 doi:10.20944/preprints201710.0013.v1

Dynamic Scheduling of a Semiconductor Production Line Based on Composite Rule Set

Yumin Ma[1], Fei Qiao[1], Fu Zhao[2,3], John W.Sutherland[3]

[1] School of Electronical & Information Engineering, Tongji University, Shanghai, 201804, China; ymma@tongji.edu.cn (Y.M. Ma), fqiao@tongji.edu.cn(F. Qiao)

[2] School of Mechanical Engineering, Purdue University, West Lafayette, IN 47907, USA; fzhao@purdue.edu(F.Zhao)

[3] Division of Environmental and Ecological Engineering, Purdue University, West Lafayette, IN 47907, USA; jwsuther@purdue.edu (J.W.Sutherland)

* Correspondence: ymma@tongji.edu.cn; Tel.: +86-216-958-8911

Abstract: Various factors and constraints should be considered when developing a manufacturing production schedule, and such a schedule is often based on rules. This paper develops a composite dispatching rule based on heuristic rules that comprehensively consider various factors in a semiconductor production line. The composite rule is obtained by exploring various states of a semiconductor production line (machine status, queue size, etc.), where such indicators as makespan and equipment efficiency are used to judge performance. A model of the response surface, as a function of key variables, is then developed to find the optimized parameters of a composite rule for various production states. Further, dynamic scheduling of semiconductor manufacturing is studied based on support vector regression (SVR). This approach dynamically obtains a composite dispatching rule (i.e. parameters of the composite dispatching rule) that can be used to optimize production performance according to real-time production line state. Following optimization, the proposed dynamic scheduling approach is tested in a real semiconductor production line to validate the effectiveness of the proposed composite rule set.

Keywords: Dynamic Scheduling; Semiconductor Manufacturing; Composite Rule Set; Support Vector Regression (SVR)

---

1. Introduction

A semiconductor manufacturing system is a dynamic system that is subject to various uncertainties (e.g., machine failures, arrival of new urgent jobs, and the modification of job due times). When unexpected events occur, a previously "optimal" schedule may no longer be optimal, and can even become infeasible. Scheduling in response to real-time events has been defined as "dynamic scheduling"[1].

Dynamic scheduling of manufacturing systems is often rule based, with a given rule selected based on the needs of the production environment [2]. Some researchers have been studying dynamic scheduling based on a machine learning approach. With this approach, a system acquires scheduling knowledge through training with optimized scheduling samples. This knowledge is then applied to obtain scheduling rules which may be utilized to obtain a feasible real-time schedule. For example, Shiue et al. [3] proposed a self-organizing map-based multiple scheduling rule selection mechanism. Tsai et al. [4] put forward a radio frequency identification (RFID)-based real-time scheduling system for an automated semiconductor manufacturing plant, which selected features for training samples and established a dynamic scheduling model based on a support vector machine (SVM). Olafsson et al. [5] suggested a dynamic scheduling strategy selection method based on a genetic algorithm (GA) and decision tree. Ma et al. [6] and Qiao et al. [7] used a binary particle swarm optimization combined with a support vector machine (BPSO-SVM) and a k-nearest neighbors (KNN) algorithm to realize dynamic scheduling for a semiconductor manufacturing system. These methods provide simple and effective heuristics for selecting real-time scheduling rules for a manufacturing system. These heuristics tend to have a local perspective, in that they ignore such broader issues as information of manufacturing system, such as job due times and equipment load. However, production scheduling in practice must consider a variety of different performance criteria and constraints, e.g., cost, job completion times, job due dates, and process requirements and limitations. That means, a global information-based dispatching rule is needed, and dynamic scheduling for manufacturing system is implemented by adjusting key parameters of rules. Li et al [8] used a back propagation (BP) neural network, a binary regression model and a particle swarm optimization to study samples, thereby obtaining a self-adapt scheduling model to meet the dynamic scheduling needs; Lee et al [9] used a real-time dispatching approach integrating autonomy and coordination, in which an advanced dispatching rule was determined based on global information. Once trigger events occurred, the parameters of dispatching rules would be adjusted dynamically. The scheduling structure of this approach is keeping stable, but the choice of key parameters is difficult.

Due to the complexity and multiple process constraints of semiconductor production line, if using advanced dispatching rules for scheduling, global information needs to be taken into account and results in computationally demanding; while using simple rules for scheduling, the effectiveness of optimization is not satisfied. Therefore, improved simple rules are suggested to use for semiconductor production scheduling [10-12]. Chen [13] fused earliest due date (EDD) and fluctuation smoothing rule for mean cycle time (FSMCT) into a new scheduling rule in a nonlinear way for optimizing mean cycle time and maximum lateness. Dabbas et al. [14] use a linear combination with relative weights to combine multiple dispatching rules into a single rule. Both of them suggested combining single rules into a composite rule, but how to obtain the parameter (weights) of composite rule in real time according to the state of manufacturing system, was not involved.

This paper proposes a simple and feasible composite dispatching rule and applies it to scheduling of a semiconductor production line for simultaneous optimization of multiple performance measures. The rest of this paper is organized as follows. In section 2 the composite dispatching rule is presented. A framework for a dynamic scheduling algorithm is described in section 3. In section 4, the dynamic scheduling method with the proposed composite dispatching rule is studied in detail. A case study for the production of 5-inch and 6-inch wafers is presented in section 5. Finally, some concluding remarks are presented in Section 6.

2. Composite Dispatching Rule

A simple heuristic dispatching rule is often sought to assess job attributes (due date, process time, etc.) and make decisions to meet some performance targets of a manufacturing system (energy, cost, throughput, etc.). A composite rule considers, and dynamically integrates, several simple dispatching rules to simultaneously optimize multiple objectives. In particular, the rule seeks for the best sequence in which a set of jobs is processed. That is, by applying the composite rule, an integrated priority P_i of job i can be determined using the priority $p_{k,i}$ of job i based on a single rule $R_k (k = 1,2, \dots, K)$, which in turn defines the job sequence.

2.1 Priority based on a single rule

Suppose that job i is in a machine buffer waiting to be processed. When using rule R_k to sort jobs, the priority $p_{k,i}$ of job i is determined by the job attribute related to rule R_k, and $0 \leq p_{k,i} \leq 1$, where the greater the value of $p_{k,i}$, the higher the processing priority for job i. There are two scenarios:

Scenario I: The greater the value of the job attribute, given by α, the higher the job processing priority. For example, the job attribute "waiting time in buffer", is used to determine the job

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 October 2017 doi:10.20944/preprints201710.0013.v1

processing sequence when applying the dispatching rule "first-in first-out (FIFO)". For a job, the longer the waiting time in the buffer, the higher the job processing priority. In this case, the priority is determined by rule R_k :

$$p_{k,i} = \frac{\alpha_i - \alpha_{min}}{\alpha_{max} - \alpha_{min}} , \quad (1)$$

Scenario II: The smaller the value of the job attribute (α), the higher the job processing priority. For example, the job attribute "due date", is used to determine the job processing sequence when applying the dispatching rule "earliest due date (EDD)". For a job, the earlier the job's due date, the higher the processing priority. In this case the priority is determined by rule R_k :

$$p_{k,i} = 1 - \frac{\alpha_i - \alpha_{min}}{\alpha_{max} - \alpha_{min}} , \quad (2)$$

here α_i is the value of attribute α of job i, and α_{max} and α_{min} are the maximum and the minimum values of attribute α of jobs waiting to be processed.

2.2 Integrated priority based on a composite rule

Integrated priority, as determined by a composite rule, is defined as follows: a composite rule is a linear combination of two or more single rules, with each rule having an associated weight. Suppose $\omega_k (k = 1,2, \dots, K)$ is the weight for rule R_k in the composite rule. Then, the integrated priority of job i is:

$$P_i = \omega_1 * p_{1,i} + \omega_2 * p_{2,i} + \cdots + \omega_K * p_{K,i} = \sum_{k=1}^{K} \omega_k * p_{k,i} , \quad (3)$$

where K is the number of single rules in the composite rule and $\sum_{k=1}^{K} \omega_k = 1$, $0 \le \omega_k \le 1$. When applying a composite rule to scheduling, the integrated priority P_i of job i waiting for processing is determined according to Eq. (3). The greater the integrated priority, the earlier job i is to be processed. Changing the weights in Equation (3) will lead to different integrated priority, thus different job sequence. In an application, manufacturing performance can be improved by optimizing the weights in a composite rule.

3. Learning Based Dynamic Scheduling

The proposed approach to solve dynamic scheduling problems follows these steps: i) analyze historical data on production state, scheduling decisions, and resulting performance through machine learning, and ii) build a model that uses the machine learning results to find the best scheduling decision for a given production state and scheduling objectives. The framework of the proposed learning-based dynamic scheduling method is shown in Fig. 1.

The framework can be divided into three modules. The modules are a) a sample generation module which creates sample production states, and finds the best decision for each performance criterion of interest, b) an offline learning (or training) module that uses the sample data to develop a scheduling library i.e. set of scheduling models, with each scheduling model giving optimal scheduling decisions based on system state for a specified scheduling objective, and c) an online module that uses the scheduling model for decision-making. Since historical data only provide manufacturing system performance for the scheduling decision taken, to develop data that can be used for training purposes, a simulation model is required to predict system performance when alternative decisions are implemented. Here a discrete event simulation model is used, which is based on the actual configuration and behavior of a semiconductor production line. Historical data from an actual line on job sizes, arrival times, machine breakdowns, etc. were described statistically, and used to characterize key simulation inputs. For each simulation trial, all possible decisions were evaluated, and values for the performance criteria of interest were noted. Thus, for every production state, the performance evaluation for every performance criterion is available for every decision.

The offline learning module builds a scheduling model for each scheduling objective based on training data, where each dataset includes production states and the corresponding best decision for the specified objective, and can be obtained by exercising the simulation model. The offline learning

doi:10.20944/preprints201710.0013.v1

module can greatly reduce the time consumed for scheduling. The online scheduling module selects a scheduling model from the scheduling model library according to the scheduling requirements of users, and outputs an optimal scheduling decision by inputting real-time state data from the semiconductor production line.

A data record in the sample base consists of the production line state (*S*), scheduling decision (*D*), and performance (*P*), given as $\{S, D, P\}$. S represents the current state of the production line, working area, machines and jobs obtained from historical data; D is the composite scheduling rule applied, and P is the recorded performance of the given production line found by applying the scheduling decision and running the simulation model for a scheduling period.

For the development of the discrete event simulation model for semiconductor production system, please refer to Ye [15]. The discussion here is focusing on the 2nd module i.e. offline learning.

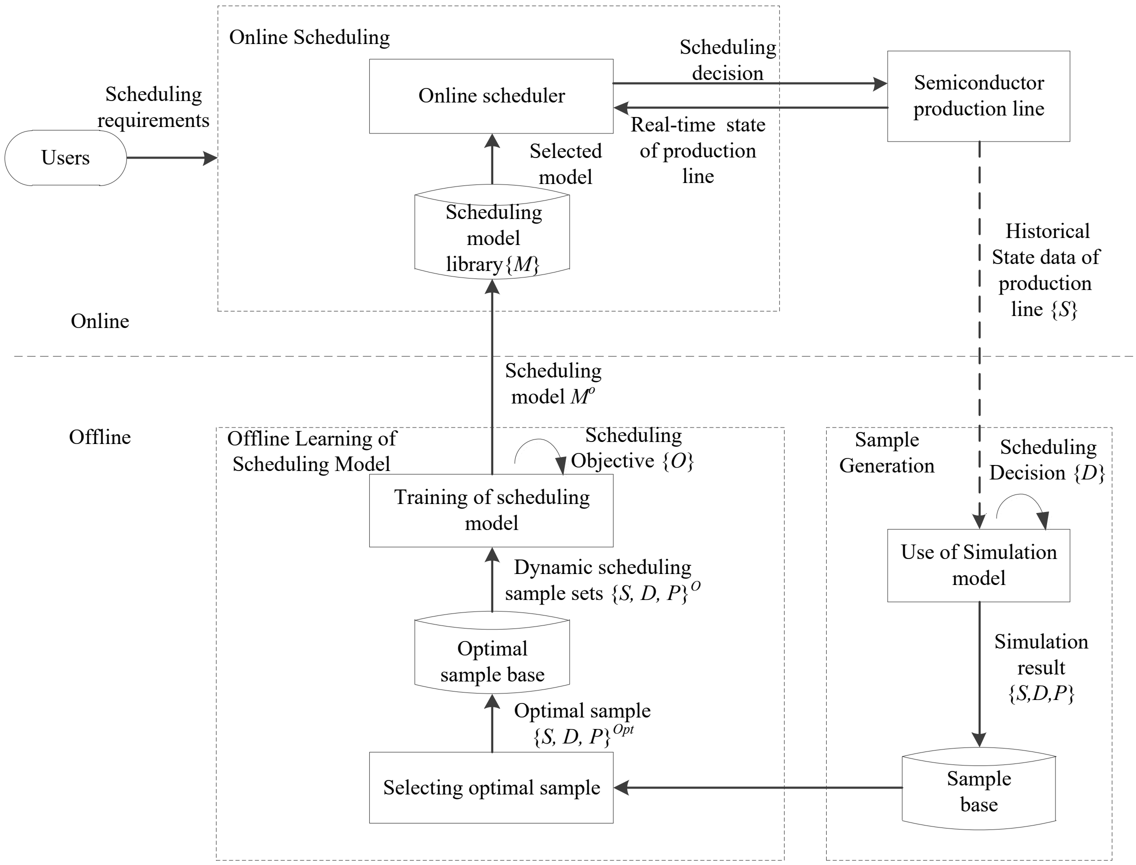

Figure 1. The framework of a dynamic scheduling system for a semiconductor production line

4. A SVR-Based Dynamic Scheduling Model

4.1 Generation of training data from sample base

Since running the discrete event simulation model is time consuming (e.g. each simulation run takes more than 30 minutes for a production line with more than 200 steps and 800 machines when processing 80000 wafers), it is infeasible to search for optimal scheduling decision (i.e. weights in a composite rule) using the search algorithm for a production state. To address this, response surface methodology (RSM) is used here. If a model for the response surface exists as a function of the weights/parameters, values may be selected to optimize the composite rule. Such an approach involves three steps: i) running trials of a "process" that depends on several variables (securing "experimental" data), ii) statistical modeling of the experimental data to secure a predicted response surface [16], and iii) using the predicted response surface to select variable settings that optimize a response. Here the weights, $\omega_i (i = 1,2, \ldots, k)$, are the variables of interest and the response is a multi-objective measure corresponding to scheduling objective. It is desired to find the levels of the weights that optimize the response, Y. A number of trials are performed using different weights

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 October 2017 doi:10.20944/preprints201710.0013.v1

(variable settings) for each production state. The performance of the composite rule is evaluated for each trial to obtain a response. A second-order model (shown in Eq. (4)) is then developed for the response as a function of the variables/weights:

$$\hat{Y} = \hat{\beta}_0 + \sum_{i=1}^{k} \hat{\beta}_i \omega_i + \sum_{i=1}^{k} \sum_{j\geq i} \hat{\beta}_{ij} \omega_i \omega_j \;, \tag{4}$$

where $\hat{\beta}$ ($\hat{\beta}_0$, β_i and $\hat{\beta}_{ij}$) are estimated parameters. Once the experimental data have been obtained, the form shown in Eq. (4) is fit to the data to obtain the predicted response surface. Then, the combination of weight values, $\omega_i^*(i = 1,2,\ldots,k)$, that optimize production performance may be obtained via calculus from Eq. (4). This set of weight values provides the optimal composite scheduling rule. In this paper, we use design expert software to find the optimal weight for a production state, and then build the optimal ample base.

4.2 Development of scheduling models

A scheduling decision here is a composite scheduling rule and can be represented by the weights of simple heuristic scheduling rules ($w_i \in [0,1]$ and $w_i \in R$). The scheduling model needs to determine the weights according to the production line state, it is different to those that need to select a scheduling rule from a defined rule set. The scheduling problem then becomes a regression problem as the training datasets cannot cover all the possible production line states. Support vector regression (SVR) is used to build the scheduling model due to its high regression accuracy and its high generalization ability, even when used for problems with a small sample size. Assuming there is a sample set $\{(x_i, y_i)|x_i \in R^m, y_i \in R^n, i = 1,2,\ldots,N\}$, the nonlinear mapping, $\Phi(x)$, of input, x, is built and then the regression function is generated as follows:

$$f(x_i) = \omega^* \cdot \Phi(x_i) + b \;, \tag{5}$$

where ω^* and b are the weight vector and bias or offset, respectively. The quadratic program is used to solve the problem and minimize the loss function as shown in the Eq. (6) [17].

$$L(f(x_i) - y_i) = \begin{cases} 0, & |f(x_i) - y_i| \leq \varepsilon \\ |f(x_i) - y_i| - \varepsilon, & |f(x_i) - y_i| \geq \varepsilon \end{cases}, \tag{6}$$

we can obtain the optimal Lagrange multipliers α_i and α_i^* , then acquire the linear regression function in a high-dimensional space, as shown in Eq. (7), where $K(x_i, x)$ is the kernel function. Campbell et al.[18] provide more detail on the SVR method.

$$f(x) = \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) K(x_i, x) + b \;, \tag{7}$$

For the scheduling of a semiconductor production line, the input vector $x_i = (x_{i,1}, x_{i,2}, \ldots, x_{i,m})^T$ is a set of production feature values that describe production state; the output vector $y_i = (y_{i,1}, y_{i,2}, \ldots, y_{i,n})^T$ is a set of rule weights of a given scheduling decision in Eq. (3). Based on a sample set $\{(x_i, y_i)|x_i \in R^m, y_i \in R^n, i = 1,2,\ldots,N\}$, a regression function can be obtained as shown in Eq. (7). So, a composite scheduling rule can be represented by vector $f(x)$ for any given production state, x.

Given the background provided above for developing a regression model for the scheduling decision, attention now shifts to outlining the stepwise procedure for creating a dynamic scheduling model. There are four steps to build a dynamic scheduling model of a semiconductor production line using SVR.

Step 1: Normalizing sample data. For one production feature, $x_{i,j}$ of j -th element of x -th data record for example, the normalized equation is shown as follows:

$$x_{i,j}^N = \frac{x_{i,j} - x_{i,j}^{min}}{x_{i,j}^{max} - x_{i,j}^{min}} \;, \tag{8}$$

where $x_{i,j}^{max}$ and $x_{i,j}^{min}$ are the maximum and minimum values of $x_{i,j}(j = 1,\cdots,m)$ in the sample set.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 October 2017 doi:10.20944/preprints201710.0013.v1

Step 2: Creating the training sample set and the test sample set. There is a total of N optimal samples in the sample set, from which we randomly build training set TE_1 and test set TE_2, respectively accounting for 4/5 and 1/5 of the total sample.

Step 3: Training a SVR based scheduling model. Use training set TE_1 to train a SVR based scheduling model with the radial basis function (RBF) kernel, which is $K(x_i, x) = exp(-\gamma\|x - x_i\|^2)$. The penalty factor C and the variance γ of the kernel function are selected to achieve the highest regression accuracy of the model through cross-validation. Based on this, a SVR based scheduling model is created. When the performance of several SVR models is the same or similar, the one with the smallest C value is chosen to reduce the complexity of the algorithm.

Step 4: Evaluating the model. The created model is evaluated with test set TE_2. If the prediction accuracy is in the error range defined based on experience, the model is the one needed, otherwise, return to step 3 and retrain the scheduling model.

Once the scheduling model is established, the focus may shift to evaluating the performance of the model, and there are many ways to evaluate the accuracy of the created scheduling model. Here, mean square error (MSE) is used to evaluate the mean error of the scheduling model, which is acquired through Eq. (9):

$$\mathrm{MSE} = \sqrt{\frac{1}{L}\sum_{i=1}^{L}(\hat{t_i} - t_i)} \ , \tag{9}$$

where L is the number of the samples in test set TE_2, $\hat{t_i}$ is the predicted weight value and t_i is the real weight value.

5 Case Study

The proposed method using optimized composite rules is tested on a real semiconductor production line, which produces 5-inch and 6-inch wafers in Shanghai. There are more than 800 machines, and the average amount of WIP (work in process) is up to 80,000 pieces in the line. With the help of a self-developed scheduling simulation system (FabSimSys, software copyright number from China: 2011SR066503) and expert design v8.0 software, this paper uses the real line production data to obtain sample data.

5.1 Selection of experimental data set

5.1.1 Production features set

Following the work of the Ma's work[19], 67 production features were selected for analysis and study. One feature selected was the amount of WIP (number of work in process) and others are distribution of machine number and bottleneck machine number. Utilizing these features, it is possible to describe the state of the both the jobs and the machines for every location in the production line.

5.1.2 Design of composite rule

Several lot attributes were selected to build the composite rule, and are considered when dispatching lots. Based on industry research, the selected attributes are i) the priority of a lot (Priority), ii) the remaining number of steps in a lot (RemainingStep), and iii) the process time constraint. The process time constraint limits the time between two or more production steps for a lot (Q-Time is a parameter, and if a manufacturing process exceeds it, the lot needs to be reworked or scrapped). These attributes reflect the lot urgency, the degree of completeness, and process constraints. The integrated priority is determined by three attributes. Based on the priorities of the three attributes of lot i ($P_{P,i}, P_{R,i}, P_{Q,i}$) and the weights of the three attributes (ω_P, ω_R and ω_Q), the integrated priority P_i for lot i is calculated (see Eq. (10)). The integrated priority is then used for dispatching the lot.

$$P_i = \omega_P * P_{P,i} + \omega_R * P_{R,i} + \omega_Q * P_{Q,i} \ , \tag{10}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 October 2017 doi:10.20944/preprints201710.0013.v1

5.1.3 Selection of performance indicators

In order to optimize the operation of the semiconductor production line, long-term and short-term performance indicators need to be considered as a whole in the research. Based on the specific application, five performance indicators were selected as the optimization objectives for scheduling: mean cycle time of total lots (MCT), total wafer movement amount (MOV), amount of work in process (WIP), production rate (PR) and overall equipment efficiency (OEE) [20]. Among them, MCT and PR are long-term performance indicators, MOV, WIP and OEE are short-term performance indicators.

5.2 Parameter settings of the experiment

As has been noted, the inputs of the scheduling model are the production features of the semiconductor production line. In order to improve the output accuracy of the model, it is necessary to reduce the number of inputs by reducing the number of production features; this can be achieved by using the genetic algorithm (GA) [19]. The parameters of the genetic algorithm are set as follows: population size is 100, maximum evolution generation is 100 generations, crossover probability is 0.8, and mutation probability is 0.05.

The parameters of the SVR algorithm are set as follows: the maximum and minimum values of the penalty parameter C are C_{max} = 32 and $C_{min} = 0$; The maximum and minimum values of the variance parameter γ in the kernel function are γ_{max} = 32 and $\gamma_{min} = 0$.

5.3 Experiment results and data analysis

Following the application of the genetic algorithm to reduce the number of production features, there are only eight production features left. They are WIP_5 (WIP number in 5-inch), PoBW_DF, PoBW_LT, PoBW_DE, PoBW_WT (proportion of WIP in diffusion area, lithography area, dry etching area, wet cleaning area to WIP), NoBL(number of hot lots in the system), NoBL_DF and NoBL_LT(proportion of hot lots in diffusion area and lithography area). Using the eight attributes, different scheduling methods are applied in the operation of the production line and the production performances are recorded and analyzed.

For most semiconductor production applications, the diffusion area and lithography area are usually the focus of scheduling, because a diffusion machine is a batch processing unit in which two or more lots are organized to be processed together, and a lithography machine is a bottleneck unit since it is very expensive. Thus, the dynamic scheduling method proposed in this paper and traditional heuristic rules are applied to these two working areas, with FIFO applied to the other working areas.

In the experiment, 100 samples were collected and used (as described before). Of these, 80 samples were randomly selected as training samples, and the other 20 samples were used as test samples. The simulation model was initialized based on sample data. Different scheduling rules are used to run the model for a scheduling period and the production performance is recorded at the end of each scheduling period. Taking indicator "MOV" as an example, Table 1 provides the scheduling results of 10 samples randomly selected from the test samples using different scheduling rules.

In Table 1, columns of 2, 3, and 4 are the results of applying traditional heuristic rules (for example, GR_SPT means the diffusion area uses a GR, or general rule which is an empirical composite rule considering several dispatching factors (e.g. priority, the remaining number of steps and Q-time) in the production line, and the lithography area uses a SPT, or shortest processing time, rule). LS is an abbreviation for least slack, listed as GR_LS in column 3. Column 5 is the result of optimized composite rules whose weights are determined by response surface methodology, and column 6 is the result of applying the proposed scheduling method in this paper.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 3 October 2017 doi:10.20944/preprints201710.0013.v1

Table 1. Performance indicator "MOV" (in step) under different scheduling methods

ID	GR_GR	GR_SPT	GR_LS	RSM	Scheduling method proposed
1	67452	83350	83983	90612	89706
2	76773	89605	89821	94796	93744
3	89285	89028	89484	90181	86574
4	85014	84864	85486	87154	85411
5	91270	91348	91307	92915	92714
6	67936	84244	83610	88961	86851
7	67246	88544	89900	95145	93224
8	91279	89851	91114	91478	92111
9	91053	93165	94757	93165	94828
10	101650	100736	101902	104383	104516
Average	82896	89474	90136	92879	91968
Optimization degree	0.893	0.963	0.970	1.000	0.990

The better operation of the production line is associated with the larger indicator "MOV" under the same or near same conditions of other production indicators. In the randomly selected 10 samples in Table 1, when compared with a single heuristic scheduling rule, the dynamic scheduling method proposed in this paper is more likely to produce an optimal MOV and it can get a better average MOV. Therefore, the dynamic scheduling method proposed in this paper is effective in terms of "MOV" indicator. Because the learning sample is collected according to overall performance of five indicators, some records show that traditional heuristic rules are better than optimized composite rule (its weights determined by RSM) and dynamic scheduling method in terms of the "MOV" indicator. But overall, the proposed dynamic scheduling method is better than traditional heuristic rules.

In order to evaluate the overall production performance of the semiconductor production line, the average of each performance indicator for the 20 test samples when using the different scheduling methods was determined. These results are shown in Table 2.

Table 2. The average of production performance indicators under different scheduling methods

Scheduling decisions	GR_GR	GR_SPT	GR_LS	RSM	Scheduling method proposed
MCT(day)	44.86	44.97	44.76	46.38	45.81
PR(%)	0.3267	0.3338	0.3322	0.3561	0.3523
MOV(step)	85231	89569	90383	92868	92011
WIP(piece)	72051	72046	72048	72030	71186
OEE(%)	0.2917	0.3072	0.3097	0.3202	0.3114

Table 2 indicates that MCT, MOV and OEE are most affected by differing scheduling methods. The MCT under the heuristic scheduling rule is better than the one under the proposed dynamic scheduling method while the MOV and OEE are otherwise. The semiconductor production cycle is very long (more than 40 days in the test case) and the scheduling interval time relatively short (only 4 hours in the test case in practice), so dynamic scheduling has little effect on MCT. In order to analyze the effect of different scheduling methods on 5 performance indicators, the above 5 performance indicator values are normalized, multiplied by their weights and added together. For

 doi:10.20944/preprints201710.0013.v1

simplicity, it is assumed that they have equal weight (i.e. weight =0.2 for each indicator), a condition that was also done for the previous sample generation. Once these conditions are applied, a comprehensive value can be obtained that reflects a variety of production performances. Those values are given in Table 2.

The normalization process is as follows: for a performance indicator, the maximum value is set to “1”, the minimum value is set to “0”, and the other value is set to between “0” and “1” depending on its position between the maximum value and the minimum value. That is, all the performance indicators are normalized. The comprehensive value is weighted sum of normalized value. The greater the comprehensive value, the better the overall performance will be. Table 3 shows that among the four scheduling methods (GR_GR, GR_SPT, GR_LS and the proposed scheduling method), the value for the proposed scheduling method is the largest, and that for the traditional heuristic rule GR_LS is the next largest. Therefore, considering the overall optimization of the five production performance indicators, the dynamic scheduling method proposed in this paper represents a significant improvement over simple heuristic rules in most circumstances, with a slight loss of comparable productivity in some instances. When applying a single heuristic rule, the scheduling rule does not change with the state of the production line. In other words, it does not consider whether the applied scheduling rules match the current state of the production line or not, while the dynamic scheduling method considers it. As a result, the overall performance is worse than that provided by the dynamic scheduling method.

Table 3. Normalized value and comprehensive value of the five performance indicators under different scheduling methods

Scheduling decisions	GR_GR	GR_SPT	GR_LS	RSM	Scheduling method proposed
MCT	0.9367	0.8732	1	0	0.3527
PE	0	0.2401	0.1855	1	0.8711
MOV	0	0.5681	0.6746	1	0.8878
WIP	1	0.9942	0.9956	0.9749	0
OEE	0	0.5439	0.6316	1	0.6912
Comprehensive value	0.2342	0.5563	0.6229	0.7500	0.7007

6 Conclusion

Often in industry, a simple dispatching rule cannot meet actual production demand. To improve production, a composite dispatching rule is proposed that considers various factors. This rule can change rule parameters dynamically to meet the requirements of different production states of a production line. One way to realize dynamic scheduling in an actual semiconductor production line is to use a machine learning method. Such a method obtains dynamic scheduling knowledge from optimized scheduling samples, and then utilizes the appropriate dispatching rules, which can be selected to optimize the performance of the production line according to its state. For this purpose, a dynamic scheduling method based on SVR was studied. A real time optimal scheduling strategy was obtained using this method. This method was tested on a 5-inch and 6-inch semiconductor production line. The experimental results show that using a scheduling method based on composite rules gives an obvious improvement in production performance when compared with a single heuristic rule.

Acknowledgments: This paper is supported in part by National Nature Foundation of China (No. 51475334, 71690234).

Author Contributions: Ma, Y.M. and Qiao, F. proposed the idea, implemented experiments and wrote the manuscript; Zhao, F. supervised the study and the manuscript writing process; Sutherland, J. co-supervised the study. And all the authors discussed the results and contributed to the final manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Mouelhi-Chibani, W. ;Pierreval, H. Training a neural network to select dispatching rules in real time. Computers & Industrial Engineering 2010, 58(2):249–256, DOI: 10.1016/j.cie.2009.03.008.
2. Priore, P.; Gomez, A.; Pino, R.; Rosillo, R. Dynamic scheduling of manufacturing systems using machine learning: An updated review. Artificial Intelligence for Engineering Design, Analysis and Manufacturing 2014, 8(1):83-97, DOI: 10.1017/S0890060413000516.
3. Shiue, Y.R.; Guh, R.S. and Tseng, T.Y. Study on shop floor control system in semiconductor fabrication by self-organizing map-based intelligent multi-controller. Computers & Industrial Engineering 2012, 62(4): DOI: 1119-1129, 10.1016/j.cie.2012.01.004.
4. Tsai, C.J.; Huang, H.P. A real-time scheduling and rescheduling system based on RFID for semiconductor foundry FABs. Journal of the Chinese Institute of Industrial Engineers 2007, 24(6): 437-445, DOI: 10.1080/10170660709509058.
5. Olafsson, S. ; Li, X.N. Learning effective new single machine dispatching rules from optimal scheduling data. International Journal of Production Economics 2010, 128(1):118-126, DOI: 10.1016/j.ijpe.2010.06.004.
6. Ma, Y. M.; Chen, X.; Qiao, F. The Research and Application of a Dynamic Dispatching Strategy Selection Approach based on BPSO-SVM for Semiconductor Production Line. The 11th IEEE International Conference on Networking, Sensing and Control (ICNSC2014), Florida, USA, 7-9 April 2014; 74-79.
7. Qiao, F.; Ma, Y. M.; Gu, X. Attribute selection algorithm of data-based scheduling strategy for semiconductor manufacturing. IEEE International Conference on Automation Science and Engineering,(CASE 2013), Madison, WI, USA , 17-20 Aug. 2013; 410-415..
8. Li, L.; Sun, Z.J.; Zhou, M. et al. Adaptive Dispatching Rule for Semiconductor Wafer Fabrication Facility. IEEE Transactions on Automation Science and Engineering 2013, 10(2):354-364, DOI: 10.1109/TASE.2012.2221087.
9. Lee, Y.F.; Jiang, Z.B.; Liong, H.C. A smart adaptive production control system for semiconductor foundry fab. The 8th World Multi-conference on System, Cybernetics and Informatics (SCI 2004), Florida, USA, 18-21 July 2004.
10. Pickardt, C.W. ; Hildebrandt, T; Branke, J ; Heger, J. Evolutionary generation of dispatching rule sets for complex dynamic scheduling problems. International Journal of Production Economics 2013,145(1):67–77, DOI: 10.1016/j.ijpe.2012.10.016.
11. Bouri, A.E. ; Amin, G.R. A combined OWA-DEA method for dispatching rule selection. Computers and Industrial Engineering 2015, 88:470-478, DOI: 10.1016/j.cie.2015.08.007.
12. Yu,Y.H.; Wang, Y.H. Design and Implementation of a Real-Time Job shop Scheduling System. International Asia Conference on Industrial Engineering and Management Innovation (IEMI2012), Beijing, China, August 10, 2012 - August 11, 2012.
13. Chen, T. An effective dispatching rule for bi-objective job scheduling in a wafer fabrication factory—considering the average cycle time and the maximum lateness. The International Journal of Advanced Manufacturing Technology2013, 67 (5): 1281-1295, DOI:10.1007/s00170-012-4565-6.
14. Dabbas, R.M; Fowler, J.W. A new scheduling approach using combined dispatching criteria in wafer fabs. IEEE Transactions on Semiconductor Manufacturing 2003;16:501-510, DOI:10.1109/TSM.2003.815201.
15. Kai, Y.; Qiao, F.; Ma, Y.M. General structure of the semiconductor production scheduling model. Applied Mechanics and Materials 2010, 20-23: 465-469.
16. Devor, R.E.; Chang, T.; Sutherland J.W. Statistical Quality Design and Control: Contemporary Concepts and Methods, 2nd. Prentice Hall: New York, USA, 2006; ISBN:9780130413444.
17. Huang, T.M.; Kecman, V.; Kopriva, I. Kernel Based Algorithms for Mining Huge Data Sets. Springer Berlin Heidelberg, 2006. ISBN:3540316817, 9783540316817.
18. Campbell C; Ying Y. Learning with support vector machines. Synthesis Lectures on Artificial Intelligence and Machine Learning 2011,10: 1-95, DOI: 10.2200/S00324ED1V01Y201102AIM010.
19. Ma, Y.M.; Qiao, F.; Chen, X. A dynamic scheduling approach based on SVM for semiconductor production line. Computer Integrated Manufacturing System 2015, 21(3):733-739,. DOI: 10.13196/j.cims.2015.03.018.
20. Jiang, Z.B. Modeling and Optimal Scheduling Control of Semiconductor Manufacturing System. Shanghai Jiaotong University Press: Shanghai, China, 2011; ISBN:7313063156, 9787313063151.