Int J Adv Manuf Technol (2007) 35:596–609
DOI 10.1007/s00170-006-0723-z

ORIGINAL ARTICLE

Design of due-date oriented look-ahead batching rule in wafer fabrication

D. Y. Sha · Sheng-Yuan Hsu · X. D. Lai

Received: 11 September 2005 / Accepted: 10 July 2006 / Published online: 22 November 2006

Abstract In wafer fabrication processes, batch processing accounts for over 30% of the overall processing time. And it's a trade-off between machine utilization and wafer waiting time. Therefore, batch machines have become one of the constraint resources during wafer fabrication. How to maintain the utilization and reduce the waiting time are important tasks for production control. Plenty of research in the past several years focused on the dispatching rules of batch processing. According to many researchers, look-ahead batch dispatching rules outperform MBS on waiting times and machine utilization. The look-ahead batching rules that have been developed are DBH, NACH, MCR, and DJAH. However, these rules do not take the due-date information of wafers into consideration, and can't accelerate the wafer's fabrication that will not be completed before the due-date. This study will develop a due-date oriented look-ahead batching rule, namely LBCR, that considers the due-date and expects to raise delivery rates and reduce the average tardiness. Firstly, this study will modify those batching rules to fit the manufacturing environment of wafer fabrication. There are serial simulation tests on those batching rules under various kinds of factors in terms of environment, including traffic intensity, product numbers and product mix rate. Finally, this study will compare the five batching rules on different performance indicators. After the simulation and statistic analysis undertaken, LBCR does outperform other batch dispatching rules on due-date related performance indicators, such as tardy rate and average tardiness.

Keywords Batching rule · Simulation · Wafer fabrication

1 Introduction

A semiconductor chip is a complex device that consists of miniaturized electronic components and their connections. There are five steps in semiconductor manufacturing: wafer fabrication, wafer probe, device assembly, class test, and final test. Wafer fabrication is the process of the most technological complex and capital intensive. Since the required capital investment is extremely large, the implementation of an improved shop floor control strategy could save costs. However, it is challenging to develop the SFC strategy for a wafer fab due to the long flow time, ever-changing products yielded, re-entrant feature of the production sequence, and stochastic aspects of the wafer fab including machine failures. There are basically two types of SFC strategies at the wafer fabrication stage. The first and the most familiar one is dispatching. Each time a workstation is ready to commence processing another order and there is a queue of work-in-process (WIP) waiting to be processed at the workstation, the dispatching strategy selects an order or orders to start processing.

Generally there are two types of machines in the fab. Serial machines which process one wafer at a time and

D. Y. Sha · X. D. Lai
Department of Industrial Engineering and Management,
National Chiao Tung University,
Hsin-Chu, Taiwan, Republic of China

D. Y. Sha
Institute of Business Administration,
Asia University,
Taichung, Taiwan, Republic of China

S.-Y. Hsu (✉)
Department of Industrial Engineering and Management,
Chienkuo Technology University,
No.1, Chieh-Sou N. Rd.,
Changhua 500, Taiwan, Republic of China
e-mail: jackhsu@ctu.edu.tw

batch machines which work on multiple wafer/lots. Most of the machines are belonging to a series machine including the bottleneck machines, photolithography. Batching machines always are critical machines, like heat-treating ovens, plating baths, kilns for drying lumber, and in semiconductor wafer fabrication, diffusion and oxidation ovens. Often the maximum batch size (machine capacity) of such operations is greater than the size of the arriving lots.

Scheduling such a complex facility, including serial and batching machines is a challenge. The dispatching methodologies on serial machine are discussed in many research works. There are hundreds of rules relating to dispatching on the serial machine, especially on e thbottleneck workstation. In wafer fabrication, most of the dispatching researches are focused on the serial bottleneck workstation, as well as photolithography. For the batch machines which are not the bottleneck in fab, MBS (minimum batching size) is always used to deal with the dispatching tasks on batch machines. Just a few researchers have interest in it. However, batch machines interact with serial machine in the production system. System performance should be affected by batch dispatching rules. Besides, batch machines, such as furnace, are critical resources in fab. That is, the utilization of batch machines is greater than the average utilization of the production system. How to increase the utilization and eliminate the extra waiting time are important tasks. Two major decisions regarding batch dispatching should be taken into account. The first one is how many lots should be accumulated to process an increase in the utilization of batch machine and the production efficiency. The second is the sequence of different product types waiting in front of the workstations. It's more difficult than serial dispatching.

The semiconductor manufacturing industry is known as a highly competitive business. Customer service is very important because unsatisfied customers can easily leave for competitors which will cause a significant loss. Due-date is one of the important factors to measure production service. However, most of the batch dispatching rules don't consider due-date related performance in their decision algorithms. To model a stable batch dispatching rule for increasing of the customers' satisfactions is the targeted objective in this research.

The remainder of this paper is organized as follows. The second section will summarize the relevant literature on wafer fabrication and batch dispatching rules. The third section will discuss some major batch dispatching rules and describe the modeling of the batch dispatching rules. The fourth section will describe the simulation model and the experiment design. In the fifth section, the simulation results will be presented. The conclusions and suggestions for future work will be included in the last section.

2 Literature review

Uzsoy et al. [1] found the following factors that made production planning and scheduling in semiconductor industry particularly difficult.

1. Complex product flows,
2. Random yield,
3. Diverse equipment characteristics,
4. Equipment downtime,
5. Production and development in shared facilities,
6. Data availability and maintenance.

Due to the above factors, it's important to exercise reasonable control over the production environment. Uzsoy et al. [2] pointed out that the researches on SFC in wafer fab are focused on order review/release (ORR) and dispatching strategies.

There are two types of machines in the wafer fabrication in general. Serial machines which proceed one wafer at a time and batch machine can serve multiple wafer or lots at the same time. In Van der Zee et al. [3], it is shown how to control strategies for batch machine which may be classified according to the amount of information which is known upon future arrivals of jobs. Three typical situations can be distinguished:

1. Information unavailable;
2. Full knowledge of future arrivals;
3. A limited number of near future arrivals known or predicted.

The first category of control strategies concerns those strategies that base their decision on local information only. The most important example of such a strategy is the minimum batch size (MBS), which is introduced by Neuts [4]. The batching machine starts service only when some minimum number of jobs are presented in the queue. Deb and Serfozo [5] showed how this critical load should be chosen in order to minimize the expected discounted cost over an infinite horizon. If the cost of serving is set to zero and the cost of waiting is linear, minimizing the expected averaged cost is equivalent to minimizing the average flow time. An overview of the type of strategies which assumes zero information on future arrival is given by Uzsoy et al. [2]. An adapted version of the MBS, which covers the multiple products, multiple machine case, MBSX, is introduced by Weng and Leachman [6] and Van der Zee et al. [3].

While the above types of strategies are used without information relating to future jobs, full knowledge of future arrivals is supposed to be available when it comes to determining machine scheduling. The relevance of this type of models is quite limited because only a little information on future arrivals is available in practice Van der Zee et al. [3].

Nowadays the information related to future arrival jobs can be forecasted based on the advanced information technology. Most of the researches are focused on the third category of strategies. Glassey and Weng [7] were first to introduce these types of strategies in wafer fabrication, which are characterized by the fact that they assumed a few near future arrivals to be known and/or predicted. They present a dynamic batching heuristic (DBH). DBH decides when to start a production cycle thereby aiming for a minimal average flow time. Fowler et al. [8] extended the DBH rule to consider the multiple product types case. Their next arrival control heuristic (NACH) probes to be a robust heuristic in case forecasting data on future arrivals are used, i.e., estimated arrival moment for new lots. In Fowler et al. [9], NACH had been extended to the more general multiple products-multiple server case. NACH used only the predictions of future arrivals times to batch server for the decision of whether or not to start processing a batch. The amount of waiting time for lots already in queue caused by waiting for future arrivals is compared to the amount of waiting time that can be saved for the future arrivals by waiting until the arrivals occur to start a batch. It does not consider the information related to due date. The simulation model they used only focused on the diffusion area. However, the diffusion area is only one of the critical workstations in wafer fab. The other critical resources, like photolithography, will have some effects on the performance.

Weng and Leachman [6] showed how performance can be improved for the multiple products and single machine case by their minimum cost rate heuristic (MCR). In two articles by Van der Zee et al. [3, 10] introduced the dynamic job assignment heuristic (DJAH). It covers the multiple product case and allows for compounding arrivals. The criterion for optimization for DJAH is the minimization of logistic costs per order (customer) on the long term. Although DJAH heuristic proved its strength as a control strategy for multiple identical machines, it is relatively unsuitable for those situations where alternative machine types are available. Van der Zee et al. [11] developed a new strategy, namely dynamic scheduling heuristic (DSH), to choose for different types of machines based on the required processing condition, product characteristics, or operation cost. DSH was strongly focuses on finding a good fit of machine and product characteristics.

In Neale and Duenyas [12], the control of manufacturing networks consisting of a batch processing machine and one or more unit-capacity machines in tandem are considered. Algorithms based on dynamic programming are presented to determine the optimal policies for the infinite/finite, deterministic/stochastic problem respectively. It presented polynomial time dynamic programming algorithms which minimize the sum of the completion times for serial-batch ($\delta \rightarrow \beta$) and batch-serial ($\beta \rightarrow \delta$) systems with deterministic release times and processing times. A heuristic policy, namely two control limit heuristic (TCLH), has been mentioned. In their research, only two simple systems ($\delta \rightarrow \beta$ and $\beta \rightarrow \delta$) are tested and only local performance are discussed.

Kim et al. [13] focused on production scheduling in a semiconductor wafer fab with multiple product types that have different due dates and different process flows. Three kinds of decision on production scheduling have been considered including lot release control, lot scheduling, and batching scheduling. Three batching rules are included, that is MMBS, MDBH, and PUCH, for testing the production performance. MMBS is modified from MBS. For the selection of a product family to be processed next, the average slack time of lots of each family is computed and a family with the least average slack time is selected. If the number of waiting lots of the family in the queue is greater than or equal to a predetermined value, the lots are grouped into a batch to be processed right away. Otherwise, processing for the selected family is deferred until the number of waiting lots becomes equal to the MBS. If there is no family to be processed at the present time, the machine must wait until a new lot (or lots) arrives at the queue for the machine, when a new scheduling decision is made.

MDBH is modified from dynamic batching heuristic (DBH) [7]. In MDBH, a family to be processed next on an available machine is selected using the average slack time of lots of each family; a family with the least average slack time is selected. Note that due date of lots are not considered in DBH. Once a family to be processed is selected, whether to start processing a lot in the selected family is to be determined. For this decision, two alternative points of time are considered: the current time (when the scheduling decision is made) and the time point when a new wafer lot (or lots) arrive at the workstation. For each alternative, they compute waiting times of the lots currently waiting in the queue and those that will arrive next and check whether the total weighted waiting time of the lots will be increased (or decreased) if processing of the available lots is started without delay rather than deferred until a new lot(s) arrives. The decision of the start time is made using the result of this comparison.

PUCH is the abbreviation of processing urgency classification heuristic. A new criterion is defined for batching decision, namely processing urgency. The urgency of a wafer lot is measured with slack time per remaining work of the lot. In this heuristic, families are classified into three classes according to the processing urgency. Families in class A are those that contain very urgent lots that have to be processed immediately or as soon as possible. Class-B families are those that do not contain very urgent lots but contain moderately urgent lots. The remaining families are classified as class C. When a machine becomes available, if there are families of class A, a family which contains the most urgent lot is selected and processed at once. If there is no class-A family, families in class-B are considered. If there is any class-B family for which the number of waiting lots is greater than or equal to MBS, a family that contains the most urgent lot among them is selected and processed. Otherwise, the class-C families in which the number of waiting lots are greater than or equal to the MBS are considered. The one with the most waiting lot will be selected. If the number of the waiting lots in the selected families is greater than the maximum batch size of the machine, lots with shorter slack times are selected first.

The batching rule developed in their research, PUSH, is focused on processing urgencies of lots and the number of waiting lots in the queue. It doesn't care about the information related to upstream and downstream workstation.

In Van der Zee et al. [14], a new dynamic scheduling heuristic (DSH) considering the non-identical batching machine has been developed. DSH is extended from DJAH, which covers the case of multiple identical machines. DSH takes due notice of both machine characteristics and machine availability. Therefore it built a schedule using a two-phase approach. First machine/product combinations are selected, given a maximum throughout criterion. In the second phase it is decided whether-or-not an available machine should be started.

Cigolini et al. [15] proposed a new dynamic look-ahead approach based on the standpoint of the wait no longer than time (WNLTT). It allowed easy modeling of real life manufacturing environments where there are parallel batching machines, various products with different batch processed, ordering expediting and re-entrant loops along the job routings. Its simulation model is too simple to test the serial-batch manufacturing system with re-entrant. Besides, the main focus is reducing the mean flow time and maintaining the utility of the batch machine. Due date oriented performance is ignored. Van der Zee [16] focused on a normal batch-serial flow shop system with one batch machine and one or four serial machines. The new dynamic job assignment heuristic for real-time flow shop control (DJAH-F) is partly based on the DJAH. DJAH only addressed the isolated problem of controlling batch operations [3]. DJAH-F only focused on a simple flow shop system. In Neale and Duenyas [17], they considered the control of a single batch processing machine with random arrivals, random processing times, and compatible job families. All jobs belonging to the same family have the same processing requirements and can be processed together in the same batch. However, in wafer fabrication, diffusion is primarily a chemical process and only lots with the same recipe can be processed together in the same batch. Besides, it only focused on the performance of the local batch machine. The optimal solution can only be computed with exponential inter-arrival and processing times and only two or three families. Heuristic method, weighted processing rate for compatible families (WPRC). They use some numerical samples to make a computational test, but the system used in their research is too simple and only local performance is discussed in their research.

Most of the research works mentioned above are mainly focused on the improving of production performance to develop their batch dispatching rules. How to decrease the waiting time and increase the utilizations of machines are major objectives. They only concern the improving of system-related performance. However, delivering goods to the customer on time will enhance the customer's satisfaction and strengthen competitive advantage. The due-date related performances are more important than system related. So this research tries to model a new batching dispatching rule considering due-date and processing time, namely due-date oriented look-ahead batching rule (LBCR). LBCR will combine with the methodologies of the dispatching rule, CR, and batching dispatching rule, NACH, for increasing the rate of on-time delivering and decreasing job's waiting time.

3 Modeling of batching rules

3.1 Types of the batch dispatching rule

Many batching rules have been developed over recent years. Due to pages constrain, we are not able to discuss all of them in this research. We will review and introduce a classification framework of the main research work about batching in job shop.

Van der Zee et al. [3] introduced a review and a classification framework of the main research works about batch dispatching rule in job shop. We will extend their classification framework to model an analytic list (Table 1). Those representative symbolic batching rules discussed in this research will be chosen from the list.

MBSX will be included as a benchmark. DBH is the fist rule to use the upstream information. DBH uses the jumping decision which can be made at any time phase. It can save much effort on trial-and-error. However, DBH can't be used on the case of multiple product. MCR is adopted, its methodology is extended from DBH and the performance is better than DBH.

NACH's rolling time phase can reduce the effect from forecasting error to future arrival jobs. MCR is the first rule which introduced a cost function considering multiple factors for decision making. DJAH is suitable for multiple product case. So we will discuss NACH, MCR, and DJAH in this research. Robinson et al. [18] has tested the RHCR and RHCR-S. Their performances are not better than MCR. That is the batching rules considering down-stream information which can't improve the system performance. We will then not adopt them. Therefore, we will test these four

Table 1 Analysis of the batch dispatching rules

Rule	No. of machines and products	Information				Decision making				
		Arriving order	Downstream queue	PT	Due date	WT	Holding cost	Average CR	Objective	Forecasting area[17]
MBSX[1]	(M,N)								FT	
DBH[2]	(M,1)	V		V		V			FT	kA, T
NACH[3]	(1,N)	V		V		V			FT	1A
MCR[4]	(1,N)	V		V		V	V		FT, Cost	kA, T
RHCR[5]	(1,N)	V		V		V	V		FT, Cost	kA, T
RHCR-S[6]	(1,N)	V	V	V		V	V		FT, Cost	kA, T
DJAH[7]	(M,N)	V		V		V	V		FT, Cost	1A,1M
DJAH-F[8]	(1,N)	V		V		V	V		FT, Cost	1A,1M
WNLTT[9]	(M,N)	V		V		V			FT	1A,1M
MMBS[10]	(M,N)			V	V				FT	1A,1M
MDBH[11]	(M,N)	V		V		V			FT	1A,1M
PUSH[12]	(M,N)	V		V	V	V			FT	1A,1M
TCLH[13]	(1,1)	V		V		V			FT	1A,1M
WPRC[14]	(M,N)	V		V			V		FT	1A
DSH[15]	(M,N)	V		V		V	V		FT, Cost	1A,1M
LBCR[16]	(M,N)	V		V	V	V		V	FT	1A,1M

1. MBSX: adapted minimum batch size rule [3]
2. DBH: dynamic batching heuristic [7]
3. NACH: next arrival control heuristic [8]
4. MCR: minimum cost rate heuristic [6]
5. RHCR: rolling horizon cost rate heuristic [18]
6. RHCR-S: considering downstream information in RHCR's cost rate function [18]
7. DJAH: dynamic job assignment heuristic [3, 10]
8. DJAH-F: new dynamic job assignment heuristic for flow shop [16]
9. WNLTT: wait no longer than time [15]
10. MMBS: modified minimum batch size rule [13]
11. MDBH: modified dynamic batching heuristic [13]
12. PUSH: processing urgency classification heuristic [13]
13. TCLH: two control limit heuristic [12]
14. WPRC: weighted processing rate for compatible families heuristic [17]
15. DSH: dynamic scheduling heuristic [14]
16. LBCR: due date oriented look ahead batching rule (this research)
17. T : processing time, 1A : the next one job will arrive at the batching machine, kA : the next k jobs will arrive at the batching machine, 1M : at the moment of one machine is ready for use

existing batching rules in this research, including MBSX, NACH, MCR, and, DJAH. Those algorithms will be described in the next section.

3.2 Batch dispatching rules in wafer fabrication

The following notation will be used to illustrate those batching rules:

C	The oven capacity
C^j	The oven capacity of a product *j*.
DD_i	The due-date of an order *i*.
N	The number of a product type.
t_0	The time epoch that the oven is idle and the number in the queue is positive.
t_i	The arriving epoch of the next *i*th lot after t_0.
t_{ij}	The arriving epoch of the next *i*th lot after t_0 of product *j*.
$TC(t_i)$	Total holding cost from t_0 to $(T+t_i)$ if the furnace is loaded at t_i.
$CR(t_i)$	Holding cost per unit time (cost rate)between t_0 and $(T+t_i)$ if the furnace is loaded at t_i.
q	The number of lots in queue (single product case).
q_j	The number of lots in queue of the product *j*.
T	Processing time at batching time.
T_j	Processing time at batching time of product *j*.
J	The set of product type identifiers j.
D_j	Total delay in case product *j* is loaded.
SN	The set of product types for which it is not worthwhile to wait for a next arrival.
SW	The set of product types for which it is worthwhile to wait for a next arrival.
W_j	The total delay experienced by the other products during processing of the products of type j $\left(T_j \sum_{i \neq j} q_i\right)$

3.2.1 MBS and MBSX

The minimum batching size (MBS) rule means that a number is picked such that an operation is started when the number of waiting lots is greater than or equal to a predetermined number (B). That is to say if the machine is idle and the number of waiting lots is smaller than B, these lots will not be processed. The possible value of B is greater than or equal to 1 and is smaller than or equal to the oven capacity. Van der Zee et al. have modified the MBS for multiple products and multiple machines, namely MBSX. The decision rule is "When there are more than one types of products and the queues are greater than or equal to B behind the oven, the one with the longest waiting time will be chosen . If there with more than one candidate, the one with the shortest processing time will be proceeded. If there are more than one with the same waiting time and processing time, the decision will be made randomly."

3.2.2 NACH

Next arrival control heuristic (NACH) is developed by Fowler et al. [8]. It was proved to be a robust heuristic in the case where forecasting data on future arrivals was used, i.e., estimated arrival moment for new lots. The decision rule of NACH in single product and single machine is:

If $q \geqq C$ then loading the jobs.

Else if $[q(qt_1 - t_0) - (t_0 + T - t_1)] < 0$ (i.e., $\mathrm{NACH}_1 < 0$) then waiting for next arrival job.

Else loading the jobs.

NACH is derived from DBH. The difference is NACH only considers the next arriving job. There are two choices at time t_0, loading the jobs or waiting for another new job. Waiting for the next new arrival job is represented for a new decision phase.

The NACH heuristic which addressed the multiple products and single machine greatly increases the complexity of the decision process. It's due to the fact that a decision made for one product which can have a major impact on the delays experienced by other lots of different product types. We will describe it as below:

Situation 1.

The oven becomes idle.

If full loads are available then select a product $j^* = \underset{q_i \geq C_j}{\arg\min} W_j \left(W_j = T_j \sum_{i \neq j} q_i \right)$ and load the oven.

Else evaluate $NACH_1(j)$ for all j, $\left(NACH_i(j) = \sum_{i=1, i \neq j}^{N} \left[q_i \left(t_{1,i} - t_0 \right) - \left(t_0 + T_j - t_{1,i} \right) \right] \right)$

If $j \in SW$ then wait (i.e., $NACH_1(j) < 0$, for all j=1 to N)

Else if $j \in SN$ (i.e., $NACH_1(j) \geqq 0$, for all j=1 to N) then select $j^* = \underset{j \in J}{\arg\min} W_j$

Else define

$$\mathrm{D_j} = \mathrm{W_j} + \sum_{i=1}^{N} \max\left(0, t_0 + T_j - t_{1,i}\right) \quad j \in SN$$

$$\mathrm{D_j} = \sum_{i=1}^{N} \left(q_i \left(t_{1,j} - t_0 \right) \right) + W_j + \sum_{i=1}^{N} \max\left(0, t_{1,j} + T_j - t_{1,j}\right)$$

$j \in SW$

Select a product type $j^* = \underset{j \in J}{\arg\min} D_j$

If $j^* \in SN$ then load the oven

Else wait.

Situation 2.

The oven is idle and a product type j arrives, then proceed as indicated by $NACH_1(j)$.

3.2.3 MCR

Minimum cost rate heuristic (MCR) is modeled by Weng and Leachman [6]. The major difference is the choice of horizon for look-ahead. It uses processing time plus any prior waiting time as the scheduling horizon.

Suppose that the furnace is idle at time epoch t_0. No decision needs to be made if there is nothing to be done. Therefore, we are only interested in the case where $0 \leqq q \leqq C$. Otherwise, a furnace cycle should start immediately with a full load. Then, the decision to be made is when to start the next furnace cycle. The start time can only be at the current time, t_0, or at furnace arrival epochs which are assumed to be known. We will call epochs $\{t_0, t_1, ..., t_n\}$ possible loading epochs, where $n = \max\{0, C - q\}$. If the operation starts at t_i, the furnace will not be free again until $(t_i + T)$. At time t_i, $(q+i)$ lots will be loaded into the furnace, where $(q+i) \leqq C$. The scheduling horizon is $(t_i + T - t_0)$, during which all lots except for those in process are waiting. The total holding cost experienced during this period is

$$TC(t_i) = q(t_i - t_0) + \sum_{t_0 < t_d < t_i} (t_i - t_d) + \sum_{t_i < t_d < t_i + T} (t_i + T - t_d) \tag{1}$$

Where $q \leqq C$.

The first two terms account for delay to lots arriving in the queue before the furnace cycle starts; the last term defines delays to lot arriving after the cycle starts up until the end of the cycle. Therefore, the holding cost per unit time in the scheduling horizon, i.e., the cost rate, is

$$CR(t_i) = \frac{TC(t_i)}{t_i + T - t_0} \tag{2}$$

The scheduling algorithm for single product is:

If $q \geqq C$ then load the ovens.

Else wait until lot k has arrived before loading the oven with $\mathrm{k} = \underset{0 \leq i \leq i_{\max}}{\arg\min} CR(t_i)$, Where $i_{\max} = \max\{0, c-q\}$.

The cost rate function (Eq. 2) can be extended to multiple products. When a particular product is loaded, all of the other products are waiting. The best loading time for each product according to the minimum cost rate of all the possible loading times can be found, and then the product with the minimum cost rate for the next furnace operation will be chosen.

If full loads are available then select a product

$$j^* = \underset{q_j \geq C_j}{\arg\min} \frac{\sum_{i=1}^{N} TC_{i,j}(t_0)}{T_j} \text{, and load the oven.}$$

Else wait until k* product of type j* have arrived before loading the oven with

$$(j^*, k^*) = \underset{\substack{(j,k): j \in J \\ 0 \leq k \leq C_j - q_j}}{\arg\min} \frac{\sum_{i=1}^{N} TC_{i,j}\left(t_{k,j}\right)}{t_{k,j} + T - t_0} \tag{3}$$

$$\mathrm{TC_{i,j}(t)} = \begin{cases} \mathrm{T_j} \times \max\left(q_j - C_j, 0\right) + \sum\limits_{t_0 < t_{d,j} < t_0 + T_j} \left(t_0 + T_j - t_{d,j}\right); t = t_0, i = j \\ q_j(t - t_0) + \sum\limits_{t_0 < t_d < t} \left(t - t_{d,j}\right) + \sum\limits_{t < t_d < t + T_j} \left(t + T_j - t_{d,j}\right); t > t_0, i = j \\ q_j\left(t + T_j - t_0\right) + \sum\limits_{t_0 < t_{d,j} < t + T_j} \left(t + T_j - t_{d,j}\right); t \geq t_0, i \neq j \end{cases} . \tag{4}$$

In Eq. (3), a distinction is made between the cost functions (TC) in case the oven is loaded at t_0 with a product of type j, and a situation in which the oven is loaded at a next arrival ($t_{i,j}$). The waiting costs for the other products are computed according to the last cost function. The product which is associated with a minimum value for average queue length (j^*) is chosen to be loaded next. Loading may be instantly ($k^*=0$), but may also require waiting for a future arrival ($k^*>0$).

3.2.4 DJAH

DJAH is developed by Van der Zee et al. [3] to combine with those advantages of MCR and NACH. DJAH, a new look-ahead batching rule, is suitable to multiple product types and multiple machines. The decision rules are described as the following: If any $q_j \geqq C_j$ then chose product j*, $j^* = \underset{q_j > C_j}{\arg\min} \frac{TC_j(t_0)}{C_j}$.

Else if $\underset{\substack{j=1\cdots N \\ q_j > 0}}{\min} \frac{1}{q_j} TC_j(t_0) > \underset{j=1\cdots N}{\min} \frac{1}{q_j+1} TC_j\left(t_{1,j}\right)$ then wait for the next decision point.

Else chose product j*, $\mathrm{j}^* = \underset{j=1\cdots N}{\arg\min} \frac{TC_j(t_0)}{q_j}$,

Where $TC_j(t_0) = (H_j^0 - t_0) \max(q_j - C_j, 0) + (H_j^0 - t_0) \sum_{i \neq j}^{N} q_i + \sum_{i=1}^{N} \sum_{t_0 < t_k < t_0 + T_j} (H_j^0 - t_{k,j})$ $TC_j(t_1, j) = q_j(t_{1,j} - t_0) + (H_j^1 - t_0) \sum_{i \neq j}^{N} q_j + \sum_{i=1}^{N} \sum_{t_0 < t_k < H_j^1} (H_j^1 - t_{k,j}) - \max(H_j^1 - t_{1,j}, 0)$ $a_{\min} = \underset{t'_a = t_0}{\min} a$, $H_j^0 = \min(\underset{a \neq a_{\min}}{\min} t'_a, t_0 + T_j,)$ $H_j^1 = \min(\underset{a \neq a_{\min}}{\min} t'_a, t_{1,j} + T_j)$.

In the above equation, spending ranges, (H_j^0, H_j^1), are used for estimated remaining processing time (RPT) of each machine. The product j has been chosen from the previous equation and will be processed at the machine that has the minimum RPT. If the decision is "loading the job j*" at its arrival time $t_{1,j}$", there will be no extra waiting time. Hence, the equation, $\max\left(H_j^1 - t_{1,j}, 0\right)$, is try to find the suitable machine for product j*.

3.2.5 LBCR

Most of the look-ahead batching dispatching rules above-mentioned focused on decreasing the job's flow time and its holding cost. They only consider the job's processing time and arriving time. If the decision making only considered the processing time, the due-date will be ignored. Due-date related performance will be deteriorated. Delivering the goods to customers on time will enhance customer's satisfaction and competitive advantage. Therefore, due-date-related performances are more important than system-related performance. So this research tries to model a new batching dispatching rules concerning due-date and processing time, namely due-date oriented look-ahead batching rule (LBCR). LBCR will combine with the methodologies of the dispatching rule, CR, and batching dispatching rule, NACH, for increasing the rate of on-time delivering and decreasing the job's waiting time.

CR can consider the job's processing time and due-date. When jobs have the same due-date, the one with the longer remaining processing time will be considered first. NACH can improve the system performance based on little information. NACH used the rolling horizon to consider the next arrival job. It is steady and outstanding when there is some inaccuracy of the future job's forecasting [8].

The scheduling algorithm of LBCR at single machine and single product are (Fig. 1):

If $q>C$ then loading the job.

Else if $ACR(t_0)<$ Critical Value (CV) then load the job at t_0.

Else If $ACR(t_1)<$ CV then load the job at t_1. Else $ACR(t_0) \geqq ACR(t_1)$ then load the job at t_0.

Else the decision will be postponed to t_1.

The algorithms of $ACR(t_0)$ and $ACR(t_1)$ are:

$$ACR(t_0) = \frac{\sum_{i=1}^{q} CR_i + \frac{\left(DD_{q+1} - RPT_{q+1} - (t_0+T)\right)}{RPT_{q+1}}}{q+1}$$

$$ACR(t_1) = \frac{\sum_{i=1}^{q+1} \frac{(DD_i - RPT_i - t_1)}{RPT_i}}{q+1}$$

When the production system has multiple product types, the algorithms should be modified. LBCR used at multiple product type and single machine are described as the following:

If there are any product $q_j>C_j$

then load one of those job (used the product selection rule to chose)

Else if there are any product $ACR_j(t_0)< CV$

then load one of those job (used the product selection rule to chose) at t_0.

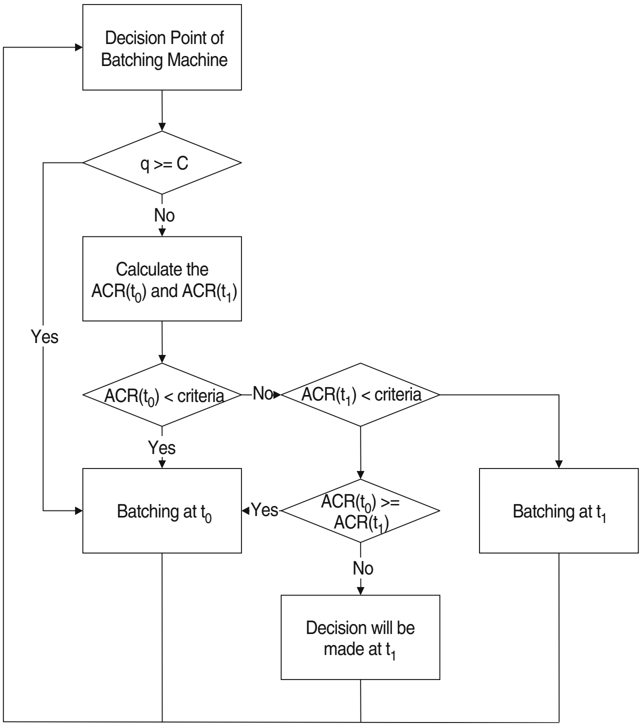


Fig. 1 Flowchart of LBCR at single product and single machine

Else if there are any product $ACR_j(t1_1)<$ CV

then load one of those jobs (used the product selection rule to chose) at t_1.

Else if the $Max(WACR_j)$ belongs to SN (i.e., $NACH_1(j) \geqq 0$, for all j=1 to N) then load the product j at t_0.

Else the decision will be postponed to t_1.

The algorithm of $WACR_j$ is

$$\mathrm{WACR_j} = \frac{\sum_{k=1,k\neq j}^{N} \sum_{i=1}^{q_k} \frac{\left(DD_i - RPT_i - \left(t_0 + T_j\right)\right)}{RPT_i} + \sum_{i=1}^{N} \frac{\left(DD_i - RPT_i - \left(t_0 + T_j\right)\right)}{RPT_i}}{\left(\sum_{i=1}^{N} q_i - q_j\right) + N}, j \in SN$$

$$\mathrm{WACR_j} = \frac{\sum_{j=1,k\neq j}^{N} \sum_{i=1}^{q_k} \frac{\left(DD_i - RPT_i - \left(t_{1,j} + T_j\right)\right)}{RPT_i} + \sum_{i=1}^{N} \frac{\left(DD_i - RPT_i - \left(t_{1,j} + T_j\right)\right)}{RPT_i}}{\left(\sum_{i=1}^{N} q_i - q_j\right) + N - 1}, j \in SN$$

The product selection rules are:

1. Calculate the $WACR_j$,
2. Select the one with the Max. $WACR_j$,
3. If there are more than one with Max. $WACR_j$ select the one which has the smallest $ACR_j(t_0)$,
4. If both of $WACR_j$ and ACR_j are the same, select one randomly.

3.3 The environmental factors related to batching rules

We use a simulation package to build a virtual fab for testing those batching rules above-mentioned. For testing batching rules' performance and their stability, some factors associated with production environment should be modeled and discussed in the simulation model. There are many

Table 2 The analysis of environmental factors

Factors	Glassey	Fowler	Wein	Robinson	Van der Zee	Significance
Interval arrivals		V			V	V
Orders forecasting accuracy				V	V	
Product types		V			V	V
Product mix		V		V	V	V
Lot size		V		V	V	
Variance of processing time		V		V	V	V
Number of machine					V	
Traffic intensity	V	V	V	V	V	V

environmental factors used in those researches related to batching rule in recent years (Table 2). The factors that have significant effect under statistical analysis, including traffic intensity, product types, and product mix, will be included in our research.

4 Simulation model and experiment design

For testing those batching rules, a virtual wafer fabrication system was modeled based on a real wafer fab. The wafer fab configuration considered in this study is a wafer fabrication factory in Taiwan. It was used in Hsu and Sha's [19] research for testing the dispatching and order releasing strategies. The fab consists of 53 workstations and 301 machines. The processing time for a lot is randomly generated from a uniform distribution between 0.95×MPT and 1.05×MPT, where the MPT (mean processing time) is given for each workstation. The setup time is included in the processing time. The virtual fab takes into account the downtime, which includes unscheduled breakdowns. The time between failure and repair for each workstation is randomly generated from exponential distributions with given mean values. A lot (a cassette for wafers) contains 24 wafers and the transferring time between workstations is ignored in the simulation. The dispatching rule at the serial machine is FIFO. Order releasing control is based on POISSON rule, the releasing rate (λ) is calculated by the traffic intensity (ρ), one of the experiment factors. The equation of traffic intensity is:

$$\rho = \lambda \sum_{j=1}^{N} \frac{S_j T_j}{M C_j}$$

S_j rate of product type *j*.
T_j processing time on bottleneck workstation of product type j.
C_j maximum production capacity on bottleneck workstation of product type j
M quantity of bottleneck machines.

The virtual wafer fab was built on personal computers with Pentium III 800 processors using the eM-plant, a simulation package developed by Tecnomatix Technologies Corp.

Table 3 Level of environmental factors

Production status	Environmental factors		
	Traffic intensity	Number of product type	Product mix
S1	0.6	2	1:1
S2	0.6	2	3:1
S3	0.6	3	1:1:1
S4	0.6	3	3:1:1
S5	0.9	2	1:1
S6	0.9	2	3:1
S7	0.9	3	1:1:1
S8	0.9	3	3:1:1

Table 4 The P-value of four factors ANOVA

Factor	Tardy rate	Tardiness	Avg. FT	Throughput
1*	0	0	0	0
2	0	0.9491	0	0.2131
3	0	0	0	0
4	0	0	0	0
12	0.1264	0.0109	0.9482	0.0447
13	0	0.0022	0	0
23	0	0.2022	0.0197	0.0118
14	0	0	0	0
24	0	0.0195	0.0121	0.0005
34	0.0007	0	0	0
123	0	0	0.0054	0
124	0	0.3956	0	0
134	0	0.0015	0.0009	0.0001
234	0	0.0009	0	0
1234	0	0.0228	0.1161	0

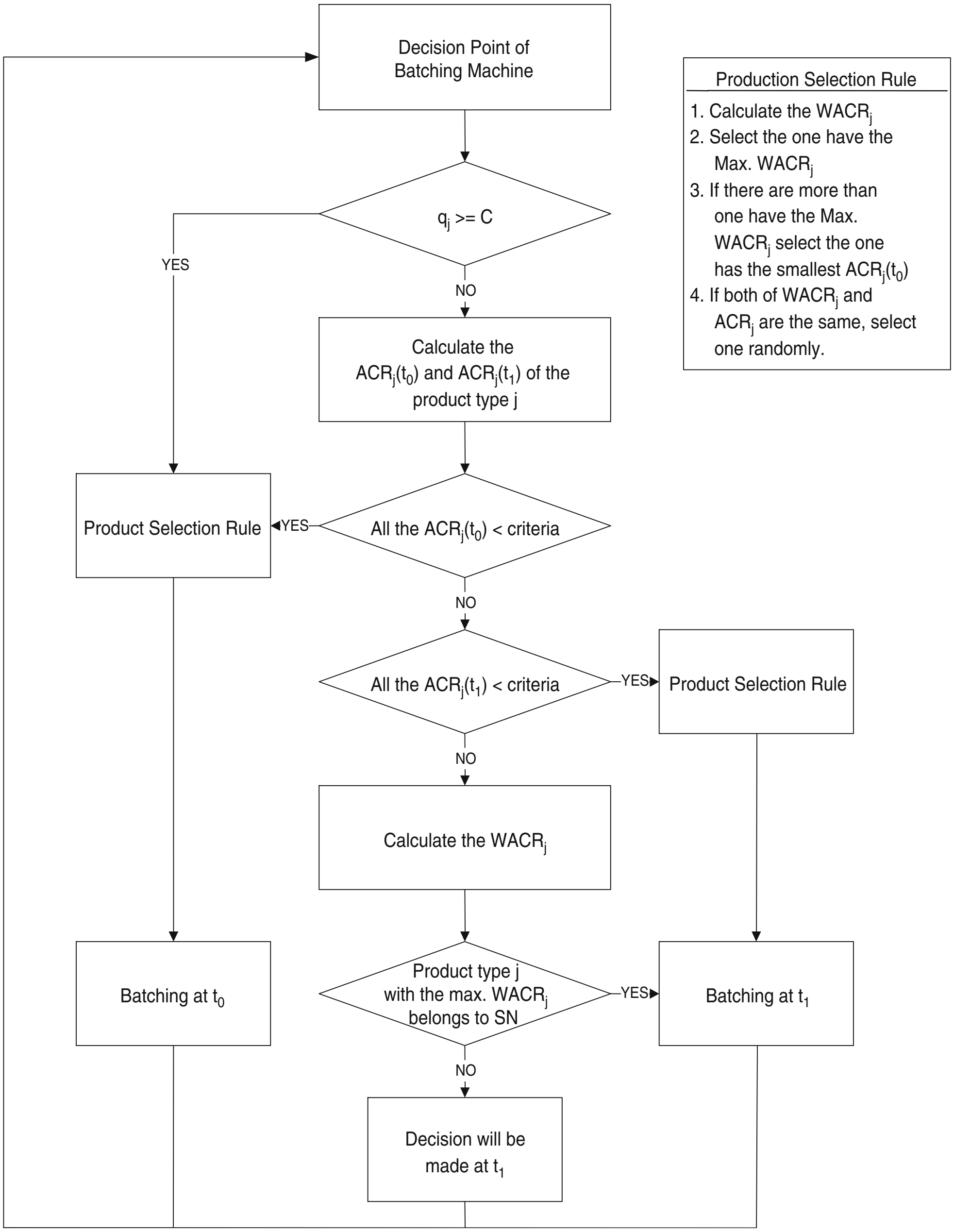
*The number represented the environmental factors. (1: traffic intensity, 2: number of product types, 3: product mix, 4: batch dispatching rule)

Table 5 The best dispatching rule of Duncan's test

Production status	Tardy rate	Tardiness	Avg. FT	Throughput
S1	1*3 4 5 2	5 2 3 4 1	1 2 3 4 5	3 2 4 5 1
S2	5 4 1 2 3	5 4 3 2 1	5 4 2 3 1	5 4 3 2 1
S3	5 1 2 3 4	4 1 5 2 3	5 4 3 2 1	5 4 3 2 1
S4	5 3 4 1 2	5 4 3 2 1	5 4 3 2 1	3 5 4 2 1
S5	5 4 3 2 1	5 4 3 2 1	4 2 3 5 1	4 2 3 5 1
S6	5 4 2 3 1	5 4 3 2 1	4 3 2 5 1	4 3 2 5 1
S7	5 4 3 2 1	5 4 3 2 1	4 2 3 5 1	4 2 3 5 1
S8	5 4 3 2 1	5 4 3 2 1	4 3 2 5 1	3 2 4 5 1

There are five batching rules applied in the simulation experiment, including MBSX, NACH, MCR, DJAH, and LBCR. Three environmental factors will be considered in the experiment design, i.e., traffic intensity, product mix, and product types. Traffic intensity is represented by the utilization of production system. In Van der Zee's research, there are three levels of traffic intensity, low (0.3), middle (0.6), and high (0.9). however, the equipments of wafer fabrication are expensive, and to maintain a high level of utility is very important. Low level of traffic intensity is not applied in practice. So we will

Fig. 2 Flowchart of LBCR at multiple products and single machine



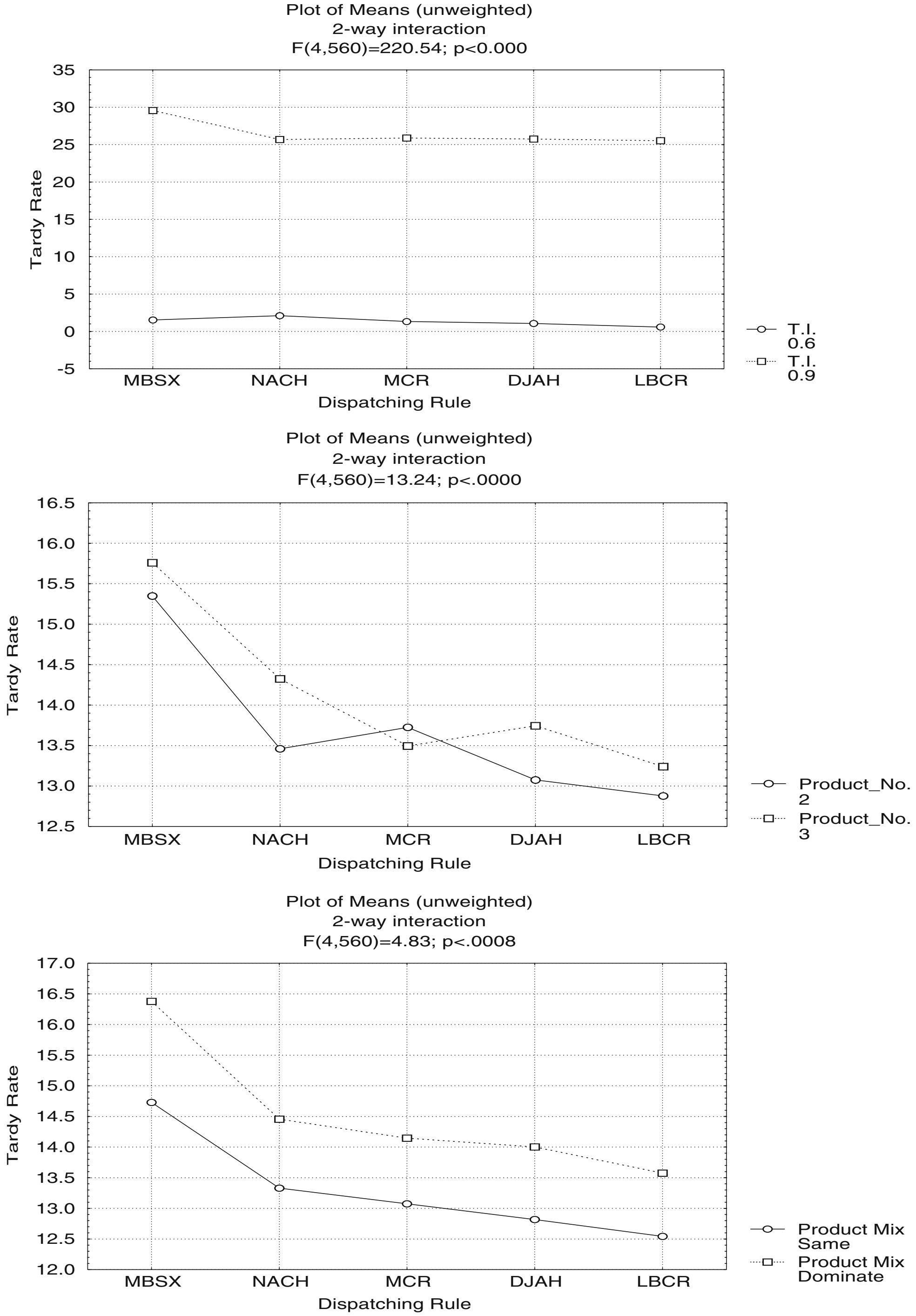
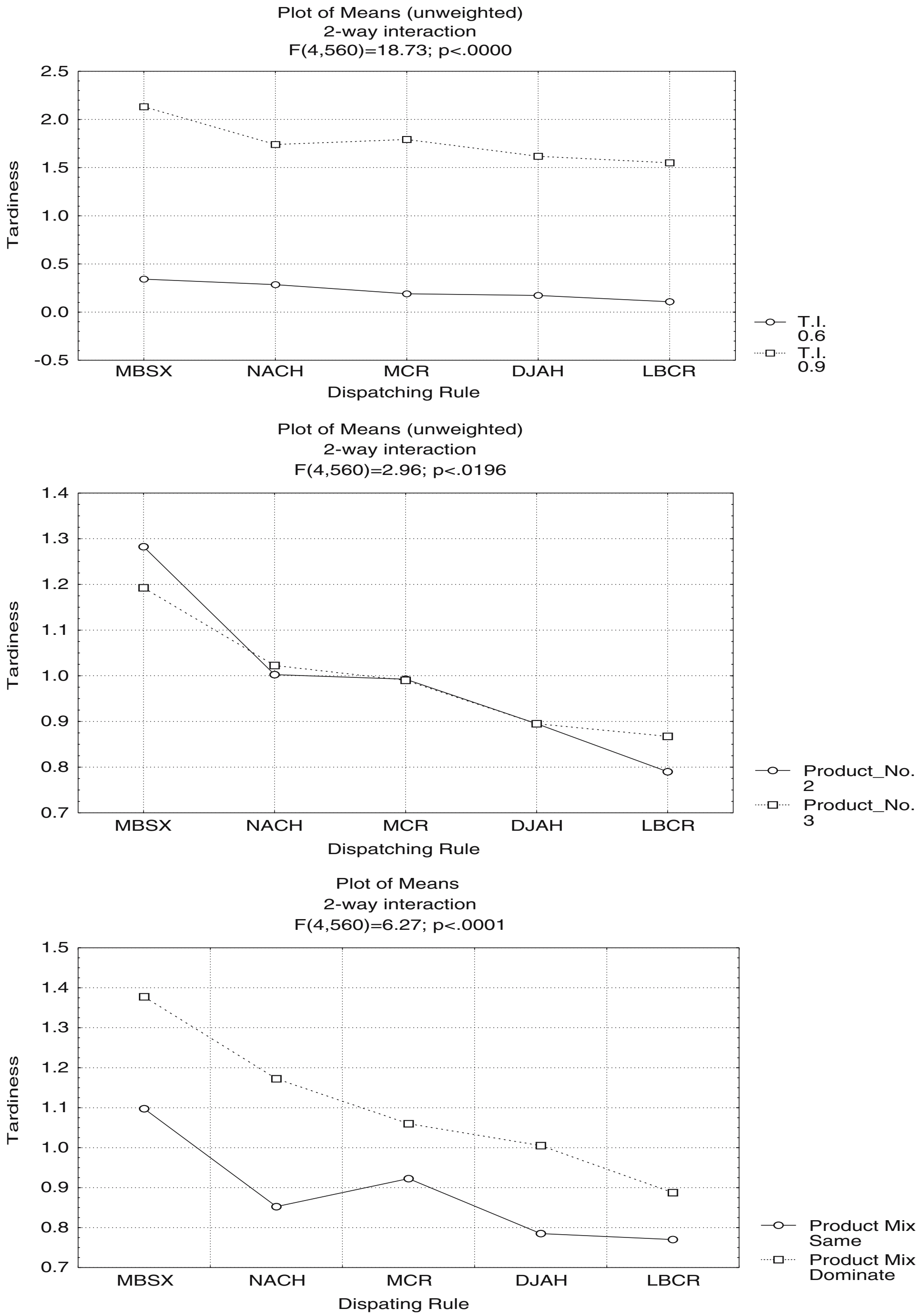


Fig. 3 Performance of batch dispatching rule: tardy rate

only adopt middle and high levels of traffic intensity to control the system utilization. We have two levels of product types, one has two product types, and the other has three product types. Product mix is represented by the ratio of different product types. For testing the effect of different rate of product mix, uniform rate and domination rate are adopted in the simulation experiment. Uniform rate means the product mix is 1:1 or 1:1:1. Domination rate will have one product type with a higher ratio than the others. The product mix in domination rate is 3:1 or 3:1:1.

In the simulation experiments, six replications (runs) of simulation in a steady state were executed for each combination of five batching rules (MBSX, NACH, MCR, DJAH, LBCR), two level of traffic intensity (0.6, 0.9), two level of product types (2,3), and two level of

Fig. 4 Performance of batch dispatching rule: tardiness



product mix (uniform, domination). Each simulation run was designed for a simulation time period of 24 hours a day and it would end when 3000 lots were finished after 150 warm days. Different random number seeds were used for the six runs, and each run was started with an empty fab. The values of required parameters for each strategy were determined by a series of preliminary simulation tests on several candidate values. Briefly, we have eight cases of production status (S1-S8, 2×2×2) based on three environmental factors (Table 3).

Performance indicators used in our simulation test are average flow time, throughput, tardy rate, and lateness. The

Table 6 The best dispatching rule under different production status

Production status	Tardy rate	Lateness	Avg. FT	Throughput
S1	LBCR	LBCR	MBSX	NACH
S2	LBCR	LBCR	LBCR	LBCR
S3	LBCR	LBCR	DJAH	LBCR
S4	LBCR	LBCR	NACH	MCR
S5	LBCR	LBCR	DJAH	DJAH
S6	LBCR	LBCR	NACH	DJAH
S7	LBCR	LBCR	DJAH	NACH
S8	LBCR	LBCR	DJAH	NACH

definition of those performance indicators are described as following:

– Tardy rate=number of tardy jobs/total of finished jobs,
– Tardiness=total of tardiness time/number of tardy jobs,
– Average flow time=total of production lead time/total of finished jobs,
– Throughput=total finished job in one years.

5 Statistics and analysis

In Table 3, we can find effects of the environmental factors' which are significant under statistical analysis. That is to say, there is significant interaction among three factors and five batch dispatching rules. We can find the performance of different batch dispatching rules in tardy rate and tardiness, look-ahead dispatching rules are better than other rules in two level of traffic intensity. In the four look-ahead dispatching rules, LBCR are outperforming the others, especially in tardy rate. The main reasons of the performance improvement in LBCR are the focus on the jobs' due date. LBCR will pay attention to the jobs with urgent due dates. Those jobs will be chosen to process by combining with some lots in the queue. Tardy rate under LBCR is smaller than the others.

By using Duncan's test we try to find the suitable batching rule under eight cases of production status and different performance indicators (Table 4). We can fine the LBCR with the significant performance improvement in tardy rate and tardiness and its system-related performance are not deteriorated. The production status with high traffic intensity (S5–S8), DJAH will have the short average flow time than LBCR. In low traffic intensity (S1–S4), LBCR will be better than the others in average flow time.

Those production statuses with few product types (S1, S2, S5, S6), will have more jobs in the queue waiting for processing. The extra flow time for having enough jobs wait for processing is less than those with three product types. The average flow time and tardiness will be shorter than others. Performance variance between different numbers of product type is significant, especially the dispatching rule, MBSX, is adopted. That's because MBSX can't adjust the minimum batching size under different system conditions.

Each production statuses will have its suitable batch dispatching rule. Under Duncan's test we will have a suggestion to the dispatching decision. In Table 5 we have a list about the best dispatching rule under different production status and performance indicators.

6 Conclusion and future work

In this paper, a new batch dispatching rule concerning the due-date, LBCR, is developed. Some popular rules were be tested at the same virtual wafer fabrication plant with different production status under three environmental factors, including traffic intensity, product types, and, product mix. After the simulation and statistical test, we have some conclusions as follows:

1. LBCR is better than others under due-date related performance indicators, due to the consideration of the methodology of LBCR for jobs' due-dates. Those jobs with urgent due-dates will have the high priority to process.
2. In average FT and throughput, LBCR is not better than the others. It's a trade-off between system-related performance indicators (throughput/flow time) and due-date related performance indicators (tardy rate/ tardiness). Nowadays, customers' satisfaction will be more important than system performance. LBCR can decrease the tardy rate and tardiness, and not cause a significant deterioration of system performance. It's a suitable batch dispatching rule for modern production system.
3. In different production status and performance indicators, suitable batch dispatching rules are suggested in the paper. LBCR can outperform the others in 19 conditions, especially when the due-date related performances are concerned. DJAH and NACH with six and five conditions respectively, are better than the others. LBCR is steadier than others as faced with the different production conditions.

In the future, there are some topics whcih can be discussed.

1. The development of integral strategy considering serial dispatching and batch dispatching rules in serial-batch production system (such as wafer fabrication) is an important task. The interaction between serial machine

and batch machine is significant [20]. Serial and batch dispatching decisions can't be considered separately.

2. LBCR can be extended to different production systems with batch machines, especially the food industry, Figs. 2, 3 and 4; Tables 4 and 6).

Acknowledgement This research acknowledges the subvention from National Science Council, Taiwan R.O.C., project No: NSC 93-2213-E-270-004.

References

1. Uzsoy R, Lee C-Y, Martin-Vega LA (1992) A review of production planning and scheduling models in semiconductor industry. PartI: system characteristics, performances evaluation and production planning. IIE Trans 24(4):47–58
2. Uzsoy R, Lee C-Y, Louis A (1994) A review of production planning and scheduling models in semiconductor industry. PartII: shop floor control. IIE Trans 26(5):44 –55
3. Van der Zee DJ, Van Harten A, Schuur PC (1997) Dynamic job assignment heuristic for multi-server batch operations - a cost based approach. Int J Prod Res 35(11):3063–3093
4. Neuts ME (1967) A general class of bulk queues with poisson input. Ann Math Stat 38:759–770
5. Deb RK, Serfozo RF (1973) Optimal control of batch service queues. Adv Appl Probab 5:340 –361
6. Weng WW, Leachman RC (1993) An improved methodology for real-time production decisions at batch-process workstation. IEEE Trans Semicon Manuf 6(3):219–225
7. Glassey CR, Weng WW (1991) Dynamic batching heuristic for simultaneous process. IEEE Trans Semicond Manuf 4(2):77-82
8. Fowler JW, Hogg GL, Philips DT (1992) Control of multiproduct bulk service diffusion / oxidation process. IIE Trans 24 (4):84–96
9. Fowler JW, Hogg GL, Phillips DT (2000) Control of multiproduct bulk server diffusion/oxidation processes. Part 2: multiple servers. IIE Trans 32(2):167–176
10. Van der Zee DJ, Van Harten A, Schuur PC (1996) Dynamic job assignment heuristics for multi-server batch operations. In: Franco JC, Rutten WGMM (eds) Proc Second International Conference on Compute Integrated Manufacturing in the Process Industries. Research Institute BETA, Eindhoven, The Netherlands, pp 558–573
11. Van der Zee DJ, Van Harten A, Schuur PC (1999) On line scheduling of multi-server batch operations. Technical Report 99A52. Research Institute SOM, University of Groningen, The Netherlands
12. Neale JJ, Duenyas I (2000) Control of manufacturing networks which contain a batch processing machine. IIE Trans 32 (11):1027–1041
13. Kim YD, Kim JG, Choi B, Kim HU (2001) Production scheduling in a semiconductor wafer fabrication facility producing multiple product types with distinct due dates. IEEE Trans Robot Autom 17(5):589–598
14. Van der Zee DJ, Van Harten A, Schuur P (2001) On-line scheduling of multi-server batch operations. IIE Trans 33 (7):569–586
15. Cigolini R, Perona M, Portioli A, Zambelli T (2002) A new dynamic loo-ahead scheduling procedure for batching machines. J Sched 5:185–204
16. Van der Zee DJ (2002) Adaptive scheduling of batch servers in flow shops. Int J Prod Res 40(12):2811–2833
17. Neale JJ, Duenyas I (2003) Control of batch processing machine serving compatible job families. IIE Trans 35:699–710
18. Robinson JK, Fowler JW, Bard JF (1995) The use of upstream and downstream information in scheduling semiconductor batch operation. Int J Prod Res 33(8):1849–1869
19. Hsu SY, Sha DY (2004) Due date assignment using artificial neural networks under different shop floor control strategies. Int J Prod Res 42(9):1727–1745
20. Sha DY, Hsu SY (2003) The effect of rework on the shop floor control in wafer fabrication. Integr Manuf Syst 15(2):184–198