Robotics and Computer-Integrated Manufacturing 29 (2013) 41–47

Contents lists available at SciVerse ScienceDirect

Robotics and Computer-Integrated Manufacturing

journal homepage: www.elsevier.com/locate/rcim

A slack-diversifying nonlinear fluctuation smoothing rule for job dispatching in a wafer fabrication factory

Yi-Chi Wang, Toly Chen*, Chi-Wei Lin

Department of Industrial Engineering and Systems Management, Feng Chia University, No. 100, Wenhwa Road, Seatwen, Taichung 40724, Taiwan

A R T I C L E I N F O

Article history:
Received 23 November 2011
Received in revised form
17 April 2012
Accepted 29 April 2012
Available online 19 June 2012

Keywords:
Wafer fabrication
Dispatching rule
Slack
Diversify
Fluctuation smoothing

A B S T R A C T

This study proposes a slack-diversifying nonlinear fluctuation smoothing rule to reduce the average cycle time in a wafer fabrication factory. The slack-diversifying nonlinear fluctuation smoothing rule is derived from the one-factor tailored nonlinear fluctuation smoothing rule for cycle time variation (1f-TNFSVCT) by dynamically maximizing the standard deviation of the slack, which has been shown to improve scheduling performance in several previous studies. The effectiveness of the proposed rule has been validated via using it with a simulated data set. Based on the findings in this research we also derived several directions that can be exploited in the future.

1. Introduction

The manufacture of semiconductors is one of the most important high-tech industries due to their widespread applications. However, the product life cycle of new semiconductor products keeps getting shorter. As a result, the semiconductor manufacturers are faced with the pressure of having to meet the various needs of their customers within a shorter time span. Increasing the speed of product development, more agile production, quicker response, are some of the responses that are considered viable strategies. One common feature of these strategies is the compression of the cycle times of the related processes. Among the various types of cycle time, the production cycle time is particularly important because it determines the time of delivery to the customer. In other words, the shorter the production cycle time, the faster the delivery to the customer will be. Therefore, shortening the production cycle time through effective job dispatching is an important task [1]. Much research has been conducted concerning semiconductor shop floor control, especially in the domains of deterministic scheduling and job dispatching. However, Chen and Lin [2], Chen and Wang [3] and Chen [4] noted that job dispatching is very difficult task in a semiconductor manufacturing factory, theoretically, it is a NP-hard problem. In practice, many semiconductor manufacturing factories suffer from lengthy cycle times, and are not able to improve on their delivery promises to their customers.

Semiconductor manufacturing can be divided into four stages: wafer fabrication, wafer probing, packaging, and final testing. The most important stage is wafer fabrication. It is also the most time-consuming one. In this study, we investigated the job dispatching for this stage. This field includes many different methods, including dispatching rules, heuristics, data mining-based approaches [5,6], agent technologies [5,7–9], and simulation. Among them, dispatching rules (e.g. first-in first out (FIFO), earliest due date (EDD), least slack (LS), shortest processing time (SPT), shortest remaining processing time (SRPT), critical ratio (CR), the fluctuation smoothing rule for the mean cycle time (FSMCT), the fluctuation smoothing rule for cycle time variation (FSVCT), FIFO+, SRPT+, and SRPT++) all have received a lot of attention over the last few years [5–7] and are the most prevalent methods used in practical applications. For details on the traditional dispatching rules, please refer to Lu et al. [10].

Some advances in this field are as follows. Altendorfer et al. [11] proposed the work in parallel queue (WIPQ) rule targeting maximizing throughput at a low level of work in process (WIP). Zhang et al. [12] proposed the dynamic bottleneck detection (DBD) approach by classifying workstations into several categories and then applied different dispatching rules to these categories. They used three dispatching rules including FIFO, the shortest processing time until the next bottleneck (SPNB) and CR. Based on the current conditions in the wafer fabrication factory, Hsieh et al. [6] chose one approach from FSMCT, FSVCT, largest deviation first (LDF), one step ahead (OSA), or FIFO. Chen [13] modified FSMCT and proposed the nonlinear FSMCT (NFSMCT) rule, in which he smoothed the fluctuation in the estimated

* Corresponding author. Tel.: 886 4 24517250x3645; fax: 886 4 24510240.
E-mail address: tolychen@ms37.hinet.net (T. Chen).

0736-5845/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.rcim.2012.04.011

remaining cycle time and balanced it with that of the release time or the mean release rate. To diversify the slack, he applied the 'division' operator instead. This was followed by Chen [14], in which he proposed the one-factor tailored NFSMCT (1f-TNFSMCT) rule and the one-factor tailored nonlinear FSVCT (1f-TNFSVCT) rule. Both rules contain an adjustable parameter to allow them to be customized for a target wafer fabrication factory. In a multiple-objective study, Chen and Wang [15] proposed a bi-objective nonlinear fluctuation smoothing rule with an adjustable factor (1f-biNFS) to optimize both the average cycle time and the cycle time variation at the same time. More degrees of freedom seem to be helpful in the performance of customizable rules. For this reason, Chen et al. [16] extended 1f-biNFS to a bi-objective fluctuation smoothing rule with four adjustable factors (4f-biNFS). For a summary of these rules please refer to Table 1. One drawback of these rules is that only static factors are used, and they must be determined in advance. To this end, most studies (e.g. [13–16]) performed extensive simulations. This is not only time-consuming but it also fails to consider enough possible combinations of these factors. Chen [17] established a mechanism that was able to adjust the values of the factor in 1f-biNFS dynamically (dynamic 1f-biNFS). However, even though satisfactory results were obtained in his experiment, there was no theoretical basis supporting the proposed mechanism. Chen [18] attempted to relate the scheduling performance to the factor values using a back propagation network (BPN). If that would have worked, then the factor values contributing to the optimal scheduling performance could have been found. However, the explanatory ability of the BPN was not good enough.

At the same time, Chen [17] stated that a nonlinear fluctuation smoothing rule uses the divisor operator instead of the subtraction operator, which diversifies the slack and makes the nonlinear fluctuation smoothing rule more responsive to changes in the parameters. Chen and Wang [15] proved that the effects of the parameters are balanced better in a nonlinear fluctuation smoothing rule than in a traditional one if the variation in the parameters is large. In other words, magnifying the difference in the slack seems to improve the scheduling performance, especially with respect to the average cycle time. For these reasons, a slack-diversifying nonlinear fluctuation smoothing rule is used in this study for job dispatching in a wafer fabrication factory, in order to further shorten the average cycle time.

The slack-diversifying nonlinear fluctuation smoothing rule is modified from 1f-TNFSVCT by maximizing the difference in the slack measured with the standard deviation of the slack. The factor value for achieving this must be determined, which turns out to be a complex optimization problem. We applied two polynomial fitting techniques to convert it into a more tractable form for which several optimal solutions can be found. After screening some values out of the specified range, the remaining values were used to construct an optimized 1f-TNFSVCT rule.

The remainder of this paper is arranged as follows. Section 2 provides the details of the slack-diversifying nonlinear fluctuation smoothing rule. In Section 3, a simulated case is used to validate the effectiveness of the slack-diversifying nonlinear fluctuation smoothing rule. The performances of some of the existing rules in this field are also examined using the simulated data. Finally, we draw our conclusions in Section 4 and provide some worthwhile topics for future work.

2. Methodology

The variables are defined as follows:

(1) R_i: the release time of job i, $i=1\sim N$.
(2) BQ_i: the total queue length before the bottlenecks at R_i.
(3) CT_i: the cycle time (actual value) of job i.
(4) CTE_i: the estimated cycle time of job i.
(5) $D_i^{(l)}$: the delay of the l-th recently completed job at R_i, $l=1\sim 3$.
(6) FQ_i: the total queue length in the whole factory at R_i.
(7) Q_i: the total queue length on the processing route of job i at R_i.
(8) RCT_{ij}: the remaining cycle time (actual value) of job i since step j.
(9) $RCTE_{ij}$: the estimated remaining cycle time of job i since step j.
(10) SCT_{ij}: the step cycle time (actual value) of job i until step j.
(11) $SCTE_{ij}$: the estimated step cycle time of job i until step j.
(12) WIP_i: the factory WIP at R_i.
(13) SK_{ij}: the slack of job i at step j.
(14) U_i: the average factory utilization at R_i.
(15) α: $\max(R_i)-\min(R_i)$.
(16) β: $\max(RCTE_{ij})-\min(RCTE_{ij})$.
(17) γ: $N-1$.
(18) λ: the mean release rate.

It is evident that

$$CT_i = SCT_{ij} + RCT_{ij} \tag{1}$$

Table 1
Summary of some dispatching rules.

Rule name	Formula
1f-TNFSMCT	$SK_{ij} = \left(\frac{\beta}{\alpha(RCTE_{ij}-\min(RCTE_{ij}))}\right)^{\xi} \times (R_i - RCTE_{ij} + \xi(RCTE_{ij}-\min(R_i)))$
1f-TNFSVCT	$SK_{ij} = \left(\frac{\beta\lambda}{\gamma(RCTE_{ij}-\min(RCTE_{ij}))}\right)^{\xi} \times \left(\frac{i}{\lambda} - RCTE_{ij} + \left(RCTE_{ij} - \frac{1}{\lambda}\right)\xi\right)$
1f-biNFS	$SK_{ij} = \frac{((i/\lambda)-(1/\lambda)^{1-\xi}(R_i-\min(R_i))^{\xi}(RCTE_{ij}-\min(RCTE_{ij}))}{((N/\lambda)-(1/\lambda)^{1-\xi}(\max(R_i)-\min(R_i))^{\xi}(\max(RCTE_{ij})-\min(RCTE_{ij}))}$
4f-biNFS	$SK_{ij} = (R_i - RCTE_{ij} + (RCTE_{ij}-\min(R_i)) \times f_1) \times \alpha^{-f_2} \times \left(\frac{i}{\lambda} - RCTE_{ij} + (RCTE_{ij} - \frac{1}{\lambda}\right) \times f_3) \times \left(\frac{\gamma}{\lambda}\right)^{-f_4} \times \left(\frac{(RCTE_{ij}-\min(RCTE_{ij}))}{\beta}\right)^{-(f_2+f_4)}$
dynamic 1f biNFS	$SK_{ij} = \frac{((i/\lambda)-(1/\lambda))^{1-\xi(t)}(R_i-\min(R_i))^{\xi(t)}(RCTE_{ij}-\min(RCTE_{ij}))}{((N/\lambda)-(1/\lambda))^{1-\xi}(\max(R_i)-\min(R_i))^{\xi}(\max(RCTE_{ij})-\min(RCTE_{ij}))}$ $\xi(t) = \frac{1}{2}\left(\sin\left(\frac{\pi}{c}t\right)+1\right)$