Proceedings of the 2017 Winter Simulation Conference
W. K. V. Chan, A. D'Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. Page, eds.

SCHEDULING STRATEGY OF SEMICONDUCTOR PRODUCTION LINES WITH REMAINING CYCLE TIME PREDICTION

Li Li

Department of Control Science & Engineering
College of Electronics & Information Engineering
Tongji University
Shanghai, 201804, CHINA

Qingyun Yu

Department of Control Science & Engineering
College of Electronics & Information Engineering
Tongji University
Shanghai, 201804, CHINA

ABSTRACT

With the rapid development of semiconductor manufacturing, customers' demand for on-time delivery rate (ODR) makes scheduling strategies face new challenges. In order to meet customers' delivery requirements, scheduling strategies generally need to comprehensively consider remaining cycle time (CT), ODR, movement (MOV) speed and machine load balancing. In order to solve these problems, this paper proposed a scheduling strategy of semiconductor production lines with remaining cycle time prediction. Firstly, gather features related to performance index and then filtrate them through dimension reduction method. Secondly, use the above feature subset to build remaining cycle time prediction model by random forest algorithm. Next, design the scheduling strategy of semiconductor production lines with remaining cycle time prediction. Finally, make simulation experiments to verify the effectiveness of the proposed scheduling strategy. Simulation results show that the proposed scheduling strategy can improve the mean CT, throughput (TH), machine utilization time (MUT) and ODR in different extant.

1 INTRODUCTION

The development level of the semiconductor manufacturing has become an important symbol which can measure national economic development and social progress, and semiconductor manufacturing process has received extensive attention from academia and industrial field (Shen et al. 2003). The scheduling problem of semiconductor production line usually has characteristics of multi-objective, multi-constraint, nonlinearity, uncertainty and difficult modeling, which greatly increase the difficulty in scheduling problems (Li et al. 2012). CT is one of the most important performance indexes in semiconductor production line (Deng et al. 2014, Haizan et al. 2012), which is closely related to ODR. So, many scheduling strategies take remaining cycle time as one of the decision variables (Hildebrandt et al. 2014, Wu et al. 2014). In addition, cycle time is an important decision variable of release control. Therefore, we must deeply study the effect of remaining cycle time on semiconductor manufacturing system.

With the development of information increasingly deepening, large number of data was produced and stored. There must exist a lot of knowledge related to scheduling in those historical data (Yu et al. 2014). How to use those data to mine knowledge related to scheduling and assist complex manufacturing systems to build scheduling model and make optimization is a problem worthy of study (Chang et al. 2012). Data-based remaining cycle time prediction can consider influences of multiple uncertain factors, dig potential knowledge for prediction model and use the predicted value as one of the decision variables, and then optimize performance indexes of production line (Yen et al. 2012). Tirkel et al. (2011) predicted cycle time of some specific processes using data in a manufacturing execution system (MES) through decision tree and neural network technology and the accuracies were 76.5% and 87.6%, respectively. Chien et al. (2005) comprehensively considered work in process (WIP) and TH in their scheduling strategy, predicted cycle time through domain knowledge and data mining and then applied the proposed strategy into a real production line. Results showed that their proposed strategy played a guidance role in decision-making process for the production line. To reduce CT, Meidan et al. (2011) suggested and investigated a data-driven approach that identified key factors and predicted their impacts on CT. They identified the most influential factors using conditional mutual information maximization, and then applied the Selective Naive Bayesian Classifier (SNBC) for further selection of a minimal, most discriminative key-factor set for cycle time prediction. Results showed their method improved the accuracy of cycle time prediction in nearly 40% while narrowing the list of factors from 182 to 20. Based on systematic research of related works worldwide, Wu et al. (2009) proposed a data-based scheduling framework composed of a data layer and a model layer for complex manufacturing processes. They discussed related theories, methodologies, and technologies for this scheduling framework as well. Based on a simulated Non-Volatile Memory (NVM) fab, Hassoun et al. (2013) showed that forecasting the steady state cycle time of process segments was possible using certain segment characteristics. They also showed that the cycle time predictability was highly dependent on the choice of the segmentation, with the more efficient segmentation corresponding to the product layers.

Make comprehensive consideration, this paper proposes a scheduling strategy with remaining cycle time prediction (SRCTP). The proposed method mainly contains two parts: remaining cycle time prediction model and scheduling algorithm.

2 PROBLEM ASSUMPTIONS AND DEFINITIONS

2.1 Problem assumptions

During the study on SRCTP, this paper makes the following assumptions:

(1) The information related to job dispatching, e.g., job processing time, WIP in the queue and available time of a machine, can be obtained from Manufacturing Execution System (MES) or other information systems of a fab.

(2) When making dispatching decisions for batch processing machines (BPMs), there are two main steps. The first one is to batch the jobs. There are two important constraints: 1) only jobs using the same recipe can be processed together as a batch and 2) the batch size should be no greater than the capacity of a BPM. In addition, good tradeoff between the wasted time and wasted capacity of a BPM should be made. The second step is to determine the priorities of the batches. The main concerns are the same as those in assumption 2 for non-BPMs.

(3) Once processing begins on one batch, no job can be removed from or added to the machine until the present one is done.

2.2 Definitions of Parameters and Variables

The proposed SRCTP is described in the form of a new workflow involving both BPMs and non-BPMs. To do so we first define variables and parameters in it.

a: the input vector
b: the output vector
i: index of the available machine
j: the index of vectors
id: index of the downstream machine
im: index of the recipes of machine i
m: the number of features
n: index of the jobs in the queue of machine i
t: dispatching decision point, i.e., dispatching time
D: database
F: the original feature set
L: category label
X: a random variable
Y: a random variable
$(\alpha, \beta, \gamma, \sigma)$: random indexes which can measure the relative importance of each information
β_k: an independent identically distributed random vector
B_i: capacity of machine i of type BPM
D_n: due date of job n
f_i: feature
$h(a, \beta_k)$: meta classifier
H(X): the information entropy of variable X
H(Y): the information entropy of variable Y

H(X|Y): the new information entropy of variable X after observing variable Y

IG(X|Y): the information gain between variable X and variable Y

M_i: number of the recipes on machine i

N_{im}: number of the jobs in the queue of machine i using recipe im

N_{id}^{k} : the maximum workload of downstream machine id

N_{ik}^{h}: the number of hot lot in the batch ik

p_n: the dispatching priority of job n

p_i^k: the required processing time of batch k on machine i

p_n^{id}: occupation time of job n on

P(x): the prior probability of variable X

P(y): the prior probability of variable Y

P(x|y): the new prior probability of variable X after observing variable Y

SU(X,Y): the new information gain after normalization

T_{id}: available time of machine id in one day

t_{im}: the processing time of recipe im

t_{id}^{m} : the processing time of recipe m on the downstream machine id

t_n^{point}: the last time to make remaining cycle time prediction

t_n^{pre}: the predicted value of remaining cycle time

WT_{in}: the dwell time of job n on machine i

x_i^B: a binary variable. If machine i is a bottleneck at time t, $x_i^B = 1$; and otherwise, $x_i^B = 0$.

x_{id}^{I}: a binary variable. If machine id is idelat time t, $x_{id}^{I} = 1$; and otherwise, $x_{id}^{I} = 0$.

x_n^{im}: a binary variable. If job n uses recipe im on machine i, $x_n^{im} = 1$; and otherwise, $x_n^{im} = 0$.

x_n^H: a binary variable. If job n is a hot lot at time t, $x_n^H = 1$; and otherwise, $x_n^H = 0$.

x_n^{pre} : a binary variable. If need to predict the remaining cycle time, $x_n^{pre} = 1$; and otherwise, $x_n^{pre} = 0$.

$\tau_i^n(t)$: urgency degree for job n to be processed on machine i at time t

$\tau_{id}^n(t)$: urgency degree for job n to be processed on downstream machine id at time t

3 DATA-BASED REMAINING CYCLE TIME PREDICTION MODEL

3.1 Framework of SRCTP

By fully considering the aforementioned problems, the scheduling framework is designed as shown in Fig.1. It mainly consists of two parts: (1) Feature selection; (2) Building prediction model based on RF. Feature selection is performed to select the production attributes that are the most relevant to the formulation of a production scheduling strategy, remove redundant and invalid production attributes, reduce modeling time, and increase the prediction model's accuracy. The prediction model based on RF can compute the theoretical remaining cycle time according to current status of the production line. As a result, the scheduling strategy can respond to the changes in remaining cycle time. The specific flowchart will be shown in section 3.4.

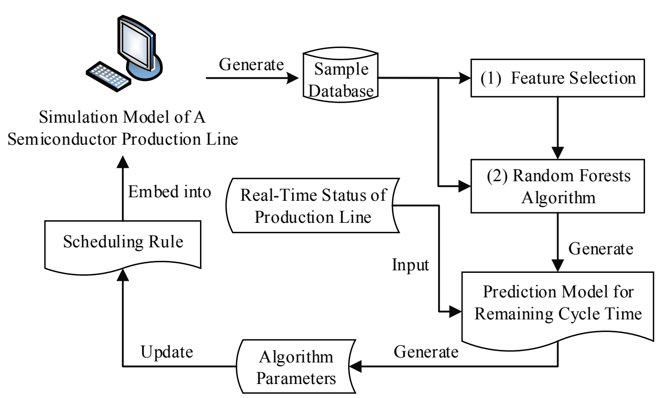

Figure 1: Scheduling Framework.

3.2 Feature Selection

This research object has dozens of features related to remaining cycle time. If use all these features to build prediction model directly, it will waste computation resources and reduce prediction accuracy. So we must reduce feature dimensions firstly. Here we adopt correlation analysis to make feature selection and then obtain the target feature subset.

In this paper, we adopt information gain as the correlation measure of characteristics and category labels. Select a set of features with strong classification ability according to information gain and then obtain the downsized feature subset (Kivijarvi et al. 2003). The definition of information gain is as below:

Set X as a random variable, the information entropy of X is defined as (1):

$$H(X) = -\sum_i P(x_i)\log_2(P(x_i)) \tag{1}$$

By observing random variable Y, the information entropy of X changes to $H(X|Y)$, defined as (2):

$$H(X|Y) = -\sum_j P(y_j)\sum_i P(x_i|y_j)\log_2(P(x_i|y_j)) \tag{2}$$

After introducing random variable Y, the new information entropy of X is smaller than $H(X)$, i.e., after introducing Y, the uncertainty degree of X will become smaller or remain unchanged. If Y is uncorrelated with X, $H(X|Y) = H(X)$; else, $H(X|Y) < H(X)$. The larger the $(H(X) - H(X|Y))$ is, the stronger the correlation between X and Y will be. Set information entropy $IG(X|Y)$ as the difference value between $H(X)$ and $H(X|Y)$, defined as (3):

$$IG(X|Y) = H(X) - H(X|Y) \tag{3}$$

Normalize information gains according to (4):

$$SU(X,Y) = 2[IG(X|Y)/(H(X) + H(Y)] \tag{4}$$

On account of these definitions about dependence measures, make features sorting and filtering based on the correlation between feature f_i and category label L. Then select several features with the strongest correlation to form the objective feature subset. The algorithm flow is shown in Algorithm 1:

Algorithm 1: *Features Filtering*(D, F, m)	
Input：	database D with category label, original feature set F, the number of features m
Output：	simplified feature subset
Steps：	1） compute information entropies SU_i between every feature f_i and category label according to (1)~(4); 2） sort these features in descending order according to SU_i; 3） select the first m features as the simplified feature subset.

3.3 Build Remaining Cycle Time Prediction Model Based on Random Forest Algorithm

Random Forest (RF) is an ensemble learning technique to improve the accuracy of methods using classification and regression trees by combining a large set of decision trees. It uses Classification and Regression Trees (CART) as the meta classifier and generates different training sample sets through Bagging. RF is suitable for high-dimensional and small sample data, does not need complex parameter selection process.

RF is a set of tree classifiers $\{h(a, \beta_k), k = 1,2,\cdots, ntree\}$. Where, meta classifier $h(a, \beta_k)$ is a classification and regression tree generated from CART algorithm without pruning. a is the input vector. β_k is an independent identically distributed random vector which can determine the growth progress of single trees. For regression problems, the final output is obtained by calculating an average of all tree predictions. The algorithm flow of RF is shown in Algorithm 2:

Algorithm 2: *Random Forest*	
Input:	1.training set $S = \{(a_j, b_j), j = 1,2,\cdots, n\}, (A, B) \in R^d \times R$
	2. testing set $a_j \in R^d$
	for $j = 1,2,\cdots, N_{tree}$
	(1) carry on sample in original training set S Bootstrap, obtain the training set S_j
	(2) generate a tree h_j without pruning by using S_j
	a. randomly select M_{try} features from d features
	b. for every node, select the best feature from M_{try} features by Gini index
	c. make mitosis until grow to the biggest one
	end
Output:	1. trees set $\{h_j, j = 1,2,\cdots, N_{tree}\}$
	2. for testing sample a_j, the decision tree export $h_j(a_j)$
	regression: $f(a_j) = \frac{1}{N_{tree}} \sum_{j=1}^{N_{tree}} h_j(x_j)$
	classification: $f(a_j) = \text{majority vote}\{h_j(a_j)\}_{j=1}^{N_{tree}}$

There are two factors which may affect the accuracy of RF: the quantity of decision trees and the minimum quantity of node samples. This paper selects the error of outside bag as the index to evaluate whether the parameters of RF is good. Select these historical data of remaining cycle time as the learning sample when WIP is 6500, 7000, 7500, and 8000 pieces. By setting different leaf-node thresholds and decision trees quantities, we can obtain the statistical results (shown in table 1):

Table 1: Prediction Accuracy of Remaining Cycle Time with Different WIP, Leaf-Node Threshold, and Trees' Quantity.

WIP	**Leaf-Node Threshold**	**Quantity of Trees**				
		10	**20**	**30**	**40**	**50**
6500	1	3.36	1.03	0.91	0.88	0.84
	2	3.94	1.28	1.11	0.98	0.89
	3	3.22	1.11	0.96	0.89	0.89
	5	3.36	1.51	1.30	1.15	1.11
	7	3.73	1.42	1.21	1.09	1.08
	9	3.51	1.53	1.32	1.26	1.24
7000	1	4.74	2.66	2.53	2.46	2.43
	2	5.52	2.71	2.60	2.54	2.49
	3	5.94	3.11	2.80	2.66	2.55
	5	5.76	2.94	2.75	2.68	2.61
	7	5.67	3.05	2.86	2.86	2.85
	9	5.89	3.24	2.99	3.00	2.96
7500	1	5.92	3.08	2.86	2.82	2.75
	2	6.22	3.26	3.02	2.95	2.91

	3	6.02	3.12	2.97	2.95	2.88
	5	6.50	3.35	3.16	3.05	2.98
	7	7.09	3.85	3.53	3.45	3.35
	9	6.53	3.72	3.48	3.43	3.36
8000	1	6.64	3.64	3.40	3.36	3.28
	2	7.04	3.73	3.44	3.41	3.32
	3	6.54	3.76	3.63	3.55	3.52
	5	7.35	4.04	3.78	3.63	3.52
	7	7.35	3.90	3.80	3.75	3.66
	9	6.98	4.24	3.91	3.83	3.76

We can conclude from table 1 that when the quantity of decision tree is less than 20, the main factor which affect the prediction accuracy is the quantity of decision tree; when the quantity of decision tree is more than 40, the main factor which affect the prediction accuracy is the leaf-node threshold; with the reduction of sample quantity, the error becomes smaller. Make overall consideration, this paper set the quantity of decision tree to 40 and set the leaf-node threshold to 2.

3.4 Flowchart of SRCTP

SRCTP considered multiple characteristics of a semiconductor production line, such as machine load balancing, dispatching of hot lot, and dispatching of batch workpiece, and it took the prediction value of remaining cycle time as one of the decision variables. The proposed SRCTP is described in the form of a new work-flow involving both BPMs and non-BPMs. The decision flow is shown in Fig.2.

Step 1: if machine i is available at time t, judge whether the machine is a BPM. If yes, turn to step 8. Otherwise, turn to step 2.

Step 2: judge whether need to predict the remaining cycle time of the current workpiece according to (5).

$$If\ \sum_{im} N_{im} \geq (24/\min(t_{im}))or\ \left(t - t_n^{point}\right) > 24hour, then\ x_n^{pre} = 1 \tag{5}$$

Step 3: gather related data from the production line, predict the remaining cycle time according to the RF algorithm mentioned in section 3.3 and update it.

Step 4: compute the emergency degree of workpiece in the queue according to (6).

$$\tau_i^n(t) = \begin{cases} MAX, & t_n^{pre} \geq D_n - t \\ t_n^{pre}/(D_n - t + 1) - P_n^i/\sum_n p_n^i, & t_n^{pre} < D_n - t \end{cases} \tag{6}$$

Step 5: compute the workload degree of machine according to (7).

$$\tau_{id}^n(t) = \sum_n P_n^{id}/T_{id} \tag{7}$$

Step 6: compute the dispatching priority of workpiece in the queue according to (8).

$$p_n = \begin{cases} WT_i^n, & \tau_i^n(t) = MAX \\ \alpha_1\tau_i^n(t) - \beta_1\tau_{id}^n(t), & \tau_i^n(t) \neq MAX \end{cases} \tag{8}$$

Step 7: sort the priorities of workpiece in the queue and select the workpiece with the highest priority to process on machine i.

Step 8: judge whether need to predict the remaining cycle time of the current workpiece according to (9).

$$If\ \sum_{im} N_{im} \geq (24B_i/\min(t_{im}))or\ \left(t - t_n^{point}\right) > 24hour, then\ x_n^{pre} = 1 \qquad (9)$$

Step 9: gather related data from the production line, predict the remaining cycle time according to the RF algorithm mentioned in section 2.2 and update the predicted value t_n^{pre}.

Step 10: compute the emergency degree of workpiece in the queue according to (6).

Step 11: traverse all workpieces in the queue before machine i, check whether exist hot lot.

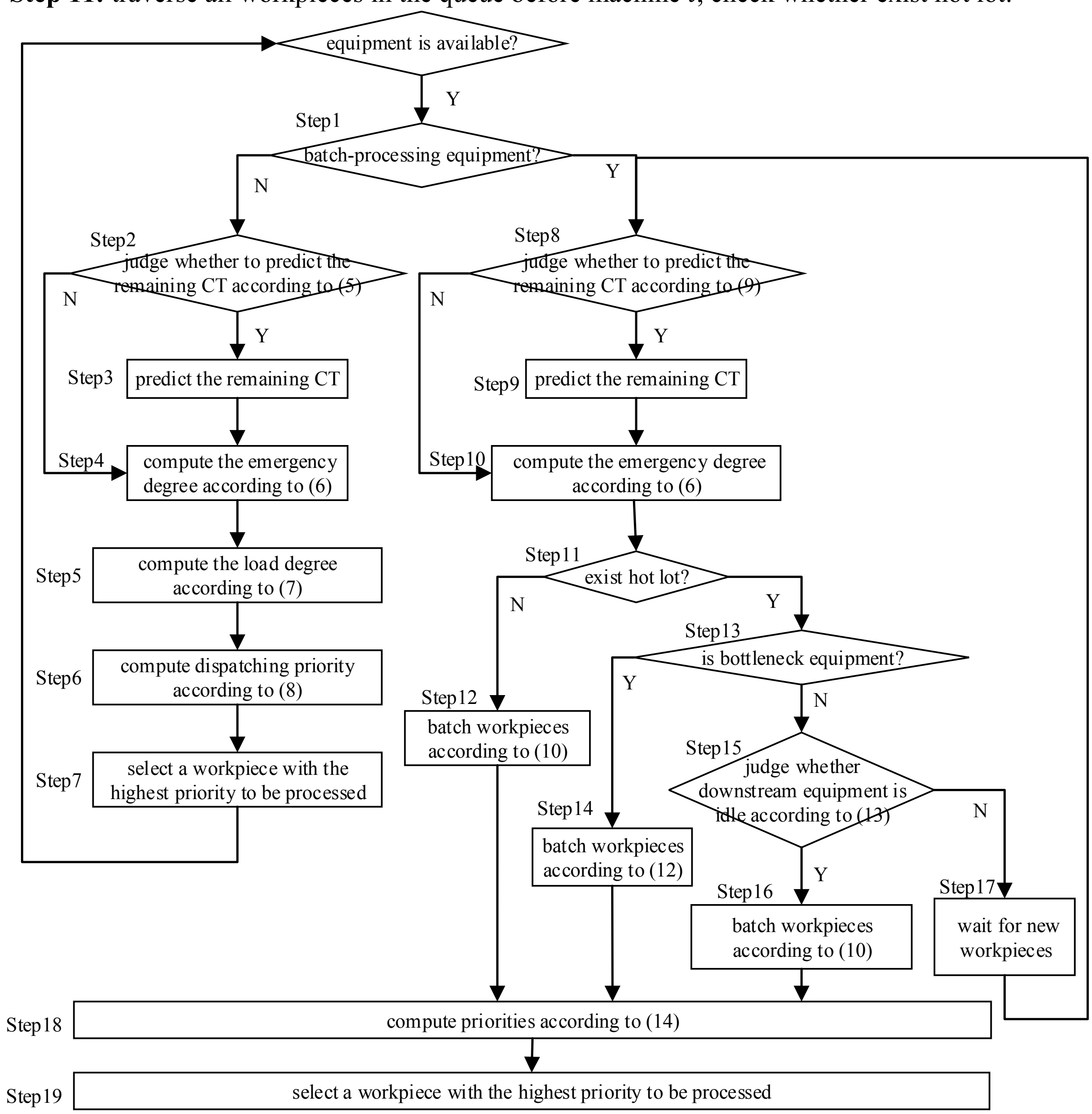

Figure 2: Decision Flow of SRCTP

Step 12: batch workpieces according to (10).

$$\begin{aligned} & for \;\; m = 1 \;\; to \; M_i \\ & \quad if \;\; 0 \leq \sum x_n^{im} x_n^H < B_i \\ & \quad then \;\; Select\{\min\{(B_i - x_n^{im} x_n^H), (N_{im} - x_n^{im} x_n^H)\}\}|_{\max(WT_i^n)} \\ & \quad else \; if \;\; \sum x_n^{im} x_n^H \geq B_i \\ & \quad then \; Select\{B_i\}|_{\max(t_n^{pre} - (D_n - t))} \end{aligned} \tag{10}$$

Step 13: judge whether machine i is a bottleneck one according to (11).

$$If \;\; \sum_{im} N_{im} \geq (24B_i/\min(t_{im})), then \; x_i^B = 1 \tag{11}$$

Step 14: batch workpieces with the same process menu according to (12).

$$Select\{B_i\}|_{\max(WT_i^n)} \tag{12}$$

Step 15: judge whether the downstream machine id is idle according to (13).

$$if \;\; \sum_{im} N_{id} \leq (24B_i/\max(t_{id}^w)), then \; x_{id}^I = 1 \tag{13}$$

Step 16: batch workpieces according to (10).

Step 17: wait for the new coming of workpieces and turn to step 8.

Step 18: compute the priority of each batch according to (14).

$$p_k = \alpha \frac{N_{ik}^h}{B_i} + \beta \frac{B_i}{\max(B_i)} - \gamma \frac{p_i^k}{\max(p_i^k)} - \sigma \frac{N_{id}^k}{\sum_k N_{id}^k + 1} \tag{14}$$

Step 19: sort all workpieces in the queue according to its priority and select a workpiece with the highest priority to process.

4 SIMULATION AND VERIFICATION

Take an actual 6" wafer production line of a wafer manufacturer in Shanghai as the object, we built a corresponding simulation model in our previous work. The production line has 11 processing areas including oxidation, diffusion, injection, epitaxial growth, photolithography, dry etching, deposition, sputtering and wet cleaning and another three non-processing areas, i.e., the virtual machine, testing and outsourcing. In addition, its machines can be divided into five different types: single-processing machines, batch-processing machines, multi-wafer processing machines, cluster tools and tanks.

The simulation system is built by using Tecnomatix Plant Simulation and selecting SQL Server 2008 as the DATA PaaS, and is mainly composed by two layers: simulation model and database. The former includes show layer of model and control layer of simulation, the latter stores information related to the production system and the simulation results.

4.1 Performance Comparison among Different Scheduling Strategies

Firstly, study the influence of remaining cycle time prediction on the system. Set these parameters $(\alpha, \beta, \gamma, \sigma)$ as 0.5. Make simulations with different scheduling strategies under 6500, 7000, 7500, and 8000 pieces WIP level, results are shown in table 2. Where, SRCTP-NB represents applying SRCTP only on non-BPMs; SRCTP-B represents applying SRCTP only on BPMs; DDR is a Dynamic Dispatching Rule (DDR) presented by Li Li et al. in 2012 (Li et al. 2012); FIFO represents a scheduling strategy named First in First out (FIFO); EDD represents a scheduling strategy named Earliest Due Date (EDD). MOV represents the average movement of workpieces and its unit is ten thousand steps; MOV_D represents the variance of

MOV, its order of magnitudes is thousand; TH represents the average TH, its unit is slice; TH_D is the variance of TH; MUT is the machine utilization time, its unit is million seconds. The simulating time span is 90 days and the first 30 days are the warming-up period.

Table 2: Performance Comparison among Different Scheduling Strategies.

WIP	Performance	Scheduling Strategies				
		SRCTP_NB	**SRCTP_B**	**DDR**	**FIFO**	**EDD**
6500	MOV	3.72	3.66	3.66	3.33	3.13
	MOV_D	4.83	5.30	5.22	5.98	3.84
	TH	7.48	7.06	7.00	7.83	6.06
	TH_D	5.66	8.46	8.54	4.40	3.65
	MUT	2.60	2.54	2.51	2.29	2.35
7000	MOV	3.65	3.55	3.57	3.39	3.16
	MOV_D	4.90	4.54	5.03	6.81	4.14
	TH	7.56	7.46	7.55	7.80	6.13
	TH_D	6.95	7.84	7.49	4.51	3.48
	MUT	2.60	2.47	2.48	2.33	2.37
7500	MOV	3.70	3.61	3.56	3.30	3.15
	MOV_D	4.90	4.54	5.03	6.00	4.12
	TH	7.56	7.46	7.55	8.15	6.22
	TH_D	6.95	7.84	7.49	4.21	3.35
	MUT	2.58	2.50	2.48	2.27	2.36
8000	MOV	3.68	3.59	3.59	3.30	3.20
	MOV_D	4.48	4.73	5.37	6.81	4.65
	TH	7.96	8.01	7.75	8.28	6.39
	TH_D	6.22	7.68	6.94	4.71	3.53
	MUT	2.65	2.51	2.53	2.25	2.40

Through comparing performances in different scheduling strategies, this paper obtained the following three conclusions.

(1) Heuristic scheduling rules usually only optimize a few performances, which result in that a few performances are much better but the else performances are not good. SRCTP_NB, SRCTP_B and DDR can optimize most performances better in different load situation, because these three methods considered multiple factors from aspects of workpiece, machine, workload situation, and batch mode. Besides, with the increase of WIP level, the advantage becomes much more obviously.
(2) Performance MOV and TH of DDR are worse than those of SRCTP_NB and SRCTP_B. This means that remaining cycle time prediction can improve TH and MOV. When WIP level is 6500 pieces, MOV and TH improved 1.6% and 6.9%, respectively; when WIP level is 7000 pieces, MOV and TH improved 2.2% and 1.5%, respectively; when WIP level is 7500 pieces, MOV and TH improved 3.7% and 1.4%, respectively; when WIP level is 8000 pieces, MOV and TH improved 2.6% and 2.6%, respectively. So, remaining cycle time prediction can improve MOV and TH of a production line.
(3) No matter how much the WIP level is, performances of SRCTP_NB are better than those of SRCTP_B. In other words, use SRCTP on non-BPMs, the production line will obtain much better performances.

4.2 CT and ODR of Different Products

Since the difference among products in real semiconductor production line is usually non-negligible. This paper counted CT and variance of CT of different 9 products with different scheduling strategies and average ODR of different 9 products, respectively shown in table 3 - table 5.

Table 3: CT of Different 9 Products.

Scheduling Strategies	Product Category								
	NO.1	NO.2	NO.3	NO.4	NO.5	NO.6	NO.7	NO.8	NO.9
SRCTP-NB	44.21	20.59	38.73	52.67	50.65	21.65	20.10	88.17	41.60
SRCTP-B	44.18	21.20	43.65	50.26	48.73	22.04	21.90	83.21	40.36
DDR	44.16	20.91	44.80	49.65	48.11	21.61	22.33	83.21	41.93
FIFO	45.48	12.29	60.07	58.97	39.84	12.74	38.42	null	50.75

Table 4: Variance of CT of Different 9 Products.

WIP	Scheduling Strategies	Product Category								
		NO.1	NO.2	NO.3	NO.4	NO.5	NO.6	NO.7	NO.8	NO.9
6500	SRCTP-NB	3.04	1.94	6.69	2.14	2.74	1.45	3.59	2.85	3.61
	SRCTP-B	1.13	0.78	3.87	0.41	0.76	0.72	2.86	0.47	3.00
	DDR	0.77	0.37	1.34	0.27	0.73	0.28	2.26	0.47	0.36
	FIFO	3.54	3.88	13.34	18.75	18.79	6.86	10.91	null	21.99
7000	SRCTP-NB	3.89	1.93	6.95	0.56	0.45	2.23	4.05	2.89	3.10
	SRCTP-B	3.63	1.52	4.82	1.53	1.91	2.34	2.66	2.15	3.52
	DDR	3.52	0.61	5.02	1.69	2.02	2.15	1.99	2.22	2.74
	FIFO	8.70	9.03	17.07	25.29	22.07	9.83	12.38	null	29.50
7500	SRCTP-NB	3.70	2.13	6.22	1.81	1.36	3.33	4.34	1.11	4.66
	SRCTP-B	3.87	1.77	4.09	2.08	2.15	2.33	2.63	4.26	2.48
	DDR	5.41	2.68	6.17	2.49	3.07	3.56	2.82	2.72	3.77
	FIFO	5.16	4.84	18.94	25.60	22.11	10.05	14.52	null	28.88
8000	SRCTP-NB	6.28	4.28	7.02	2.13	2.29	4.84	4.40	2.05	5.52
	SRCTP-B	6.43	3.56	7.95	3.58	3.21	4.43	3.81	5.16	5.57
	DDR	5.64	3.55	5.45	3.64	4.05	4.05	3.28	2.90	4.05
	FIFO	10.41	9.99	17.35	27.60	23.97	10.39	14.28	null	27.88

Table 5: Average ODR of Different 9 Products.

WIP	Scheduling Strategies		
	SRCTP-NB	SRCTP-B	DDR
6500	51.80%	38.06%	36.84%
7000	37.44%	25.95%	34.07%
7500	23.29%	9.11%	25.96%
8000	17.72%	4.85%	17.86%

Through table 4-6, this paper obtained the following three conclusions.

(1) There is almost no production in NO.8. That is because FIFO always select the first-in workpiece to process. Table 3 shows that CT of NO.8 is the longest one, which determine the workpiece is difficult to be selected by machine to be processed under strategy FIFO.
(2) Variance of CT under strategy SRCTP_NB, SRCTP_B and DDR are all better than strategy FIFO no matter in which WIP level and which product category. This is because strategy SRCTP_NB, SRCTP_B and DDR all comprehensively considered machine information, workpiece information and production line information. For strategy SRCTP_NB, SRCTP_B and DDR, when WIP level is 6500 pieces, the CT variance of SRCTP is larger than that of DDR. With the increase of workload level, the CT variance of SRCTP and that of DDR become closer. So, SRCTP can improve the production system much more under high load level.
(3) With the increase of workload level, the descent velocity of ODR under SRCTP is more rapidly than that of DDR. But, take MOV and TH into account, SRCTP performed better than DDR.

5 CONCLUSIONS

This paper proposed a scheduling strategy with remaining cycle time prediction for semiconductor production lines. Inspected the effectiveness of the proposed method based on a simulation model of a real production line in semiconductor manufacturing company of Shanghai. Although the proposed method performed well in the mentioned simulation model, it still has room for improvement: consider more practical problems in semiconductor production line. Such as rework issues of workpiece, addition of test workpieces and machine downtime. The follow-up work will add these aspects so that perfect the proposed method.

ACKNOWLEDGMENTS

This paper is supported in part by the National Nature Science Foundation of China under grant No. 51475334.

REFERENCES

Shen, Y., and R. C. Leachman. 2003. “Stochastic Wafer Fabrication Scheduling”. *IEEE Transactions on Semiconductor Manufacturing* 16(1): 2-14.

Li, L., Z. J. Sun, and J. C. Ni. 2012. “Data-based Scheduling Framework and Adaptive Dispatching Rule of Complex Manufacturing Systems”. *The International Journal of Advanced Manufacturing Technology* 66(9): 1891-1905.

Deng, J., Z. Cao, and M. Liu. 2014. “A Bottleneck Prediction and Rolling Horizon Scheme Combined Dynamic Scheduling Algorithm for Semiconductor Wafer Fabrication”. In *Proceedings of the International Conference on Networking Sensing and Control*, 58-63. Miami: Academic Press.

Haizan, C., L. Chang, and R. Yuan. 2012. “Research on Dynamic Dispatching Rule for Semiconductor Assembly Production Line”. In *Proceedings of the IEEE International Conference on Industrial Engineering and Engineering Management*, 493-497. Hong Kong: Academic Press.

Hildebrandt, T., D. Goswami, and M. Freitag. 2014. “Large-Scale Simulation-based Optimization of Semiconductor Dispatching Rules”. In *Proceedings of the Winter Simulation Conference*, edited by A. Tolk, S. Y. Diallo, I. O. Ryzhov, L. Yilmaz, S. Buckley, and J. A. Miller, 2580-2590. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Wu, H., and T. Chen. 2014. “A Fuzzy-Neural Ensemble and Geometric Rule Fusion Approach for Scheduling a Wafer Fabrication Factory”. *Mathematical Problems in Engineering* 2013 (Id: 956978).

Yu, K., M. G. Lee, and C. Lee. 2014. "Difference Priority Algorithm in Semiconductor Scheduling Problems". In *Proceedings of the IEEE International Conference on Industrial Engineering and Engineering Management*, 1545-1549. Bangkok: Academic Press.

Chang, C. Y., and K. H. Chang. 2012. "An Integrated and Improved Dispatching Approach to Reduce Cycle Time of Wet Etch and Furnace Operations in Semiconductor Fabrication". In *Proceedings of the IEEE International Conference on Computer Supported Cooperative Work in Design*, 734-741. Wuhan: Academic Press.

Yen, L. Y., and K. H. Chang. 2012. "Cycle Time Reduction for Photolithography Area with Multi-Workstation". In *Proceedings of the IEEE International Conference on Computer Supported Cooperative Work in Design*, 742-746. Wuhan: Academic Press.

Tirkel, I. 2011. "Cycle Time Prediction in Wafer Fabrication Line by Applying Data Mining Methods". In *Proceedings of the IEEE/SEMI Advanced Semiconductor Manufacturing Conference*, 1-5. Saratoga Springs: Academic Press.

Chien, C. F., C. W. Hsiao, and C. Meng. 2005. "Cycle Time Prediction and Control Based on Production Line Status and Manufacturing Data Mining." In *Proceedings of the IEEE International Symposium on Semiconductor Manufacturing*, 327-330. Piscataway: Academic Press.

Meidan, Y., B. Lerner, and G. Rabinowitz. 2011. "Cycle-Time Key Factor Identification and Prediction in Semiconductor Manufacturing Using Machine Learning and Data Mining". *IEEE Transactions on Semiconductor Manufacturing* 24(2): 237-248.

Wu, Q. D., F. Qiao, and L. Li. 2009. "Data-Based Scheduling for Complex Manufacturing Processes". *ACTA Automatic Science* 35(6):807-813.

Hassoun, M. 2013. "Prediction of Product Layer Cycle Time Using Data Mining". In *Proceedings of the Winter Simulation Conference*, edited by R. Pasupathy, S.-H. Kim, A. Tolk, R. Hill, and M. E. Kuhl, 3905-3911. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Kivijarvi, J., P. Franti, and O. Nevalainen. 2003. "Self-Adaptive Genetic Algorithm for Clustering". *Journal of Heuristics* 9(2):113-129.

AUTHOR BIOGRAPHIES

LI LI received the B.S. and M.S. degrees in electrical automation from Shenyang Agriculture University, Shenyang, China, in 1996 and 1999, respectively, and the Ph.D. degree in mechatronics engineering from Shenyang Institute of Automation, Chinese Academy of Science, in 2003. She joined Tongji University, Shanghai, China, in 2003, and is presently an Associate Professor of Control Science and Engineering. She has over 50 publications including 2 books, 30+ journal papers, and 2 book-chapters. Her research interests are in production planning and scheduling, computational intelligence, semiconductor manufacturing, and energy systems. Her email address is lili@tongji.edu.cn.

QINGYUN YU received the B.S. degree in Internet of Things from Jiangnan University, Wuxi, China, in 2013. Since 2013, she has been working towards the Ph.D. degree in control science and engineering at the School of Electronics and Information Engineering, Tongji University, Shanghai, China. Her research interests include complex manufacturing systems, data mining, scheduling and adaptive dynamic dispatching rule. Her email address is 953717916@qq.com.