine attributes. Therefore, heuristic scheduling rules are responsive to the real-time state of the production line. However, the existing heuristic scheduling rules aim to optimize performance indices, such as cycle time and delivery time. Owing to the complexity of SMSs, scheduling based on heuristic rules is commonly used to solve scheduling problems (Chiang and Fu, 2012).

Predictive–reactive scheduling refers to the decision making after unexpected events or disturbances occur. The reaction generally takes the form of either repairing the existing schedule or generating a completely new one (Vieira et al., 2003). For semiconductor production scheduling problems, some researchers have been studying reactive scheduling. The rescheduling framework of SMSs is presented as a series of layered scheduling strategies with an optimization rescheduling decision mechanism (Zhang et al., 2014). The rescheduling decision model based on the fuzzy neural network can choose an optimized rescheduling strategy according to current system states. The rescheduling method can hardly ensure the global performance indices of the system.

Robust proactive scheduling considers uncertainties when designing the initial schedules. Such scheduling method also considers future disturbances during the generation of initial schedules. This type of approach tries to produce a schedule or a family of schedules that are relatively insensitive to uncertain factors (Kouvelis et al., 2000). Aydilek et al. (2015) studied the two-machine flow shop scheduling problem where processing and setup times are uncertain and bounded to minimize makespan. Fuzzy theory is generally used in such approaches. Uncertain factors are described as scenarios, whereas the uncertain scheduling problem is approximated by several certain problems. In addition, robust scheduling models are developed based on scenario planning approaches. Feng, Choi, and Chung (2016) considered the min–max regret version of a single-machine scheduling problem with

uncertain processing time to determine which job to be processed by outsourcing.

After analysis of the current research status, the current research methods have the following limitations:

(1) For completely reactive scheduling, the order in which the operation is processed is determined by the priority of jobs, and decisions are made in real time. Therefore, heuristic scheduling rules are responsive to the real-time state of the production line. In addition, they are quick, concise, and easy to implement relative to other intelligent algorithms. However, the existing heuristic scheduling rules only aim to optimize performance indices, such as cycle time and delivery time.

(2) Predictive–reactive scheduling is designed to adjust the scheduling scheme to the current production line state when disturbances occur. However, the target of scheduling adjustment is still the optimal or approximate optimal performance index, and the anti-interference capability of the scheduling scheme is not considered. The adjusted scheduling scheme is not a robust one. When new disturbance occurs, the scheduling scheme must be further adjusted. Frequent scheduling changes can cause instability in production lines, which may result in unnecessary system cost.

(3) Robust proactive scheduling is designed to generate a robust initial scheduling scheme with certain endurance to disturbances. The initial scheduling scheme is static, and it cannot be adjusted in real time during production execution. The scheduling scheme will fail if the incoming disturbance exceeds its anti-interference capability.

Therefore, this paper proposes a novel real-time scheduling scheme that also considers robustness. The rest of this paper is organized as follows. Section 2 describes and analyzes the problem. Section 3 discusses the existing robustness measure methods and proposes a new robust SR based on the analysis. Section 4 provides case studies and discusses the results. Finally, Section 5 concludes the paper and identifies the possible directions for future research.

2 Problem formulation

In this section, the dynamic scheduling problem for the semiconductor system is introduced. The jobs are placed into the production line according to a certain release strategy. Each job i has j operations. Denotes the operation j of job i . $p_{i,j}^*$ is the expected processing time of operation j for job i . These operations are carried out according to a predefined route named the scheduling scheme. In practice, the actual processing time $p_{i,j}$ in the manufacturing system deviates from the expected value. The following assumptions are made about dynamic scheduling for SMS:

- Constant release interval is the release policy. Not all the jobs are available at the initial time.
- Machines are re-entrant.

- Each machine is available at the initial time and never breaks down.
- Each machine cannot simultaneously process more than one job.
- Processing times of operations are uncertain, and their expected values are given.
- Operations of a job are independent of each other.
- Operations of a job are independent of those of other jobs.
- No semi-finished production is scrapped.
- No job needs to be reworked.

Cycle time (CT), on-time delivery rate (ODR), and movement (MOV) are some of the scheduling objectives considered in the SMS. However, scheduling schemes that are established based on optimizing performance indices are overly sensitive to uncertainties. In the presence of uncertainty, the performance loss using the scheduling scheme under uncertainty should be measured. The performance loss is defined as the deviation between the expected value of the performance index in deterministic environment and the value of the actual performance index with disturbances:

$$\delta(S) = P(S) - P_0(S), \tag{1}$$

where $P_0(S)$ is the value of the expected performance index for scheduling scheme S and $P(S)$ is the value of the actual performance index.

To avoid the dimension problem caused by different types of performance indices, we represent the scheduling loss using the relative deviation:

$$\delta'(S) = \frac{P(S) - P_0(S)}{P_0(S)}. \tag{2}$$

The expected value of the scheduling loss can be written as follows:

$$E[\delta'(S)] = \frac{E[P(S) - P_0(S)]}{P_0(S)}, \tag{3}$$

$E[\delta'(S)]$ is applied to indicate the robustness of the scheduling scheme. The smaller the value of $E[\delta'(S)]$ is, the better the robustness of the scheduling scheme is.

According to the analysis above, the scheduling objectives of the dynamic scheduling problem for SMS has two aspects:

$$\begin{aligned} \min f_1 &= P_0(S), \\ \min f_2 &= E[\delta'(S)]. \end{aligned} \tag{4}$$

3 Slack-based robust scheduling rule

3.1 Robustness measure

The calculation of the robustness measure f_2 is difficult.

Scenario-based and slack-based methods are the most common in the literature.

The scenario-based method is often applied to scheduling problems with uncertainties, especially robust proactive scheduling, as shown in Fig. 1. The uncertainty can be represented by a set of scenarios. A scenario is used to express one actual situation. Thus, the robustness measure given in Eq. (3) can be calculated as follows:

$$RM1 = \frac{abs\left(\frac{1}{M} \sum_{m=1}^M P_m(S) - P_0(S)\right)}{P_0(S)}, \tag{5}$$

where M is the scenario count, and $P_m(S)$ is the actual performance index of the scheduling scheme S in the scenario m . For Eq. (5), the scheduling scheme is operated in each scenario, and the corresponding robustness measure of the scheduling scheme can be calculated. Likewise, the robustness measures of all alternative scheduling schemes can be obtained, and the best one is selected. This robustness measurement method relies on the description of the uncertainty by the scenario set. The size of the scenario set M is usually large to express the uncertainty completely. This leads to complex calculation and high time cost. However, the characteristics of a robust scheduling scheme cannot be reflected through the scenario-based robustness calculation method, that is, the mechanism characteristics of the robust processing sequence cannot be summarized from this method.

The slack time between jobs can absorb disturbances and uncertainties in the production process, reduce the influence of uncertainties on the scheduling scheme, and improve the robustness of scheduling. The experiment shows a strong correlation between the scheduling robustness and the slack time between jobs (Leon et al., 1994). Therefore, for the same scheduling problem, slack time can be applied to the qualitative analysis on the robustness of different scheduling schemes. Thus, it can be used to indicate the robustness measurement. Two types of slacks, i.e., total slack and free slack, have been commonly

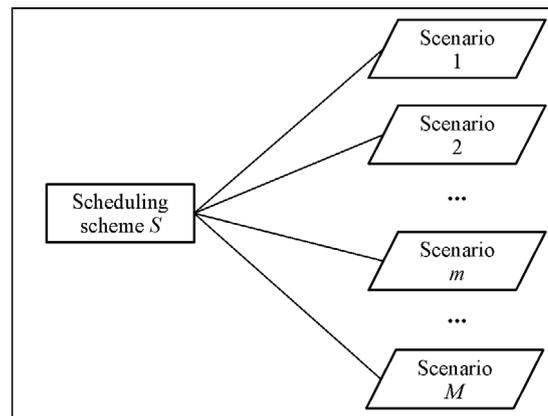


Fig. 1 Scenario-based method

discussed for robust scheduling (Hazır et al., 2010). Total slack is the difference between the earliest start time and the latest start time of an operation. Free slack is the maximum time that an operation can be delayed if the start of the next operation is not delayed. Let us consider an example of the four jobs and four machines given in Table 1. Figure 2 shows an example of slack times.

Table 1 Example of four jobs and four machines

Job	Operation (Processing time)			
	Machine 1	Machine 2	Machine 3	Machine 4
Job 1	$O_{1,1}(6)$	$O_{1,2}(6)$	$O_{1,3}(7)$	$O_{1,4}(5)$
Job 2	$O_{2,2}(4)$	$O_{2,3}(4)$	$O_{2,4}(5)$	$O_{2,1}(4)$
Job 3	$O_{3,3}(3)$	$O_{3,1}(3)$	$O_{3,4}(5)$	$O_{3,2}(9)$
Job 4	$O_{4,4}(6)$	$O_{4,2}(6)$	$O_{4,1}(5)$	$O_{4,3}(5)$

The gray part in the figure represents the free slack time of the operation, FS_{ij} . For example, the free slack time of operation $O_{3,3}$ is 3, whereas operation $O_{4,3}$ does not have free slack time. The total free slack time is used to measure the robustness of the scheduling scheme. This type of robustness measure is denoted as $RM2$, which is calculated as follows:

$$RM2 = \sum_{i=1}^K \sum_{j=1}^{k_i} FS_{ij}, \tag{6}$$

where K indicates the number of jobs, k_i represents the number of operations of the job i , and FS_{ij} represents the free slack time of the operation O_{ij} .

Different from free slack time, total slack time is defined as the difference between the earliest start time and the latest start time of an operation. The operation $O_{4,3}$ in the Fig. 2 is taken as an example. This operation has no free slack time, but it has a total slack time of three. The

robustness measure based on total slack is denoted as $RM3$ and is calculated as follows:

$$RM3 = \frac{\sum_{i=1}^K \sum_{j=1}^{k_i} TS_{ij}}{\sum_{i=1}^K k_i}, \tag{7}$$

where K indicates the number of jobs, k_i represents the number of operations of the job i , and TS_{ij} represents the total slack time of the operation O_{ij} .

3.2 Design of the slack-based robust scheduling rule

Heuristic scheduling rule is a completely reactive dynamic scheduling method. In this scheduling, the order in which the operation is processed is determined by the priority of jobs, and decisions are made in real time. The job with the highest priority is selected to be processed next from a set of jobs. The priority of a job is determined based on job and machine attributes. Therefore, heuristic scheduling rules are responsive to the real-time state of the production line. In addition, they are quick, concise, and easy to implement relative to other intelligent algorithms. However, the existing heuristic scheduling rules aim to optimize performance indices, such as cycle time and delivery time. We analyze the structural characteristics of the robust scheduling scheme (processing sequence) according to the slack-based robustness measurement and propose a new scheduling rule considering robustness called SR.

$$SR_{\mu} = \omega_{\alpha} \cdot \alpha_{\mu} + \omega_{\beta} \cdot \beta_{\mu} + \omega_{\gamma} \cdot \gamma_{\mu}, \tag{8}$$

where SR_{μ} expresses the processing priority of operation O_{μ} in the buffer. Assume l jobs to be processed in the buffer. ω_{α} , ω_{β} , and ω_{γ} are the adjustable coefficients between interval $[0,1]$ used to adjust the proportion of three rules. α_{μ} , β_{μ} , and γ_{μ} are the priorities calculated by the three simple rules.

Simple rule α considers the urgency of the job. First,

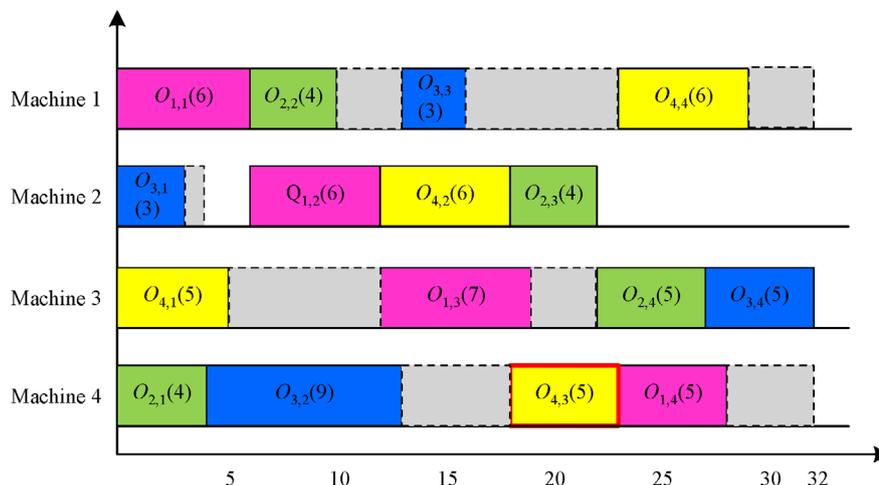


Fig. 2 Example of the slack-based robust measure

whether the remaining processing time of the job exceeds the difference between due time and current time is determined. If $D_\mu - NOW \leq RPT_\mu$, then the job is of the highest priority. If $D_\mu - NOW > RPT_\mu$, then the priority of the job is calculated as follows:

$$\alpha_\mu = \begin{cases} \text{Priority processing, } D_\mu - NOW \leq RPT_\mu \\ \frac{1 + RPT_\mu}{1 + D_\mu - NOW}, D_\mu - NOW > RPT_\mu \end{cases}, \quad (9)$$

where D_μ is the due time of the job μ , NOW is the current time, and RPT_μ is the remaining processing time of the job.

Simple rule β aims to increase the slack time of the operation O_μ .

$$\beta_\mu = \frac{NO_\mu}{\sum_{\mu=1}^l NO_\mu}, \quad (10)$$

where NO_μ means the sum of operation processing times in front of the machine on which the next operation of the job μ will be processed.

Simple rule γ aims to increase the slack time of the previous operation of the job.

$$\gamma_\mu = \frac{WT_\mu}{\sum_{\mu=1}^l WT_\mu}, \quad (11)$$

where WT_μ is the waiting time of job μ in front of the machine.

3.3 Implementation of the proposed SR

The flowchart of the proposed SR is shown in Fig. 3. When the machine becomes available, the priority of every job in the queue of the machine is computed iteratively. Before computing the priority of each job, whether or not it is a hot job is first determined. If yes, the job is selected to be processed on the machine first; otherwise, the priorities based on SR are computed for each job in front of the machine, and the job with the highest priority is selected to be processed on the machine first.

The SR is applied to the example of four jobs and four machines in Section 2. Given no due time constraint, the coefficients in this example are $\omega_\alpha = 0$, $\omega_\beta = 1$, and $\omega_\gamma = 1$. The Gantt chart of the scheduling result of this problem is given in Fig. 4. The robustness measures $RM2$, $RM3$, and makespan are calculated and listed in Table 2.

Table 2 Robustness measures and makespan of the example applying SR

Scheduling scheme	$RM2$	$RM3$	Makespan
Existing one	33	3.56	32
SR	34	4.13	34

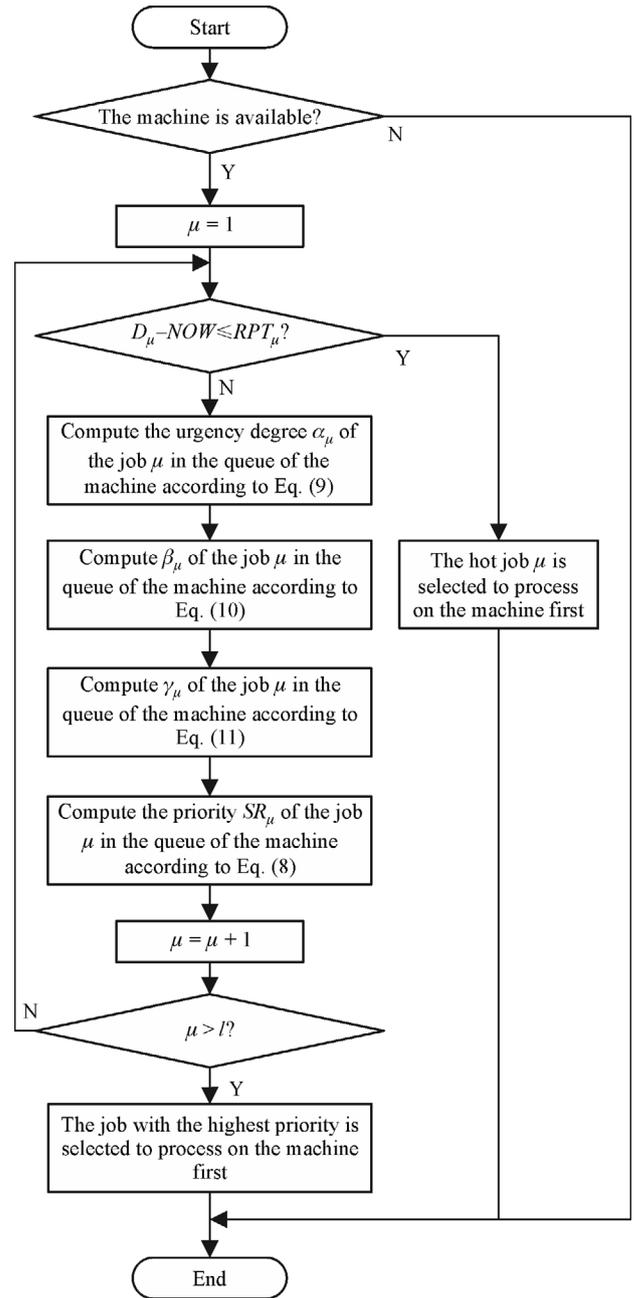


Fig. 3 Flowchart of the proposed SR

The makespan of the scheduling scheme applying the proposed SR is slightly longer than that of the original scheduling scheme because the relative deviation is only 6.25%. Both the free slack time and total slack time are extended. The increase rates of $RM2$ and $RM3$ are 3.03% and 16.01%, respectively. Hence, the proposed SR can improve the robustness of the scheduling scheme with a slight performance loss.

4 Case study

The MiniFab model, which is a simplified yet typical

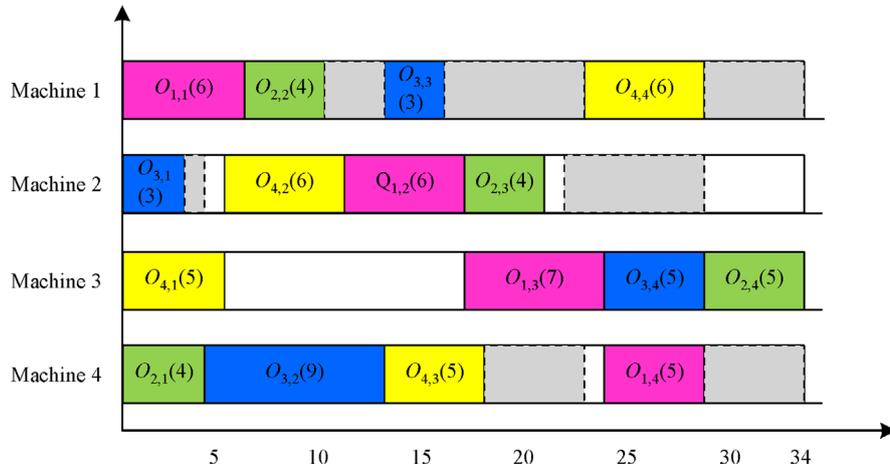


Fig. 4 Gantt chart of the scheduling scheme applying SR

semiconductor production line model that contains five machines and three work areas, is selected as the experimental object. The jobs have the same production flow as shown in Fig. 5. The experiment runs in the computer with Intel Core i7 CPU and a memory of 8.00 GB. Plant Simulation is selected as the simulation platform.

4.1 Design of the experiment

To demonstrate the effectiveness of the proposed slack-based robust SR, we compared it with existing heuristic rules, including Earliest Due Date (EDD), Smallest Remaining Processing Time (SRPT), and Critical Ratio (CR). We selected three groups of coefficients for SR.

The *RM1* is applied to indicate the robustness of the SR owing to the numerous operations in the MiniFab model. The uncertain processing times on *Me* should be considered because the machine *Me* is generally the bottleneck of the system. We assume that the processing time of operation $O_{i,j}$ is any value within the interval $[p_{ij}^L, p_{ij}^H]$, where $p_{ij}^L = p_{ij}^* - \delta, p_{ij}^H = p_{ij}^* + \delta,$ and $\delta = 0.1p_i^*.$ The attainable probability of each scenario is equal. We create 30 scenarios to represent 30 types of actual production conditions.

4.2 Experiment results and discussions

We apply each scheduling rule on the MiniFab model in 30 scenarios for 50 days. The performance indices, including CT, ODR, and MOV, are collected. We also obtain 180 groups of raw data on performance. The scenario-based method is used to calculate the robustness measure *RM1* of each scheduling rule according to Eq. (5), and the results are shown in Table 3. *RM1* indicates the robustness measures of the rules. The better the robustness of the scheduling rule, the smaller the value of *RM1*. Min, Max, and Avg represent the minimum, maximum, and the average of performance indices (CT, ODR, and MOV) in 30 scenarios, respectively.

The results of simulation in various scenarios validate that the proposed SR is superior to the traditional heuristic rules in robustness. For further and clear analysis, the data in the table are illustrated in Figs. 6–8. The charts show that the proposed SR has very good robustness compared with the traditional heuristic rules. The proposed SR method can also guarantee satisfactory production performance.

The results of performance index MOV (Fig. 8) verify the effectiveness of our method. The upper part of the graph is a comparison of robustness, and the lower part is a comparison of performance indices. In the upper line chart, the horizontal coordinate represents SRs, whereas the

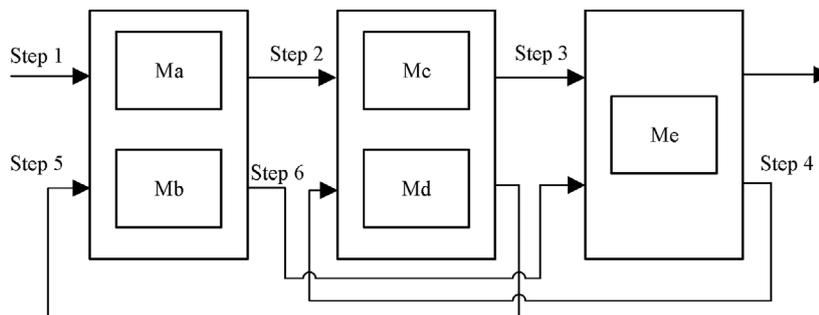


Fig. 5 Production flow of MiniFab model

Table 3 Robustness and performance measure of SR and other heuristic rules

SR	CT				ODR				MOV			
	<i>RM1</i>	Min	Max	Avg	<i>RM1</i>	Min	Max	Avg	<i>RM1</i>	Min	Max	Avg
SR1	0.0016	90.95	124.49	106.44	0.0016	91.71%	92.41%	92.04%	0.0001	5024	5116	5062
SR2	0.0024	44.65	45.58	45.07	0.0012	91.60%	92.16%	91.86%	0.0012	5056	5172	5116
SR3	0.0024	44.65	45.58	45.07	0.0004	90.57%	91.25%	90.98%	0.0011	5057	5174	5117
EDD	0.0025	44.65	45.58	45.08	0.0016	91.69%	92.60%	92.04%	0.0016	5024	5190	5127
SRPT	0.0029	44.54	45.58	45.08	0.002	90.73%	91.82%	91.31%	0.0013	5057	5172	5116
CR	0.005	92.92	124.14	107.37	0.0067	90.50%	91.70%	91.32%	0.0042	4946	5152	5060

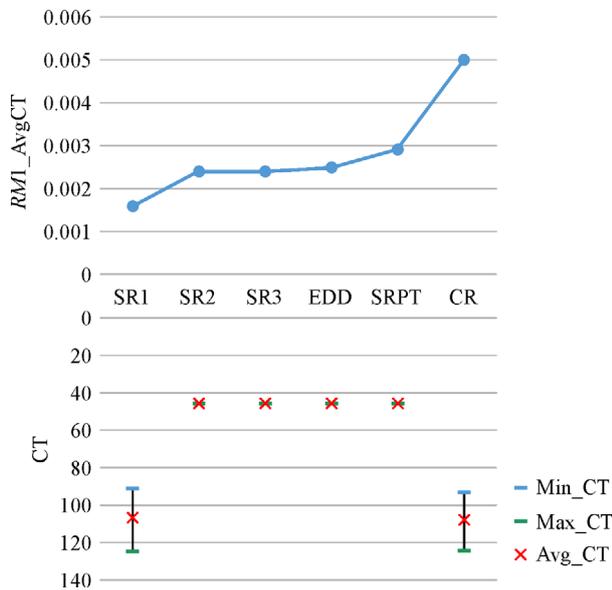


Fig. 6 Robustness and performance measure of SR and other heuristic rules in terms of CT

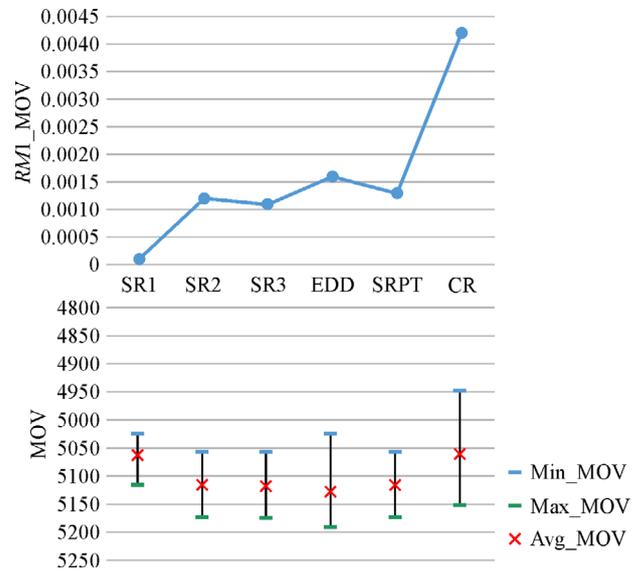


Fig. 8 Robustness and performance measure of SR and other heuristic rules in terms of MOV

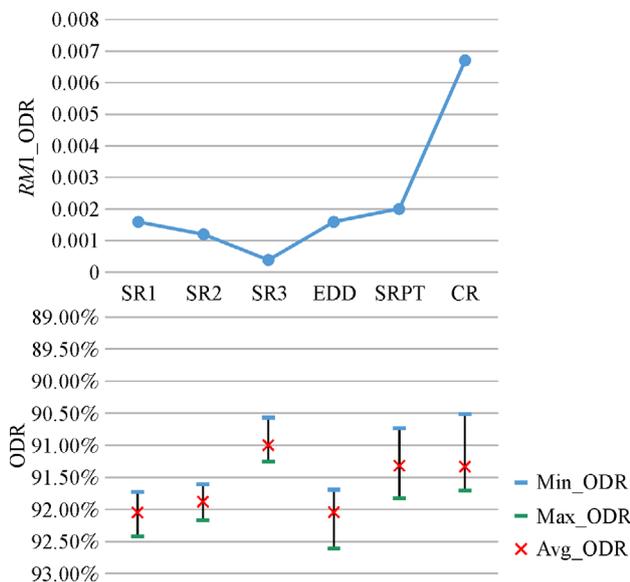


Fig. 7 Robustness and performance measure of SR and other heuristic rules in terms of ODR

vertical coordinate represents the robustness in terms of MOV, which is noted as *RM1_MOV*. Compared with an existing heuristic SR, the SR method proposed in this paper can achieve better robustness in terms of MOV. Therefore, the proposed SR is superior to the three other traditional heuristic rules in terms of robustness.

The fluctuation ranges of the performance index MOV using SRs are very small in the lower part of the graph. The SR1 rule (the performance index MOV is the worst) and the EDD rule (the performance index MOV is the best) are selected for comparison. The relative deviation of MOV of SR1 and EDD is only -1.27% . Hence, the proposed SR can effectively improve the robustness of the scheduling scheme and guarantee the performance optimization.

5 Conclusions

SMS scheduling is confronted with uncertainties in practical production. In this paper, we analyzed the

dynamic scheduling approaches. Basing from the analysis of the robustness measures, we propose a novel slack-based robust SR for an SMS with uncertain processing time. The decision in the SR is made in real time with consideration of robustness. The proposed SR was verified under different scenarios, and the results were compared with the existing heuristic rules. The results illustrate that our SR is superior to other rules in terms of robustness.

SR has three adjustable coefficients. The SR with constant coefficients may be inapplicable when the production environment changes dramatically. How to adjust the coefficient in a more complex and changeable production environment needs further study. Therefore, future research work should focus on the relationship between coefficient and production state, as well as coefficient adjustment strategies.

References

- Aydilek H, Aydilek H, Allahverdi A (2015). Production in a two-machine flowshop scheduling environment with uncertain processing and setup times to minimize makespan. *International Journal of Production Research*, 53(9): 2803–2819
- Chiang T C (2013). Enhancing rule-based scheduling in wafer fabrication facilities by evolutionary algorithms: Review and opportunity. *Computers & Industrial Engineering*, 64(1): 524–535
- Chiang T C, Fu L C (2012). Rule-based scheduling in wafer fabrication with due date-based objectives. *Computers & Operations Research*, 39(11): 2820–2835
- Choi B, Chung K (2016). Min-max regret version of a scheduling problem with outsourcing decisions under processing time uncertainty. *European Journal of Operational Research*, 252(2): 367–375
- Hazır Ö, Haouari M, Erel E (2010). Robust scheduling and robustness measures for the discrete time/cost trade-off problem. *European Journal of Operational Research*, 207(2): 633–643
- Jamrus T, Chien C F, Gen M, Sethanan K (2017). Hybrid particle swarm optimization combined with genetic operators for flexible job-shop scheduling under uncertain processing time for semiconductor manufacturing. *IEEE Transactions on Semiconductor Manufacturing*, 28(3): 353–366
- Kouvelis P, Daniels L R, Vairaktarakis G (2000). Robust scheduling of a two-machine flow shop with uncertain processing times. *IIE Transactions*, 32(5): 421–432
- Kumar P R (1993). Re-entrant lines. *Queueing Systems*, 13(1–3): 87–110
- Leon V J, Wu S D, Storer R H (1994). Robustness measures and robust scheduling for job shops. *IIE Transactions*, 26(5): 32–43
- Murata T (1989). Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, 77(4): 541–580
- Ouelhadj D, Petrovic S (2009). A survey of dynamic scheduling in manufacturing systems. *Journal of Scheduling*, 12(4): 417–431
- Pinedo M, Hadavi K (1992). Scheduling: Theory, algorithms, and systems. *IIE Transactions*, 28(8): 695–697
- Vieira G E, Herrmann J W, Lin E (2003). Rescheduling manufacturing systems: A framework of strategies, policies, and methods. *Journal of Scheduling*, 6(1): 39–62
- Wang S G, You D, Seatzu C (2017). A novel approach for constraint transformation in Petri nets with uncontrollable transitions. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 48(8): 1403–1410
- Wein L M (1988). Scheduling semiconductor wafer fabrication. *IEEE Transactions on Semiconductor Manufacturing*, 1(3): 115–130
- Yugma C, Blue J, Dauzère-Pérès S, Obeid A (2015). Integration of scheduling and advanced process control in semiconductor manufacturing: Review and outlook. *Journal of Scheduling*, 18(2): 195–205
- Zhang J, Ding G, Zou Y, Qin S, Fu J (2017). Review of job shop scheduling research and its new perspectives under industry 4.0. *Journal of Intelligent Manufacturing*, (3): 1–22
- Zhang J, Qin W, Wu L H, Zhai W B (2014). Fuzzy neural network-based rescheduling decision mechanism for semiconductor manufacturing. *Computers in Industry*, 65(8): 1115–1125