ORIGINAL ARTICLE

Muh-Cherng Wu · Y.L. Huang · Y.C. Chang · K.F. Yang

Dispatching in semiconductor fabs with machine-dedication features

Abstract This research develops dispatching algorithms for a fab with machine-dedication characteristics. Machine-dedication, a new feature in a modern fab, has not been addressed in previous studies of dispatching. Three performance indices, including hit rate, mean cycle time, and throughput are of concern in dispatching. This research develops an algorithm, called LB-SA, based on a proposed simplification model of the process route. The line balance (LB) component aims to smooth the flow rate of the process route; and the starvation avoidance (SA) component aims to ensure that the bottleneck machine is not "starving" and has enough work-in-progress (WIP) to process all the time. Thirty dispatching algorithms, including the LB-SA algorithm, are compared by simulation. The LB-SA algorithm outperforms the other 29 algorithms both in terms of hit rate and mean cycle time, and is only slightly less than the best benchmark in throughput. Of the 29 other algorithms, one algorithm, called CR-SA, also performs very well. These two algorithms are both recommended for fabs with machine-dedication feature.

Keywords Dispatching · Machine-dedication · Semiconductor · Stepper

1 Introduction

Semiconductor manufacturing is a process to make integrated circuits (IC) on a wafer. Wafers are transported in a lot, which typically includes 25 wafers. The factory, called a fab, includes hundreds of machines, categorized into several groups of workstations. The process route of a product involves hundreds of operations, with a re-entry characteristic. That is, before its completion, a product may enter a workstation several times. A fab may produce several products simultaneously. Therefore, the types of work-in-process (WIP) waiting at a workstation are highly

varied. To effectively control a fab's operational performance, dispatching decisions are very important.

Dispatching decisions for semiconductor manufacturing have been extensively studied. Most studies developed dispatching algorithms for increasing throughput and reducing cycle time [1–5]. Some others [6–9] focused on reducing tardiness and hit rate, in addition to improving throughput and cycle time. These studies have established significant milestones in semiconductor dispatching. Yet, one characteristic of a modern fab, called machine-dedication, has been rarely concerned in previous studies on dispatching.

Machine-dedication is a feature associated with a stepper, a machine in a fab. A stepper mainly performs the exposure operation, which is to "photo-print" electronic circuits on a wafer. Stepper workstations, as well as exposure operations, may be categorized into two types, according to their specification in resolution: high-resolution and normal-resolution.

High-resolution operations, processed by high-resolution steppers, have a machine-dedication feature. That is, for a product, all its high-resolution operations need to be processed by a particular high-resolution stepper. Other steppers, even identical in specification, cannot process these operations. The underlying reason is to ensure manufacturing quality; this is because any two steppers in high-resolution scenarios are slightly different. Conversely, normal-resolution operations, processed by normal-resolution steppers, have a sharing feature. That is, for a product, any normal-resolution stepper can process any of its normal-resolution operations. To highlight the machine dedication feature, we refer to the high-resolution steppers as "dedicated" steppers and the normal-resolution steppers as "non-dedicated" steppers.

This research aims to develop dispatching algorithms for make-to-order fabs with machine-dedication characteristics. The performance indices involve hit rate (rate of on-time delivery), throughput, and cycle time. Of the three indices, hit rate is most important in a competitive make-to-order fab, where it is very important to fulfill due-date commitments to customers.

This paper is organized as follows. Section 2 analyzes the releasing and dispatching decisions in the fab of interest, and

M.-C. Wu (✉) · Y.L. Huang · Y.C. Chang · K.F. Yang
Department of Industrial Engineering and Management
National Chiao Tung University
Hsin-Chu, Taiwan, R.O.C.
E-mail: mcwu@cc.nctu.edu.tw

identifies the dispatching decisions focused in this research. Section 3 presents the proposed dispatching algorithms. Simulation results are showed in Sect. 4, and concluding remarks are discussed in the last section.

2 Dispatching decisions

This section aims to clarify the dispatching decisions considered in our research. We first analyze the relevant decisions in a fab. Of these decisions, algorithms for two dispatching decisions will be addressed. The other decisions, not a focus of this research, are solved by existing methods available in the literature.

2.1 Releasing decisions

Releasing activities in a particular fab involve two main decisions: when to release? And to which dedicated-stepper should a released lot be assigned? For a make-to-order fab, we assume that the releasing sequence of wafer lots has been predetermined.

For the first decision – when to release – there are two major approaches: open-loop control and closed loop control [10]. Open-loop control means releasing a lot based on a predetermined schedule. A typical example is uniform loading (UL) [11], where a fixed number of lots is released in each time period. Closed-loop control means releasing a lot based on the current WIP status of the fab. Typical examples of closed-loop control include starvation avoidance (SA) [12], workload regulation (WR) [13], and two-boundary control [14]. In this research, we adopt uniform loading for the first decision because this method has been widely used in make-to-order fabs.

For the second decision – dedicated-steppers assignment – this research adopts the accumulated load balance method, a widely used approach in make-to-order fabs. This method aims to ensure the accumulated working load of each dedicated-stepper is as balanced as possible. For example, suppose there are n dedicated-steppers and n wafer lots to be released. Then, each dedicated-stepper is assigned one lot, whether the stepper is functionally down or up. The accumulated load balance method is, in essence, an open-loop control paradigm.

2.2 Dispatching for batch workstation

The dispatching decision involves determining which lot to process first, given the WIPs waiting at a workstation. A workstation is a set of functionally identical machines with shared characteristics. That is, any machine at a workstation can process the operations assigned to the workstation. Therefore, a dedicated-stepper workstation involves only one machine, and other workstations may involve more than one.

Workstations can further be classified into two types: series and batch. A machine in a batch workstation processes several lots of wafers simultaneously; for example, a furnace machine can process up to six lots (150 wafers) at a time. A machine in a series workstation, on the other hand, processes one piece of wafer at a time. Steppers are a series-type machine.

A number of dispatching algorithms for batch workstations have been developed. This research adopts the minimum batch size (MBS) algorithm [15] for the dispatching of a batch workstation. The MBS algorithm denotes that each batch machine cannot process wafers unless the number of lots processed at one time exceeds a predefined threshold. Among the various WIPs that exceed the threshold value, we use the concept of first-in-first-out (FIFO) [11] to dispatch lots.

2.3 Research focus

This research focuses on developing dispatching algorithms for two types of workstations – dedicated-steppers and non-dedicated series workstations. Notice that a dedicated-stepper workstation involves only one stepper. We therefore use the terms dedicated-stepper and dedicated workstation interchangeably. Likewise, non-dedicated series workstations are hereafter called non-dedicated workstations.

3 Dispatching algorithms

The two proposed dispatching algorithms are developed based on a proposed simplification model of process routes. The process route of a product may involve hundreds of operations, and is decomposed into many segments. Each segment involves a sequence of operations, and the last operation is processed by a dedicated-stepper. Namely, in each segment, all the operations (except for the last) are for non-dedicated workstations, either batch or series.

3.1 Dispatching for dedicate-steppers

Suppose a fab that produces m products has k dedicated-steppers. The route of each product is identical, and is comprised of n segments. Then, for a dedicated-stepper, the number of WIP types waiting for dispatching is $m \times n$. The dispatching decision for a dedicated-stepper is to select one type from the $m \times n$ WIP types.

The basic idea of the proposed dispatching algorithm for a dedicated-stepper is line-balance (LB). The LB algorithm is based on a simplification model of the process route. That is, the non-dedicated operations of each segment are modeled by a black-box production line with a fixed cycle time. A fab, by such modeling, becomes a re-entry flow line (Fig. 1). The idea of the LB algorithm is to make the flow rate of each segment as equal as possible. Details of the dispatching algorithm for a dedicated-stepper are presented below.

Step 1: Compute the flow rate (v_i) of each segment entering the dedicated-stepper. WIP_i represents the current amount of WIP in segment i. CT_i denotes the cycle time of segment i, which has been estimated in advance by simulation.

$$v_i = \frac{WIP_i}{CT_i} \qquad 1 \leq i \leq n$$

Fig. 1. LB algorithm for dedicated-stepper M_1
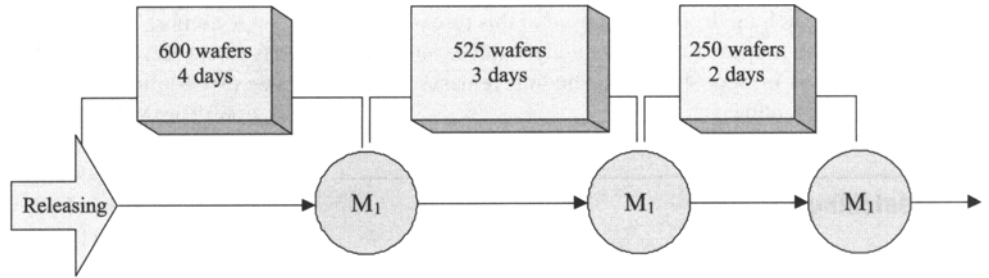
Fig. 2. An example of SA algorithm applied to a non-dedicated workstation
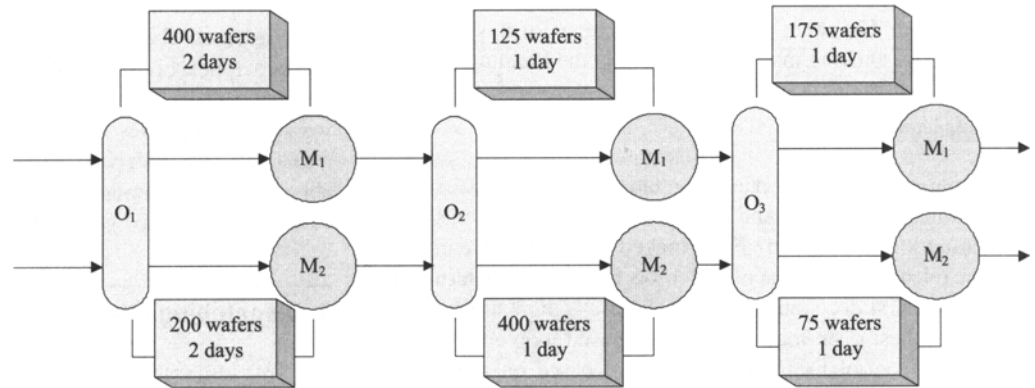
Step2: Identify the segment i^* which has the maximum flow rate.

$$i^* = ArgMax(v_i) \quad \text{for} \quad 1 \leq i \leq n$$

Step 3: Use FIFO the rule to prioritize the lots in segment i^*. Dispatch the lot with the highest priority.

The LB algorithm is further explained by the example shown in Fig. 1. The dedicated-stepper M_1 has three segments of WIPs to dispatch. The cycle time of each segment has been pre-estimated by simulation, where $CT_1 = 4$ days, $CT_2 = 3$ days, and $CT_3 = 2$ days. The values of CT_i are predetermined and kept unchanged in making all the dispatching decisions. The amount of WIP in each segment, which could change dynamically, are now $WIP_1 = 600$ wafers, $WIP_2 = 525$ wafers, and $WIP_3 = 250$ wafers. Thus, the flow rate in segment 2 is the highest, which can also be interpreted as the "most jammed segment" in a pipeline. Therefore, the highest priority of the dedicated-stepper is to process or "remove" the WIPs in segment 2 to make the segment less jammed. Dispatching in such a manner appears to make the production line more balanced.

3.2 Dispatching for non-dedicated-steppers

For a non-dedicated workstation, the number of WIP types waiting for dispatching is $m \times n \times k$, where m denotes the number product types, n denotes the number of segments, and k denotes the number of dedicated-steppers. The reason for including k is that any two WIP lots, even if they are identical in product type and segment, may have been assigned to different dedicated-steppers. This would imply different production lines

in the downstream. The dispatching decision for a non-dedicated workstation is therefore to select one from the $m \times n \times k$ WIP types.

The basic idea of the proposed algorithm for dispatching non-dedicated workstations is based on the concept starvation avoidance (SA) [12]. The SA algorithm is based on a simplified model of the process route. That is, in each segment, the non-dedicated workstation for dispatching is modeled as a "lot-supplying resource" from which lots are distributed to different dedicated-steppers through various pipelines (Fig. 2). Suppose there are k dedicated-steppers and n segments. Then, there are $n \times k$ pipelines to be supplied by the non-dedicated workstation. Each pipeline is assumed to have a fixed cycle time.

The dispatching decision is to determine which pipeline should be supplied first. The SA algorithm is intended to keep the downstream dedicated-steppers from being starved. The flow rate of a pipeline represents the "daily food" supplied to a dedicated-stepper. The pipeline that has the lowest flow rate tends "starve" its downstream stepper and should first be "supplied" by the non-dedicated workstation. Details of the SA algorithm are presented below.

Step 1: Compute the flow rate (v_{ij}) of each segment entering the dedicated-stepper. WIP_{ij} represents the amount of WIP, which currently stays in segment i and will leave for dedicated-stepper j; and CT_{ij} denotes the associated cycle time, which has been pre-estimated by simulation.

$$v_{ij} = \frac{WIP_{ij}}{CT_{ij}} \qquad 1 \leq i \leq n; \quad 1 \leq j \leq k$$

Step 2: Identify the pipeline (i^*, j^*) with the lowest flow rate.

$$(i^*, j^*) = ArgMin(v_{ij})$$

Step 3: Use FIFO to prioritize the lots that wait before the non-dedicated workstation and will leave for the pipeline (i^*, j^*). Dispatch the lot with the highest priority.

An example of the above SA dispatching algorithm is shown in Fig. 2. In the figure, O_i represents the operations processed by the non-dedicated workstations; and M_i denotes the two dedicated-steppers. There are six pipelines emanating from the non-dedicated-steppers. The cycle time of each pipeline is known, as indicated. The pipeline in segment 3, which leads to dedicated-stepper M_2, has the lowest flow rate. Therefore, the WIP lots leaving for the pipeline have the highest priority of dispatching.

Notice that in the LB algorithm, we are concerned with the flow rates entering the dedicated-stepper. In the SA algorithm, we are concern with the flow rates emanating from the non-dedicated workstation. The two algorithms, when integrated, are called the LBSA dispatching algorithm.

4 Simulation experiments

The proposed algorithm has been compared to some other dispatching algorithms by simulation.

Table 1. Thirty dispatching algorithms compared

Dedicated-steppers Non-dedicated WS.	FIFO	CR	LS	EDD	MSEC2	LB
FIFO	D01	D06	D11	D16	D21	D26
CR	D02	D07	D12	D17	D22	D27
LS	D03	D08	D13	D18	D23	D28
EDD	D04	D09	D14	D19	D24	D29
SA	D05	D10	D15	D20	D25	D30

Table 2. ANOVA analysis of experiments in terms of hit rate

Factor	SS	df	MS	F
Ded. WS. (A)	7.85	5	1.57	613.0
Non-ded. WS. (B)	11.72	4	2.93	1144.2
A × B	4.52	20	0.23	88.2
Error	1.46	570	0.00	
Total	25.54	599		

4.1 Fab data and setup for experiment

The fab of interest involves 60 workstations, of which 51 are series-type and nine are batch-type. A product family which involves five products is produced. Each product has an identical process route, but has different operation times. A standard product is taken as the benchmark. The operation times of the other

Table 3. Duncan test of experiments in terms of hit rate

Ded. WS.	Non-ded. WS.	Hit rate	1	2	3	4	5	6	7	8	9	10	11	12
FIFO	EDD	0.45%	****											
MSEC2	EDD	0.88%	****											
EDD	FIFO	1.06%	****											
EDD	EDD	1.23%	****											
LS	EDD	2.19%	****	****										
EDD	CR	2.33%	****	****										
LS	FIFO	2.86%	****	****										
LS	CR	5.17%		****	****									
FIFO	LS	5.43%		****	****									
CR	EDD	7.46%			****	****								
MSEC2	LS	7.72%			****	****								
EDD	LS	7.89%			****	****								
FIFO	CR	9.26%				****	****							
LS	LS	9.80%				****	****							
EDD	SA	10.23%				****	****							
MSEC2	CR	10.30%				****	****							
LB	EDD	12.75%					****	****						
CR	CR	14.85%						****						
CR	LS	15.10%						****						
LS	SA	20.19%							****					
LB	CR	21.39%							****	****				
FIFO	FIFO	21.86%							****	****				
MSEC2	FIFO	22.45%							****	****				
LB	LS	23.99%								****				
MSEC2	SA	41.43%									****			
CR	FIFO	50.61%										****		
FIFO	SA	52.72%										****		
LB	FIFO	56.64%											****	
CR	SA	66.42%												****
LB	SA	67.90%												****

Table 4. ANOVA analysis of experiments in terms of cycle time

Factor	SS	df	MS	F
Ded. WS. (A)	819.97	5	163.99	1353
Non-ded. WS. (B)	176.39	4	44.10	364
A × B	722.32	20	36.12	298
Error	69.11	570	0.12	
Total	1787.79	599		

four products are given by multiplying the standard operation time by a uniform distribution value (0.95, 1.05). The process route, the real data provided by industry, involves 344 steps, and passes the dedicated-steppers 11 times and non-dedicated-steppers 12 times.

Three performance indices, including hit rate, cycle time, and throughput, are measured. Of the three indices, hit rate, the percentage of on-time delivery lots, is most important. The due date of each lot is given by $d_i = r_i + x\, pt_i$, where d_i represents the due date of lot i, r_i denotes the release date, pt_i denotes the processing time, and $x = 1.54$ denotes a predetermined scale factor.

The product mix ratio is 1:1:1:1:1, and is released to the fab with a uniform loading policy. Each experiment is carried out for 20 runs, with different random seeds in each run. The time horizon for the simulation is 270 days. The first 90 days

are a "warm-up" period, and the data of the last 180 days are collected to represent the fab's performance. The simulation program is performed by a personal computer, equipped with an Intel Pentium IV 2.4 GHz CPU. Each simulation takes about 10 min.

4.2 Experiments and results

(A) Compared algorithms

Thirty dispatching algorithms, including the proposed one, are compared (Table 1). The 30 algorithms are developed by the combination of five dispatching algorithms for non-dedicated workstations and six dispatching algorithms for dedicated-steppers. The proposed algorithm is experiment D30. In Table 1, FIFO denotes first-in-first-out; CR denotes critical ratio; LS denotes least slack; EDD denotes earliest due date; and MESC2 denotes an algorithm proposed by Kim [7].

(B) Hit rate comparison

Table 2 shows the ANOVA analysis of the experiments in terms of hit rate. The table indicates that dispatching algorithms indeed have a significant impact on the hit rate. The Duncan test [16] for the 30 experiments in terms of hit rate is shown in Table 3. The table shows that the proposed LB-SA algorithm outperforms the other 29 algorithms.

Table 5. Duncan test of experiments in terms of cycle time

Ded. WS.	Non-ded. WS.	Mean CT	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
LB	SA	23.57	****														
FIFO	SA	23.98		****													
CR	FIFO	24.05		****													
LB	FIFO	24.11		****													
CR	CR	24.42			****												
LB	CR	24.45			****	****											
CR	SA	24.46			****												
LB	LS	24.68				****	****										
MSEC2	CR	24.83					****	****									
CR	LS	24.87					****	****									
MSEC2	FIFO	24.87					****	****									
FIFO	CR	24.87					****	****									
FIFO	FIFO	24.89					****	****									
LS	CR	25.03						****									
LB	EDD	25.07						****									
LS	LS	25.31							****								
MSEC2	SA	25.36							****	****							
LS	SA	25.41							****	****	****						
MSEC2	LS	25.46							****	****	****						
FIFO	LS	25.51							****	****	****						
CR	EDD	25.56								****	****						
EDD	LS	25.63								****							
LS	EDD	26.08										****					
EDD	CR	26.26										****	****				
FIFO	EDD	26.36											****	****			
EDD	EDD	26.50												****			
MSEC2	EDD	26.58												****			
LS	FIFO	28.38													****		
EDD	SA	28.98														****	
EDD	FIFO	32.24															****

Table 6. ANOVA analysis of experiments in terms of throughput

Factor	SS	df	MS	F
Ded. WS. (A)	$2.05 \times 10^{+06}$	5	$4.09 \times 10^{+05}$	1190
Non-ded. WS. (B)	$3.91 \times 10^{+05}$	4	$9.77 \times 10^{+04}$	284
A × B	$1.61 \times 10^{+06}$	20	$8.07 \times 10^{+04}$	235
Error	$1.96 \times 10^{+05}$	570	$3.44 \times 10^{+02}$	
Total	$4.25 \times 10^{+06}$	599		

(C) Cycle time comparison

Table 4 shows the ANOVA analysis of the experiments in terms of mean cycle time. The table indicates that dispatching algorithms indeed have a significant impact on mean cycle time. The Duncan test of the experiments in terms of mean cycle time is shown in Table 5. Here, the proposed algorithm LB-SA once again outperforms the other 29 dispatching algorithms.

(D) Throughput comparison

In terms of throughput, the ANOVA analysis of the experiments is shown in Table 6. It shows that dispatching has a significant on throughput. The Duncan test is shown in Table 7. Notice that the proposed LB-SA algorithm is not the best one in throughput, but still stays in the second cluster, only slightly below the best benchmark (CR-SA) by about 0.2%.

By reviewing the performance of the CR-SA algorithm in terms of hit rate and cycle time, we found that the CR-SA algorithm also performs well in hit rate (the second best), but is not as effective as the LB-SA algorithm in mean cycle time (a difference of 3.6%).

Considering the three performance indices altogether, the LB-SA algorithm and the CR-SA algorithm both perform very well. Notice that the LB-SA algorithm was intentionally developed based on a simplified model of the process route. The CR-SA, which we also developed, was created through a combination of different dispatching algorithms.

5 Conclusion

Dispatching decisions for a semiconductor are very important in the operational performance of a fab. Machine-dedication, an important feature in a modern fab, has not been concerned in previous studies on dispatching. This research develops dispatching algorithms for a make-to-order fab with machine-dedication characteristics. We addressed the performance indices of hit rate, mean cycle time, and throughput. Of the three indices, hit rate is most important.

The proposed algorithm LB-SA is developed based on a proposed, simplified model of the process route. Here, the line balance (LB) component for the dispatching of dedicated-steppers aims to smooth the flow rate of the process route; and the

Table 7. Duncan test of experiments in terms of throughput

Ded. WS.	Non-ded. WS.	throughput	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
EDD	FIFO	5366	****																	
EDD	SA	5455		****																
LS	FIFO	5529			****															
LS	SA	5636				****														
EDD	CR	5639				****	****													
MSEC2	EDD	5649					****	****												
EDD	EDD	5650					****	****												
EDD	LS	5650					****	****												
LS	EDD	5660						****	****											
CR	EDD	5665							****	****										
MSEC2	LS	5672								****	****									
LS	LS	5673								****	****									
FIFO	EDD	5674								****	****									
FIFO	LS	5679									****									
LB	EDD	5695										****	****							
CR	LS	5695										****								
MSEC2	SA	5697										****	****	****						
LB	LS	5700										****	****	****	****					
LS	CR	5704										****	****	****	****					
FIFO	FIFO	5708											****	****	****	****				
MSEC2	FIFO	5709											****	****	****					
FIFO	CR	5712												****	****					
MSEC2	CR	5713												****	****					
LB	FIFO	5721													****	****				
FIFO	SA	5727														****	****			
LB	SA	5732														****	****	****		
CR	FIFO	5738															****	****	****	
LB	CR	5738																****	****	****
CR	CR	5741																	****	****
CR	SA	5746																		****

starvation avoidance (SA) component for the dispatching of non-dedicated workstations aims to ensure that the dedicated-steppers are not "starving" and have enough WIP to process all the time.

Thirty dispatching algorithm, including the LB-SA algorithm, are compared by simulation. The thirty algorithms are created by a combination of five dispatching algorithms for non-dedicated workstations and six algorithms for dedicated-steppers. Simulation experiments show that the LB-SA algorithm outperforms the other 29 algorithm both in terms of hit rate and mean cycle time, and performs good in throughput, only slightly less than the highest benchmark.

Among the 29 other dispatching algorithms, the CR-SA algorithm also performs very good in terms of both hit rate and throughput, and is acceptably good in terms of mean cycle time. Therefore, we would recommend the LB-SA and CR-SA algorithms for semiconductor fabs with machine-dedication characteristics.

References

1. Yoon HJ, Lee DY (2000) A control method to reduce the standard deviation of flow time in wafer fabrication. IEEE Trans Semicond Manuf 13(3):389–392
2. Lee YH, Park J, and Kim S (2002) Experimental study on input and bottleneck scheduling for a semiconductor fabrication line. IIE Trans Semicond Manuf 34(2):179–190
3. Li S, Tang T, and Colons DW (1996) Minimum inventory variability schedule with applications in semiconductor fabrication. IEEE Trans Semicond Manuf 9(1):145–149
4. Lu SCH, Ramaswamy D, and Kumar PR (1994) Efficient scheduling policies to reduce mean and variance of cycle-time in semiconductor manufacturing plant. IEEE Trans Semicond Manuf 7(3): 374–388
5. Kim YD, Lee DH, Kim JU, and Roh HK (1998) A simulation study on lot release control, mask scheduling, and batch scheduling in semiconductor wafer fabrication facilities. J Manuf Syst 17(2):107–117
6. Dabbas RM, Fowler JW (2003) A new scheduling approach using combined dispatching criteria in wafer fabs. IEEE Trans Semicond Manuf 16(3):501–510
7. Kim YD, Kim JG., Choi B, and Kim HU (2001) Production scheduling in a semiconductor wafer fabrication facility producing multiple product types with distinct due dates. IEEE Trans Robot Automat 17(5):589–598
8. Kim YD, Kim JU, Lim SK, and Jun HB (1998) Due-date based scheduling and control policies in a multiproduct semiconductor wafer fabrication facility. IEEE Trans Semicond Manuf 11(1):155–164
9. Lu SCH, Kumar PR (1991) Distributed scheduling based on due dates and buffer priorities. IEEE Trans Semicond Manuf 36(12):1406–1416
10. Glassey CR, Resende MGC (1988) Closed-loop job shop release control for VLSI circuit manufacturing. IEEE Trans Semicond Manuf 1(1):26–46
11. Morton TE, Pentico DW (1993) Heuristic scheduling systems. Wiley, New York
12. Glassey CR, Resende MGC (1988) A scheduling rule for job release in semiconductor fabrication. Oper Res Lett 7(5):213–217
13. Wein LM (1988) Scheduling semiconductor wafer fabrication. IEEE Trans Semicond Manuf 1:115–130
14. Lou SXC, Kager PW (1989) A robust production control policy for VLSI wafer fabrication. IEEE Trans Semicond Manuf 2(4):159–164
15. Neuts MF (1967) A general class of bulk service queue with Poisson input. Ann Math Stat 38:759–770
16. Montgomery DC (1991) Design and analysis of experiments. Wiley, New York