Journal of the Korean Institute of Industrial Engineers
Vol. 35, No. 4, pp. 266-279, December 2009.

A Scheduling Algorithm for Workstations with Limited Waiting Time Constraints in a Semiconductor Wafer Fabrication Facility

Byung-Jun Joo[1] · Yeong-Dae Kim[2†] · June-Young Bang[2]

[1]Productivity Research Institute, LG Electronics
[2]Department of Industrial and Systems Engineering, KAIST

대기시간 제약을 고려한 반도체 웨이퍼 생산공정의 스케쥴링 알고리듬

주병준[1] · 김영대[2] · 방준영[2]

[1]LG전자 생산성연구원 / [2]한국과학기술원 산업및시스템공학과

This paper focuses on the problem of scheduling wafer lots with limited waiting times between pairs of consecutive operations in a semiconductor wafer fabrication facility. For the problem of minimizing total tardiness of orders, we develop a priority rule based scheduling method in which a scheduling decision for an operation is made based on the states of workstations for the operation and its successor or predecessor operation. To evaluate performance of the suggested scheduling method, we perform simulation experiments using real factory data as well as randomly generated data sets. Results of the simulation experiments show that the suggested method performs better than a method suggested in other research and the one that has been used in practice.

Keywords: Scheduling, Semiconductor Wafer Fabrication, Waiting Time Limit, Tardiness

1. Introduction

We consider a scheduling problem in a semiconductor manufacturing system producing multiple product types according to customer orders. To survive in today's competitive business environments, manufacturers including those of semiconductor products should satisfy requirements of the customers in terms of quantity, quality and delivery. Especially in make to order systems, such as those producing system large scale integrated-circuit (LSI) products or application specific integrated-circuit (ASIC) products, meeting due dates is very important in maintaining or increasing the competitiveness of the firm.

In general, the semiconductor manufacturing process is composed of four major stages, wafer fabrication, electronic die sorting (EDS), assembly, and final test. Among the four, wafer fabrication is the most complex and time-consuming, since it involves a complex sequence of processing steps with a large number of operations. In this paper, we focus on a scheduling problem in a semiconductor wafer fabrication facility (fab) that produces multiple types of products to order, and develop an effective and efficient scheduling method to minimize total tardiness of orders.

An order for wafers is specified by the product (wafer) type, due date, and number of wafers to be produced. In wafer fabs, wafers are usually processed in a lot of 25 wafers or less. A wafer lot is the basic processing and transfer unit, i.e., a set of wafers that are processed and

† Corresponding author : 305-701 Dept. of Industrial & Systems Engineering, KAIST, Guseong-dong 373-1, Yuseong-gu, Daejeon, Korea,
Fax: 82-42-350-3110, E-mail: ydkim@kaist.ac.kr

Received August 12, 2009; Revision Received October 21, 2009; Accepted November 11, 2009.

moved together. One or more wafer lots need to be processed for an order. An order is considered to be completed in the wafer fab, only if all wafer lots for the order have been completed. The due date of an order for wafers in the fab can be set by considering the due date of the order for the final products (semiconductor chips) associated with the wafers and the production lead time required for the other three stages, i.e., EDS, assembly, and final test.

There are two types of workstations in the fab, serial-processing workstations and batch-processing workstations. In the former, wafer lots are processed individually, while multiple wafer lots can be processed simultaneously in batches in the latter type. However, for the wafer lots to be processed together on a batch-processing machine, the product types of the lots and production specifications on the machine should be compatible. Such a group of lots with compatible specifications is called a *wafer group* in this study. Therefore, wafer lots in a wafer group can be processed together on a batch-processing machine. In general, wafers of the same product type may have different processing specifications even on the same workstation since wafer lots visit the same workstation multiple times and hence there may be wafer lots (of the same wafer type) at different degrees of completion at the same workstation.

In the fab considered in this research, there is a time limit between the completion time of an operation for a wafer lot at a workstation and the start time of the subsequent operation for the wafer lot at the subsequent workstation for several pairs of workstations, such as pairs of workstations for deposition, diffusion, dry etching, metal, and wet etching. In other words, a wafer lot must be started at the downstream workstation within a certain time period after it is completed at the upstream workstation. If a lot cannot be started within the time period at the downstream workstation, it must be scrapped or processed again at the upstream workstation because of chemical characteristics of the processes.

For example, after a cleaning operation for a wafer lot is completed on the wet chemical etching workstation, the next operation should be started within a certain time period on the diffusion workstation so that cleaned surfaces of the wafers do not get contaminated or natural oxidation does not occur. Such a time period is called the *waiting time limit* and such a constraint is called a *limited waiting time constraint*, in this paper. The lengths of these time periods may differ for different wafer lots according to their chemical characteristics or processing specifications. The managers of the system considered in this study think that it is very important but difficult to deal with these constraints effectively and that it is necessary to develop an effective method for scheduling wafer lots in the fab that considers the waiting time constraints explicitly.

In general, the fab scheduling problem is a dynamic version of the static job shop scheduling problem, which is proven to be NP-hard (Garey and Johnson, 1975). Furthermore, the fab scheduling problem with the limited waiting time constraints may be considered a generalized version of the one without the constraints, and the former may be considered to be more difficult than the latter. In addition, if the lots that violate the limited waiting time constraints between two subsequent operations, these operations for the lots should be re-processed or new lots should be released into the fab, which makes the control of the fab more complicated and the scheduling problem more difficult.

There have been a number of research papers on scheduling problems in semiconductor wafer fabs, including those on lot release control problems and lot/batch scheduling problems as listed in <Table 1>. A large portion of these studies employ priority rule-based scheduling approaches. Among these, Kim *et al.* (1998a, 2001) present dispatching rules that are designed for fabs with low-volume and high-variety production settings for the objective of minimizing total tardiness of orders. There also have been studies on lot scheduling problems in batch processing machines/workstations, such as Glassey and Weng (1991), Gurnani *et al.* (1992), Weng and Leachman (1993), Chun and Hong (1996), and Fowler *et al.* (2000).

In addition, researchers give research results on scheduling problems in wafer fabs or other related problems. Kim *et al.* (2003), Min and Yih (2003), Upasani *et al.* (2006), and Sourirajan and Uzsoy (2007) propose various methods for fab scheduling problems, and Kim *et al.* (2008) suggest order-lot matching algorithms to minimize total tardiness of orders in a fab. Meanwhile, Lee *et al.* (2003) and Yildirim *et al.* (2006) suggest dispatching rules for scheduling problems in printed circuit board manufacturing systems, which have similar system characteristics to those of semiconductor manufacturing systems.

Scheduling problems with the limited waiting time constraints have been studied by several researchers including Hodson *et al.* (1985), Johri (1993), Wikum *et al.* (1994), Yang and Chern (1995), Robinson and Giglio (1999), Scholl and Domaschke (2000), Su (2003), Chen and Yang (2006), Fondrevelle *et al.* (2006), Artigues *et al.*

Table 1. List of research papers on scheduling problems in semiconductor wafer fabs

Measures	Decisions		
	Lot release control	Lot scheduling	
		Based on priority (dispatching) rule	Based on optimization modeling
Cycle time	Wein (1988) Lu *et al*. (1994) Lee *et al*. (2002)	Wein (1988) Kumar (1994) Lu *et al*. (1994) Li *et al*. (1996) Scholl and Domaschke (2000)* Lee *et al*. (2002)	
Workload/WIP balance	Kim *et al*. (1998b)	Kim *et al*. (1998b) Duwayri *et al*. (2006)	Kim *et al*. (2002)
Cost	Liao *et al*. (1996) Sloan and Shanthikumar (2002)	Sloan and Shanthikumar (2002)	Liao *et al*. (1996)
Setup	Chern and Huang (2004)		
Tardiness	Kim *et al*. (1998a) Kim *et al*. (2001)	Kim *et al*. (1998a) Kim *et al*. (2001)	Hung (1998)
Multi-criterion	Hsieh *et al*. (2001) Lee *et al*. (2002)	Hsieh *et al*. (2001) Dabbas and Fowler (2003)	

* Research paper in which the limited waiting time constraints are considered

(2006), Sheen and Liao (2007), Caumond and Lacomme (2008), and Joo and Kim (2009). Among these, Scholl and Domaschke (2000) suggest a scheduling algorithm for operations at the wet etching and diffusion workstations in semiconductor wafer fabs for the objective of minimizing mean cycle time. However, to the best of our knowledge, there has been no research on scheduling problems in wafer fabs with the limited waiting time constraints for the objective of minimizing total tardiness of orders.

In this paper, we suggest a scheduling method for a fab with limited waiting time constraint for the objective of minimizing total tardiness of orders. We assume that information such as the arrival times of orders and processing times of the lots are given and deterministic, and we do not consider unexpected events such as arrivals of urgent orders and machine breakdowns in this study. We suggest a new scheduling policy to deal with the constraint effectively, and develop scheduling methods based on the new scheduling policy and an existing policy. We evaluate the performance of the scheduling methods through a simulation study. Throughout the paper, a lot or a batch of lots and an operation for a lot or a batch are said to be (*waiting*) *time-constrained* if the lot or the batch is located at one of a pair of workstations at which there is a limited waiting time constraint on the operations for the lot or batch.

2. Scheduling Policies

In this study, we consider two types of scheduling policies for time-constrained wafer lots, a *successor-focused scheduling policy* (SFSP) and an *interdependent scheduling policy* (ISP). The SFSP is suggested by Scholl and Domaschke (2000) and may be considered a *pull policy*. Under the SFSP, scheduling decisions in a pair of workstations with a waiting time constraint are made based on, or with emphasis on, the downstream workstation of the pair, that is, a schedule at the downstream workstation is obtained first, and then scheduling decisions at the predecessor (upstream workstation) are made according to the schedule at the downstream workstation. At the upstream workstation, a time-constrained lot is scheduled in a way that it can be started at the downstream workstation at its scheduled time. Therefore, to make a scheduling decision at the downstream workstation, one needs to consider not only lots in the queue for the downstream workstation but also in the queue for the upstream workstation.

Under the SFSP, time-constrained lots are *pre-scheduled* on the machines at the downstream workstation such that they can be started at the pre-scheduled time points when they are completed at the upstream workstation. Here, a wafer lot and a machine are said to be pre-scheduled, if the lot is scheduled to be processed on the machine at a specific point of time in the future. Hence, a machine at the downstream workstation that is pre-scheduled for a time-constrained lot may have to idle and wait until the lot arrives at the downstream workstation after it is completed at the upstream workstation although there are other lots that are available for the machine. This may result in unnecessary under-utilization of the downstream workstation. In addition, if a time-constrained lot is pre-scheduled on a machine at the downstream workstation, it is to be scheduled on the machine as soon as it arrives at the downstream workstation, although there are more urgent lots in the queue at the downstream workstation. See Scholl and Domaschke (2000) for more details of the SFSP.

Under the interdependent scheduling policy (ISP) developed in this research, scheduling decisions at each workstation are made based on the states of both workstations. Under the ISP, time-constrained lots are started at the upstream workstation in a way that prevents the waiting time constraints from being violated (too often) by using information on the states of both workstations. At the downstream workstation, non-time-constrained lots are started prior to time-constrained lots if it is found that the non-time-constrained lots are more urgent than the time-constrained lots. Therefore, unlike under the SFSP, neither of the two workstations has dominating influence on the schedules under the ISP, and schedules are generated based on up-to-date information of time-constrained and non-time-constrained lots.

Under both scheduling policies, lots are scheduled using the list scheduling method in this study. In the list scheduling method, when a machine becomes available, a lot with the highest priority, i.e., a lot with the smallest priority value, is selected from those that can be processed on the machine, and the selected lot is scheduled on the machine. List scheduling methods are frequently used in many real manufacturing systems, since it can be easily implemented and used in systems with complex material flow and under dynamic environment. Note that batching decisions should also be made at batch-processing workstations. In this study, priorities of the lots are computed by using ES/RW2, a priority rule suggested in Kim *et al.* (2001), under both policies.

In ES/RW2, priorities of lots are determined based on the estimated slack time per remaining work. This rule is used in this study since it is reported that this rule works better than others for the objective of minimizing tardiness of orders (Kim *et al.* 2001). In ES/RW2, the priority of lot i at workstation k, π_{ik}, is computed as

$$\pi_{ik} = \max\Big[\max\Big\{d_i - R_{l_i} - W^r_{l_i} - t + \omega \cdot \frac{h_i}{P_i(n^r_i + 1)},\ 0\Big\}/(R_i + W^r_i),\ \delta \cdot P_{ik}\Big],$$

where d_i is the due date of lot i (due date of the order associated with lot i),

l_i is the index of the lot with the least progress among lots in the order corresponding to lot i,

R_i is the remaining work of lot i, which can be computed as $R_i = \sum_{u=k}^{K} P_{iu}$, where K is the index of the last operation of lot i,

W^r_i is the total waiting time of lot i until it is completed in the fab,

t is the time when the scheduling decision is to be made,

h_i is the difference in remaining works of lot i and l_i,

n^r_i is the number of lots with less progress than lot i among lots in the order associated with lot i,

p_{ik} is the processing time of lot i at workstation k,

P_i is the total processing time for all operations of lot i, and

ω, δ are scheduling parameters that have to be predetermined.

Here, W^r_i is computed using average waiting time per layer, which can be estimated through a series of preliminary simulation runs. ES/RW2 may be considered as an improved version of the slack per remaining work rule proposed by Anderson and Nyirenda (1990). See Kim *et al.* (2001) for more details of ES/RW2.

In the fab considered in this study, there are limited waiting time constraints between seven pairs of workstations as given in <Table 2>, and there are no such constraints between any other pair of workstations. Note that the same type of operations may have to be performed consecutively in some cases, and in this case, lots have to visit the same workstation consecutively. The workstation pairs (for upstream and downstream operations) can be a serial-processing workstation and a serial-processing workstation, a serial-processing workstation and a batch-processing workstation, or a batch-processing workstation and a batch-processing workstation. Note that there is no case in which the upstream workstation is a batch-processing workstation and the down-

Table 2. Workstation pairs with limited waiting time constraints

Upstream	Downstream	Type
Dry etching	Dry etching	serial-serial
Dry etching	Wet etching	serial-serial
Wet etching	Dry etching	serial-serial
Wet etching	Metal	serial-serial
Wet etching	Deposition	serial-batch
Wet etching	Diffusion	serial-batch
Diffusion	Diffusion	batch-batch

stream workstation is a serial-processing workstation. There may also be multiple upstream workstations for a downstream workstation, and multiple downstream workstations for an upstream workstation, because processing routes for different wafer lots (and for different layers of the same wafer lot) may differ in the fab considered in this study. Note that there are no limited waiting time constraints among three or more (consecutive) operations in the fab considered in this research.

3. Scheduling Algorithms for the Interdependent Scheduling Policy

The interdependent scheduling policy (ISP) is developed for the purpose of scheduling not only time-constrained lots but also non-time-constrained lots, unlike other dispatching or priority rule-based scheduling policies including Kim *et al.* (1998a) and Kim *et al.* (2001), in which limited waiting time constraints are not considered. Although Scholl and Dosmaschke (2000) consider the limited waiting time constraints in fabs, they only consider a case in which time-constrained lots can be processed at the downstream workstations immediately after they are completed at the upstream workstations. In other words, the waiting time constraints of the lots are not considered in the priority rule at the upstream workstations and the time-constrained lots always have higher priority values than the non-time-constrained lots at the downstream workstations.

In the interdependent scheduling policy (ISP), scheduling decisions are made based on the current states as well as look-ahead information of both workstations related to the limited waiting time constraint. In the following, we describe scheduling algorithms used under the interdependent scheduling policy for the workstation pairs with the limited waiting time constraints.

Before we present the scheduling algorithms, we give the notation used for description of the algorithms.

i	index of lots
f	index of wafer groups
k	index of workstations
M_k	number of machines at workstation k
B_k	batch capacity (number of lots that can be processed together) of the machines at workstation k (It is assumed in this study that all machines at the same workstation have the same capacity.)
p_{ik}	processing time of lot i at workstation k
z_{ik}	waiting time limit of lot i between workstation k and its successor workstation
t	time point when the scheduling decision is to be made
π_{ik}	priority of lot i at workstation k, which is computed by using the priority function of a priority rule at time t
L_k	set of all lots that are waiting at workstation k at time t
L_k^C	set of time-constrained lots that are waiting at workstation k at time t

3.1 Algorithms for the pairs of a serial workstation and a serial workstation

Although this type of workstation pairs is composed of two serial-processing workstations, scheduling procedures for the two workstations are different. Each of the scheduling procedures is described in detail in the following.

(1) Scheduling procedure for the upstream serial-processing workstation

When a machine at the upstream serial-processing workstation (USW) becomes available, priorities, π_{ik}'s, of the lots that are waiting in the queue are computed using ES/RW2, and a wafer lot with the highest priority is selected. If the selected lot is not time-constrained or if it can be started at the downstream serial-processing workstation (DSW) within its waiting time limit after it is completed at the USW, the selected lot is scheduled at the machine immediately. On the other hand, if the selected lot is time-constrained and it cannot be started within the waiting time limit, a lot with the next highest priority is selected and whether this lot can be started is checked again. If all lots in the queue are time-constrained and cannot be started within their waiting time limits, no lot is selected and the machine is kept idle until the next scheduling decision point, i.e., the time when another machine becomes available.

Since (future) start times of lots at the DSW cannot be known exactly at the time when the scheduling decision is made at the USW, we check whether the selected lot can be started at the DSW within its waiting time limit using the following condition,

$$t + p_{ik} + z_{ik} \geq t + \alpha\left(\sum_{i \in L_{k+1}^{C} \cup A_{k+1}^{C}} P_{i,k+1}/M_{k+1}B_{k+1}\right), \tag{1}$$

where k and k+1 are the indices of the USW and DSW, respectively, A_k^C is the set of time-constrained lots that will arrive at workstation k in time interval [t, $t+p_{ik}$], and α is a scheduling parameter which is used to estimate the start times of time-constrained lots at the DSW. Here, B_{k+1} is set to 1 since workstation k+1 is a serial-processing workstation, but the value for α should be selected through a test on a few candidate values. Note that the second term of the right-hand-side of (1) can be regarded as an estimated workload of the DSW due to the time-constrained lots. If the value for α is set to a larger value, it is less likely that the waiting time constraint of the selected time-constrained lot will be violated even though the selected lot is directly processed at the upstream workstation.

Since there is a time delay between the time when the scheduling decision for the USW is made and the time when the lot arrives at the DSW, for an accurate estimation of the workload of the DSW as well as for the computation of the right-hand-side of (1), it is necessary to identify A_k^C as accurately as possible. Here, for the identification of A_k^C for each USW, a schedule during the time gap is obtained using the list scheduling method with ES/RW2 as the priority rule. Lots that become available at workstation k within the time gap in the schedule are included in A_k^C. Note that there may be multiple USWs for a DSW because processing routes for different wafer lots may differ in the fab considered in this study.

The procedure of selecting and scheduling wafer lots at USWs can be summarized as follows. Here, N denotes the number of wafer lots waiting in the queue of the workstation being considered currently.

Procedure 1. (*Lot selection/scheduling procedure for USW*)

Step 1. When a machine becomes available at USW, compute priorities of the lots that are waiting in the queue for USW using a priority rule (ES/RW2 is used here).

Step 2. Let $n \leftarrow 1$.

Step 3. If the lot with the n-th highest priority is not time-constrained, schedule the lot, and exit from the procedure. Otherwise, go to step 4.

Step 4. If the lot with the n-th highest priority satisfies condition (1), schedule the lot, and exit. Otherwise, go to step 5.

Step 5. If $n < N$, let $n \leftarrow n+1$, and go back to Step 3. Otherwise, exit.

(2) Scheduling procedure for a downstream serial-processing workstation

If the above procedure is used for USWs, there may be cases in which urgent time-constrained lots cannot be started at a USW. Such cases occur if there are too many time-constrained lots at the DSW, since condition (1) cannot be satisfied. To prevent urgent time-constrained lots from waiting too long at the USW, time-constrained lots waiting at the DSW should be processed as early as possible. However, if the time-constrained lots are less urgent than other lots in the DSW, tardiness of these other lots may be increased by processing the time-constrained lots first at the DSW. Therefore, schedules at the DSW should be constructed by considering the effects of resulting schedules on time-constrained lots at the USW and DSW and non-time-constrained lots at the DSW.

We develop an urgency index for the DSW to consider such effects. The urgency index is computed using three types of information related to the states of the USW and DSW, which are two ratios of the average priority values of time-constrained lots to that of all lots, and a workload ratio of the workload for time-constrained lots to that for all lots at the DSW at the time when the scheduling decision is to be made. The urgency index for workstation k is given as

$$U_k = \{\beta \cdot e_{k-1} + (1-\beta) \cdot e_k\} \sum_{i \in L_k - L_k^C} p_{ik} / \sum_{i \in L_k} p_{ik}, \tag{2}$$

where $e_k \equiv \min\{(\overline{\pi}^C - \pi_{\min}) / (\overline{\pi} - \pi_{\min}), 1\}$ is the ratio of priority values at workstation k. Here, $\pi_{\min}$ is the minimum priority value among all lots, and $\overline{\pi}$ and $\overline{\pi}^C$ are the average values of the priorities of all lots and of time-constrained lots that are waiting at workstation k, respectively. The priority value of a lot is computed by the priority function of ES/RW2 given in Section 2. If there are multiple upstream workstations for workstation k, e_{k-1} is computed as if all lots waiting at the upstream workstations are waiting at a single (artificial) upstream workstation, i.e., workstation $k-1$.

If the urgency index of the DSW is smaller than a predetermined level (γ), priorities of time-constrained lots waiting at the DSW are increased with a multiplier,

$$\frac{r_{\max} - r_{\min}}{G \bullet (r_{\max} - r_i/2)}, \tag{3}$$

where r_i is the remaining waiting time limit of lot i (after it is completed at the USW), and $r_{\max}$ and $r_{\min}$ are the longest and shortest remaining waiting time limits of all time-constrained lots waiting at workstation k, respectively, and G is a parameter with a large value that should be pre-determined. (In this study, the value of G was set to 10000 after a test on several candidate values that range from 10 to 100000.) Then, the lot with the highest priority is selected and scheduled on the available machine at the DSW. By increasing priorities of the time-constrained lots with this multiplier, we let these lots have higher priorities, i.e., smaller priority values, so that those lots have to be processed prior to the non-time-constrained lots. Also, time-constrained lots with shorter remaining waiting time limits will be processed earlier than those with longer remaining waiting time limits. The urgency index ranges from 0 to 1, and as the urgency index becomes closer to 0, it is more likely that the time-constrained lots waiting at the DSW are processed earlier than others.

The following summarizes the procedure for selecting/scheduling wafer lots at the DSW.

Procedure 2. (*Lot selection/scheduling procedure for DSW*)

Step 1. When a machine becomes available at DSW, say workstation k, compute priorities, π_{ik}, of lots that are waiting in the queue of the workstation using a priority rule.

Step 2. If $U_k < \gamma$, go to Step 3. Otherwise, go to Step 4.

Step 3. For all time-constrained lots i at workstation k, let $\pi_{ik} \leftarrow \pi_{ik} \frac{r_{\max} - r_{\min}}{G \bullet (r_{\max} - r_i/2)}$.

Step 4. Select a lot with the highest priority and schedule it on the machine at the workstation. Exit from the procedure.

3.2. Algorithms for the pairs of a serial workstation and a batch workstation

For serial-batch pairs, the procedure of selecting/scheduling wafer lots at the upstream serial-processing workstation (USW) is the same as that for serial-serial pairs. In the procedure of downstream batch-processing workstation (DBW), when a machine becomes available, the urgency index for the DBW is computed with equation (2), which is used in the scheduling procedure for serial-serial pairs. If the index is smaller than the predetermined level, the priorities of the time-constrained lots, which are computed with ES/RW2, are increased with a multiplier given as

$$\frac{r_i - r_{\min}}{r_{\max} - r_{\min}} \bullet \max\left[\left(\frac{B_k - |L_k^C|}{B_k}\right)^2, 0\right]. \tag{4}$$

Then, the average priority values of the groups are computed using the priority values of lots waiting at the DBW, and a wafer group with the smallest average priority value is selected for batching.

In batch-processing workstations, a batching rule is needed to make batching and scheduling decisions. In this research, we use a heuristic method suggested in Kim *et al.* (2001), called the modified dynamic batching heuristic (MDBH), for batching at the batch-processing machines. In this method, if the number of available lots of the selected group is greater than or equal to the batch capacity, a full batch is formed with the lots with smaller priority values and scheduled on the available machine immediately. Otherwise, we compare two alternatives for batching. One alternative is to form a batch with lots that are available at the current time, and the other is to form a batch with the currently available lots and another lot that is expected to arrive first (after the current time). The alternative with a less total weighted waiting time is selected. See Kim *et al.* (2001) for more details of MDBH.

If in a batch related to the first alternative there is any time-constrained lot for which the waiting time constraint will be violated if processing of the batch is delayed until the arrival of a new lot, this first alternative is selected and the batch is started immediately. Otherwise, the second alternative is selected, and in this case, no scheduling decision is made at the current time point, that is, the available machine is kept idle until a new lot arrives at the DBW.

The procedure for batching and scheduling wafer lots at the DBW is summarized below.

Procedure 3. (*Batch scheduling procedure for DBW*)

Step 1. When a machine becomes available at DBW, say workstation *k*, compute priorities, π_{ik}'s, of wafer lots that are waiting in the queue of the workstation using a priority rule (ES/RW2 is used here). If $U_k < \gamma$, go to Step 2. Otherwise, go to Step 3.

Step 2. For all time-constrained lots, *i*'s, let $\pi_{ik} \leftarrow \pi_{ik}$

$$\frac{r_i - r_{\min}}{r_{\max} - r_{\min}} \bullet \max\left[\left(\frac{B_k - \left|L_k^C\right|}{B_k}\right)^2,\ 0\right].$$

Step 3. Select a wafer group with the smallest average priority value. If the number of available lots of the selected group is less than the batch capacity of the available machine, go to Step 4. Otherwise, form a batch with the lots with smaller priority values and schedule the batch on the machine, and exit from the procedure.

Step 4. Select a batching alternative with the smaller total weighted waiting time between two alternatives for batching, one with the currently available lots and the other with the currently available lots and a lot that is expected to arrive at DBW first. Check whether there is any time-constrained lot in the batch of the first alternative for which the waiting time constraint will be violated if processing of the batch is delayed until a new lot arrives. If there is one, schedule the batch of the first batch alternative on the available machine immediately; otherwise, keep the available machine idle until a new lot arrives at DBW. Exit from the procedure.

3.3. Algorithms for the pair of a batch workstation and a batch workstation

The scheduling procedure for the downstream batch-processing workstation (DBW) of the batch-batch pair is identical to that for the DBW of serial-batch pairs. In the procedure for the UBW, when a machine becomes available at the upstream batch-processing workstation (UBW), priorities of the lots that are waiting in the queue are computed using a priority rule (ES/RW2 is used here too). Then, the average priority values of wafer groups are computed by using priority values of time-constrained lots that satisfy condition (1) and non-time-constrained lots, and a wafer group is selected for batching. Batch scheduling is done as in MDBH.

The procedure for batch scheduling at the UBW can be summarized as follows.

Procedure 4. (*Batch scheduling procedure for UBW*)

Step 1. When a machine becomes available at UBW, say workstation *k*, compute priorities of the lots that are waiting in the queue of the workstation using a priority rule, ES/RW2.

Step 2. For each wafer group associated with lots that are waiting at workstation *k*, compute the average priority value of associated time-constrained lots that satisfy condition (1) and non-time-constrained lots. Select a group with the least average priority value, and let f^* be the index of the selected group.

Step 3. If the number of available lots of group f^* is less than the batch capacity, go to Step 4. Otherwise, form a full batch with the lots with smaller average priority values and schedule the batch on the available machine, and exit from the procedure.

Step 4. Select a batching alternative with a smaller total weighted waiting time between two batching alternatives, one with the currently available lots and the other with the currently available lots and a lot that is expected to arrive at UBW first. If the first batching alternative is selected, schedule the batch on the available machine; otherwise, keep the available machine idle until a new lot arrives at UBW. Exit.

4. Simulation experiments

We performed simulation experiments to evaluate performance of the suggested algorithms. For the experiments, we constructed a simulation model and generated problem instances based on data of a fab in a semiconductor manufacturing company in Korea. The simulation model (and the real fab data) can be summarized as follows.

1) The simulation model includes eight workstations, four serial-processing workstations (dry etching, metal, photolithography, and wet etching) and four batch-processing workstations (deposition, diffusion, ion implant, and polishing). There are approximately 530 machines including 200 batch-processing machines. Each workstation is composed of multiple identical machines.

2) There are waiting time constraints between seven pairs of workstations, as given in <Table 2>. Approximately 40% of the lots for which the time-constraints are violated are scrapped, while about 60% of those are reworked at the upstream workstation.
3) There are approximately 1100 product types processed in the fab.
4) Each product is composed of 3 to 31 layers of circuits with an average of 17, and the number of operations required for a product ranges from 36 to 388 (with an average of 192).
5) The processing time of a wafer lot on a machine ranges from 5 to 430 minutes.
6) The waiting time limits for operation pairs with time constraints are set to one of 0.5, 1, 4, 8, and 24 hours.
7) The number of wafers for each order ranges from 25 to 300, and hence, the number of wafer lots for each order ranges from 1 to 12.

In the simulation model, the orders were generated based on the data in an order list of a real fab, in which 30 orders arrive daily on average (resulting in approximately 3000 wafers a day). According to the data, the due dates of the orders can be fit to the following probability distribution,

$$D_o = a_o + \widetilde{P}_o \cdot \mathrm{TN}(2.268,\ (0.90)^2;\ 1.01,\ 6.03), \qquad (5)$$

where D_o is the due date of order o, a_o is the arrival time of order o, $\widetilde{P}_o$ is the total processing time for all operations of order o, and TN(μ, σ ; l, u) is a random number generated from the truncated normal distribution with mean μ, variance σ^2, lower limit l, and upper limit u. In the simulation tests, due dates of orders were randomly generated using (5). Note that in the real fab considered in this study, the due dates are set considering various factors such as priorities of the customers (strategic relationships with the customers), the current production plan and schedule in the fab, order volume, and/or the price of the wafers, and hence the range of the due dates is relatively large. Three scenarios were considered. There is no urgent order in scenario O1, and 5% and 10% of orders are urgent in scenarios O2 and O3, respectively. Due dates of the urgent orders were set as $D_o = a_o + \widetilde{P}_o$ for order o in scenarios O2 and O3.

The limited waiting time constraint was imposed on the operation pairs (workstation pairs) based on real data (process plans for the products) in the simulation model. For example, the constraint was imposed on 36% of pairs of consecutive diffusion and diffusion operations, approximately 23% of pairs of consecutive wet etching and diffusion operations, and 6% of pairs of consecutive dry etching and wet etching operations. Overall, approximately 5% of all operation pairs for all product types in the fab have the waiting time constraints, and for each product type 10 operation pairs have the waiting time constraints on average.

To evaluate the performance of the scheduling methods suggested in this research, we obtained benchmark solutions using the method currently used in the semiconductor wafer fab considered in this research. In the fab, a list scheduling method with a simple slack rule was used at the time when this research was conducted. Note that the priority of lot i is computed as $(d_i - R_i - t)$ in the slack rule. Also, a simple scheduling policy was used for dealing with the waiting time constraints. In the scheduling policy, there is no specific method for dealing with the waiting time constraints at upstream workstations, and the simple slack rule is used for scheduling. On the other hand, at downstream workstations, time-constrained lots are processed prior to non-time-constrained lots. In case of ties, a lot with the smallest remaining waiting time limit is scheduled first among time-constrained lots, while a lot with the least slack is scheduled first among non-time-constrained lots. At batch-processing workstations, the minimum batch size (MBS) rule was used for batching. This method will be denoted as REAL1 in this paper. It is obvious that if REAL1 is used for scheduling, time-constrained lots are processed first even if they are less urgent than non-time-constrained lots.

In addition to REAL1, we tested another method, denoted as REAL2, which is the same as REAL1 except that ES/RW2 and MDBH are used instead of the simple slack rule and MBS rule, respectively, for scheduling non-time-constrained lots. Note that ES/RW2 and MDBH are known to work well for the objective of minimizing total tardiness of orders (Kim *et al.* 2001). Recall that in the successor-focused scheduling policy (SFSP) and the interdependent scheduling policy (ISP) as well, ES/RW2 and MDBH are used as priority rules for serial-processing and batch-processing workstations, respectively, without limited waiting time constraints.

We generated 30 problem instances, 10 problems for each of the three scenarios related to the percentage of urgent orders. Four scheduling policies (REAL1, REAL2, SFSP, and ISP) were tested in the simulation experiments.

The simulation model was coded with Factor/AIM, a simulation software package developed by Pritsker Corporation, with additional user codes written in the C programming language. The simulation experiments were performed on a personal computer with a Pentium IV processor operating at 3.2GHz clock speed. In each simulation run, the period of 6 months was simulated and results of the last 5 months were used for comparison.

Values of parameters used in the algorithms were determined through a series of preliminary simulation tests. Although the results of the tests are not given here, the parameter values were set according to the results as follows: in ES/RW2, ω and δ were set to 11.0 and 0.0001, respectively; in SFSP, τ and the minimum batch size (*MBS*) at batch-processing workstation k were set to 1 hour and $B_k - 1$, respectively, where B_k is the capacity of a batch-processing machine in workstation k; and in ISP, α and β were set to 0.7 and 0.5, respectively.

Results of the simulation experiments are given in <Table 3>, which shows the average of the percentage reduction in total tardiness of the three scheduling policies (REAL2, SFSP and ISP) from that obtained from REAL1, the average percentage of (the number of) tardy orders, and the number of cases in which each policy gave the best solution. As can be seen from the table, SFSP and ISP worked better than the others. This may be because in ISP and SFSP the states of the lots waiting at the downstream workstations are considered when the scheduling decision is made at the upstream workstations as well, while in REAL1 and REAL2, the waiting time constraints are considered only at the downstream workstations. Between SFSP and ISP, ISP worked better possibly because a non-time-constrained lot may be processed prior to time-constrained lots at the downstream workstations if the former is very urgent.

As the percentage of urgent orders increases, the performance gap (in terms of the percentage reduction and the percentage of tardy orders) between SFSP and ISP becomes larger. This may be because scheduling decisions are made based on more up-to-date information in ISP than in SFSP. Recall that, under ISP, time-constrained lots are not processed prior to other lots at downstream workstations if the time-constrained lots are less urgent than those other lots so that more urgent (non-time-constrained) lots can be processed before the time-constrained lots. On the other hand, under SFSP, time-constrained lots have to be started in all cases in which the waiting time constraints would be violated if not started. That is, under ISP, more flexibility is given and more up-to-date information is used when scheduling decisions are made. REAL2 worked better than REAL1 although the same simple scheduling method is used for dealing with the limited waiting time constraints in both policies. This shows that ES/RW2 and MDBH work better than the methods used in the real system. REAL1 did not work well as expected because time-constrained lots are always processed first even though they are not urgent at downstream workstations.

To see how the relative performance of the scheduling policies changes when the number of time-constrained lots is increased, we performed an additional series of simulation tests. In this series of tests, we considered three constraint-related scenarios (scenarios C1, C2 and C3) varying the number of time-constrained lots for

Table 3. Performance of the scheduling policies

scenarios	REAL1			REAL2			SFSP			ISP		
	total tardiness		PTO‡	total tardiness		PTO	total tardiness		PTO	total tardiness		PTO
	PR†	NBR*		PR	NBR		PR	NBR		PR	NBR	
O1	0.0 (0.0)	0	4.7 (2.3)	17.1 (2.6)	0	3.8 (1.9)	37.3 (2.4)	1	2.8 (0.8)	42.5 (4.2)	9	2.7 (0.7)
O2	0.0 (0.0)	0	25.7 (2.9)	16.3 (2.5)	0	19.7 (2.5)	38.7 (3.0)	2	8.3 (3.7)	44.2 (2.4)	8	7.8 (2.8)
O3	0.0 (0.0)	0	42.9 (3.1)	17.4 (2.8)	0	33.6 (2.1)	38.9 (3.0)	1	16.8 (3.9)	48.1 (1.8)	9	15.0 (3.6)
overall	0.0	0	24.4	16.9	0	19.1	38.4	4	9.3	45.0	26	8.5

† Average and standard deviation (in parenthesis) of percentage reduction in total tardiness obtained from each scheduling policy from that obtained from REAL1

‡ Average and standard deviation (in parenthesis) of percentage of tardy orders

* The number of cases (out of 10, and out of 30 in the cells for overall results) in which the policy gave the best result

each of the three order-related scenarios (scenario O1, O2 and O3). In scenario C1, the real fab data (used in the first series of tests) were used without modification, while the numbers of operation pairs with the limited waiting time constraints were increased by 10% and 20% in scenarios C2 and C3, respectively. We generated 90 problem instances, 10 problems for each of the nine scenario combinations. <Table 4> shows the average percentage reduction in total tardiness from that of REAL1, the average percentage of tardy orders, and the average percentage of the lots for which the waiting time constraints were violated, for the four scheduling policies (REAL1, REAL2, SFSP and ISP). The results for the average percentage reduction in total tardiness are given in <Figure 1> for a better exposition.

As can be seen from the table and figure, ISP showed consistently better performance (in terms of total tardiness) than the others for all nine scenario combinations. ISP reduced total tardiness by more than a half compared to that from REAL1. Such reduction in the total tardiness is larger when there are more urgent orders and there are more time-constrained orders, when the scheduling problem is generally considered more difficult. This may be because better scheduling decisions can be made under ISP by not only the procedures for dealing with the waiting time constraints effectively but also the use of the most up-to-date information on the fab states, especially when the scheduling complexity in the fab is high. ISP worked best for the measure of the number of tardy orders as well, although there are slightly more operation pairs for which the limited waiting time constraints are violated compared with SFSP. In the simulation results, the waiting time constraints for less than 2% of the lots are violated, and the majority of wafers of which the waiting time constraints are violated can be reworked at the upstream workstations. In the fab under consideration, the cost incurred by such violation of the waiting time constraints is considered to be significantly lower than the cost incurred due to due date violation (tardiness penalty paid to the customers or loss of goodwill). Recall that we focus on the semiconductor wafer fab which produces customized products such as

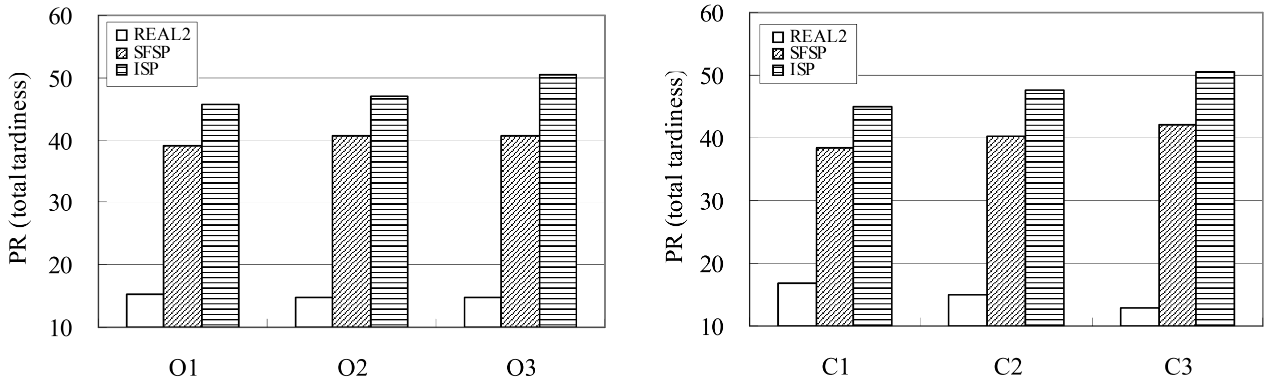
Table 4. Performance of the scheduling policies for all scenario combinations

scenario combinations		REAL1			REAL2			SFSP			ISP		
		PR†	PTO‡	PCV*	PR	PTO	PCV	PR	PTO	PCV	PR	PTO	PCV
O1	C1	0.0 (0.0)	4.7 (2.3)	0.2 (0.1)	17.1 (2.6)	3.8 (1.9)	0.2 (0.1)	37.3 (2.4)	2.8 (0.8)	0.0 (0.0)	42.5 (4.2)	2.7 (0.7)	1.1 (0.3)
	C2	0.0 (0.0)	11.7 (3.4)	1.1 (0.3)	15.8 (1.9)	9.2 (2.5)	1.0 (0.5)	39.2 (3.8)	2.2 (0.9)	0.0 (0.0)	45.2 (2.8)	1.9 (0.7)	1.1 (0.4)
	C3	0.0 (0.0)	24.3 (3.4)	1.9 (0.5)	13.3 (3.1)	17.3 (2.7)	1.9 (0.6)	41.0 (2.6)	3.9 (0.9)	0.0 (0.0)	49.4 (2.6)	3.2 (0.8)	1.4 (0.3)
O2	C1	0.0 (0.0)	25.7 (2.9)	0.6 (0.2)	16.3 (2.5)	19.7 (2.5)	0.6 (0.3)	38.7 (3.0)	8.3 (3.7)	0.0 (0.0)	44.2 (2.4)	7.8 (2.8)	1.1 (0.3)
	C2	0.0 (0.0)	37.0 (2.8)	2.2 (0.5)	15.7 (2.0)	26.4 (2.7)	1.6 (0.6)	42.2 (4.4)	13.7 (3.2)	0.0 (0.0)	48.0 (1.9)	10.6 (2.9)	1.3 (0.3)
	C3	0.0 (0.0)	52.2 (3.3)	3.1 (0.7)	12.0 (3.0)	37.4 (4.1)	2.5 (0.7)	41.5 (4.4)	20.8 (3.9)	0.0 (0.0)	48.7 (2.9)	18.2 (3.9)	1.7 (0.4)
O3	C1	0.0 (0.0)	42.9 (3.1)	1.0 (0.3)	17.4 (2.8)	33.6 (2.1)	1.0 (0.3)	38.9 (3.0)	16.8 (3.9)	0.0 (0.0)	48.1 (1.8)	15.0 (3.6)	1.3 (0.4)
	C2	0.0 (0.0)	49.8 (3.5)	3.1 (0.7)	13.4 (2.1)	47.1 (2.4)	2.5 (0.5)	39.1 (3.3)	28.1 (3.8)	0.0 (0.0)	49.8 (2.3)	22.0 (4.1)	1.4 (0.3)
	C3	0.0 (0.0)	63.9 (2.7)	3.6 (0.8)	13.1 (2.4)	54.1 (2.4)	3.1 (0.7)	43.8 (3.1)	40.0 (4.2)	0.0 (0.0)	53.6 (2.9)	34.3 (4.1)	1.8 (0.5)
overall		0.0	34.8	1.9	14.9	27.6	1.6	40.2	15.1	0.0	47.7	12.9	1.4

† Average and standard deviation (in parenthesis) of percentage reduction in total tardiness obtained from each scheduling policy from that obtained from REAL1

‡ Average and standard deviation (in parenthesis) of percentage of tardy orders

* Average and standard deviation (in parenthesis) of percentage of the lots for which the waiting time constraints are violated



(a) For different scenarios related to urgent orders (b) For different scenarios related to waiting time constraint

Figure 1. Performance of the scheduling policies for different scenarios

Table 5. Result of the paired t-tests

Scenarios		Results[†]
Order-related	O1	ISP (45.7) > SFSP (40.5) > REAL2 (15.4) > REAL1 (0.0)
	O2	ISP (47.0) > SFSP (39.7) > REAL2 (14.4) > REAL1 (0.0)
	O3	ISP (50.5) > SFSP (40.2) > REAL2 (14.5) > REAL1 (0.0)
Constraint-related	C1	ISP (45.0) > SFSP (37.6) > REAL2 (17.8) > REAL1 (0.0)
	C2	ISP (47.7) > SFSP (39.8) > REAL2 (13.9) > REAL1 (0.0)
	C3	ISP (50.6) > SFSP (43.1) > REAL2 (12.7) > REAL1 (0.0)
overall		ISP (47.7) > SFSP (40.2) > REAL2 (14.9) > REAL1 (0.0)

[†] "A > B" denotes that there is statistically significant difference at the significance level of 0.01 between policies A and B, that is, percentage reduction of policy A is larger than that of policy B. The average percentage reduction is given in the parenthesis.

Table 6. Flow time of the lots

scenario combinations		REAL1	REAL2	SFSP	ISP
O1	C1	2.13 (0.61)[†]	2.07 (0.58)	2.05 (0.53)	2.04 (0.53)
	C2	2.19 (0.65)	2.13 (0.59)	2.06 (0.55)	2.05 (0.55)
	C3	2.28 (0.73)	2.19 (0.67)	2.10 (0.56)	2.05 (0.53)
O2	C1	2.15 (0.63)	2.09 (0.60)	2.05 (0.55)	2.06 (0.60)
	C2	2.23 (0.66)	2.12 (0.64)	2.09 (0.58)	2.09 (0.60)
	C3	2.31 (0.76)	2.19 (0.71)	2.11 (0.58)	2.09 (0.58)
O3	C1	2.15 (0.66)	2.12 (0.64)	2.07 (0.60)	2.10 (0.64)
	C2	2.25 (0.69)	2.16 (0.67)	2.10 (0.59)	2.12 (0.67)
	C3	2.36 (0.79)	2.21 (0.73)	2.16 (0.62)	2.13 (0.67)
overall		2.23 (0.68)	2.14 (0.65)	2.09 (0.57)	2.09 (0.59)

[†] Average and standard deviation (in parenthesis) of the ratio of the flow time to the total processing time of the lots

system LSI products.

To see if there is significant difference in the performance of the scheduling policies in terms of total tardiness, we performed a series of paired *t*-tests, and the results are given in <Table 5>. There were significant differences in the (reduction percentages of) total tardiness at the significance level of 0.01 among the scheduling policies, including those between ISP and SFSP, for all scenarios. This means that good policies/algorithms work consistently better than the others regardless of problem characteristics.

To see if there is significant difference in the performance of the scheduling policies in terms of total tardiness, we performed a series of paired *t*-tests, and the results are given in <Table 5>.

Finally, flow times of the lots resulting from the four policies are given in <Table 6>. The table shows the average and the standard deviation of the ratio of the flow time to the total processing time of the lots. Paired *t*-tests showed that there are significant differences (at the significance level of 0.01) between the flow time under ISP and those under REAL1 and REAL2. However, there is no significant difference between the flow times of ISP and SFSP. Considering the outperformance of ISP over other methods in terms of tardiness and/or flow time, we can argue that ISP is an effective and viable method for scheduling lots in wafer fabs with waiting time constraints.

5. Concluding remarks

In this study, we considered a scheduling problem in a semiconductor wafer fabrication facility in which there are limited waiting time constraints between certain pairs of consecutive operations. We presented a new scheduling policy, called the interdependent scheduling policy, in which scheduling decisions for waiting time-constrained lots are made based on up-to-date information of the fab and progresses of wafer lots. From a series of simulation tests, it was found that the suggested scheduling policy worked better than existing policies. We expect that the scheduling policy suggested in this study can be easily adopted in real fabs with limited waiting constraints, since the suggested scheduling method employs a priority-rule-based scheduling procedure, which can be easily implemented and is generally used in practice.

Minimization of tardiness of orders was considered as the objective of the scheduling problem in this research, since meeting due dates of customers' orders was considered the most important operational objective in the fab considered in this study, which produces many different types of products according to the customers' orders. However, one may need to consider other objectives, such as those related to costs, production lead time or throughput, for other types of fabs, in which a small number of product types are produced in large volume. For instance, in many fabs that produce wafers for memory-type products such as random access memory or flash memory, there may not be tightly managed due dates for orders, or meeting due dates of customer orders may be less important. Also, cost incurred by wafer loss or rework due to the violation of the waiting time constraints may be considered more important in certain cases.

References

Anderson, E. J. and Nyirenda, J. C. (1990), Two New Rules to Minimize Tardiness in a Job Shop, *International Journal of Production Research* **28**(12), 2277-2292.

Artigues, C., Dauzere-Peres, S., Derreumaux, A., Sibille, O., and Yugma, C. (2006), A Batch Optimization Solver for Diffusion Area Scheduling in Semiconductor Manufacturing, *Proc. 2006 IFAC International Symposium on Information Control Problems in Manufacturing* 727-732.

Caumond, A. and Lacomme, P. (2008), A Memetic Algorithm for the Job-shop with Time-lags, *Computers and Operational Research* **35**(7), 2331-2356.

Chen, J. S. and Yang, J. S. (2006), Model Formulations for the Machine Scheduling Problem with Limited Waiting Time Constraints, *Jornal of Information and Optimization Sciences* **27**(1), 225-240.

Chern C. C. and Huang, K. L. (2004), A Heuristic Input Control Method for a Single-product, High-volume Wafer Fabrication Process to Minimize the Number of Photomask Changes, *Journal of Manufacturing Systems* **23**(1), 30-45.

Chun, K.-W. and Hong, Y. (1996), Batch Sizing Heuristic for Batch Processing Workstations in Semiconductor Manufacturing, *Journal of the Korean Institute of Industrial Engineers* **22**(2), 231-245.

Dabbas R. M. and Fowler, J. W. (2003), A New Scheduling Approach using Combined Dispatching Criteria in Wafer Fabs, *IEEE Transactions on Semiconductor Manufacturing* **16**(3), 501-510.

Duwayri, Z., Mollaghasemi, M., Nazzal, D., and Rabadi, G. (2006), Scheduling Setup Changes at Bottleneck Workstations in Semiconductor Manufacturing, *Production Planning and Control* **17**(7), 717-727.

Fondrevelle, J., Oulamara, A., and Portmann, M. C. (2006), Permutation Flowshop Scheduling Problems with Maximal and Minimal Time Lags, *Computers and Operations Research* **33**(6), 1540-1556.

Fowler, J. W., Hogg, G. L., and Phillips, D. T. (2000), Control of Multiproduct Bulk Server Diffusion/oxidation Processes. Part 2: Multiple Servers, *IIE Transactions* **32**(2), 167-176.

Garey, M. R. and Johnson, D. S. (1975), Complexity Results for Multiprocessor Scheduling Under Resource Constraints, *SIAM Journal on Computing* **4**(4), 397-411.

Glassey, C. R. and Weng, W. W. (1991), Dynamic Batching Heuristic for Simultaneous Processing, *IEEE Transactions on Semiconductor Manufacturing* **4**(2), 77-82.

Gurnani, H., Anupindi, R., and Akella, R. (1992), Control of Batch Processing Systems in Semiconductor Wafer Fabrication Facilities, *IEEE Transactions on Semiconductor Manufacturing* **5**(4), 319-328.

Hodson, A., Muhlemann A. P., and Price, D. H. R. (1985), A Microcomputer based Solution to a Practical Scheduling Problem, *Journal of the Operational Research Society* **36**(10), 903-914.

Hsieh, B.-W., Chen, C.-H., and Chang, S.-C. (2001), Scheduling Semiconductor Wafer Fabrication by using Ordinal Optimization-based Simulation, *IEEE Transactions on Robotics and Automation* **17**(5), 599-608.

Hung, Y. F. (1998), Scheduling of Mask Shop E-beam Writers. *IEEE Transactions on Semiconductor Manufacturing* **11**(1), 165-172

Johri, P. K. (1993), Practical Issues in Scheduling and Dispatching in Semiconductor Wafer Fabrication, *Journal of Manufacturing Systems* **12**(6), 474-485.

Joo, B.-J. and Kim, Y.-D. (2009), A Branch and Bound Algorithm for a Two-machine Flowshop Scheduling Problem with Limited Waiting Time Constraints, *Journal of the Operational Research Society* **60**(4), 572-582.

Kim, Y.-D., Bang J.-Y., An K.-Y., and Lim, S.-K. (2008), A Due-date based Algorithm for Lot-order Assignment in a Semiconductor Wafer Fabrication Facility, *IEEE Transactions on Semiconductor Manufacturing* **21**(2), 209-216.

Kim, Y.-D., Kim, J.-G., Choi, B., and Kim, H.-U. (2001), Production Scheduling in a Semiconductor Wafer Fabrication Facility Producing Multiple Product Types with Distinct Due Dates, *IEEE Transactions on Robotics and Automation* **17**(5), 589-598.

Kim, Y.-D., Kim, J.-U., Lim, S.-K., and Jun, H.-B. (1998a), Due-date Based Scheduling and Control Policies in a Multiproduct Semiconductor Wafer Fabrication Facility, *IEEE Transactions on Semiconductor Manufacturing* **11**(1), 155-164.

Kim, Y.-D., Lee, D.-H., Kim, J.-U., and Roh, H.-K. (1998b), A Simulation Study on Lot Release Control, Mask Scheduling and Batch Scheduling in Semiconductor Wafer Fabrication Facilities, *Journal of Manufacturing Systems* **17**(2), 107-117.

Kim, Y.-D., Shim, S.-O., Choi, B., and Hwang, H. (2003), Simplification Methods for Accelerating Simulation-based Real-time Scheduling in a Semiconductor Wafer Fabrication Facility, *IEEE Transactions on Semiconductor Manufacturing* **16**(2), 290-298.

Kim, S., Yea, S.-H., Kim, B. (2002), Shift Scheduling for Steppers in the Semiconductor Wafer Fabrication Process, *IIE Transactions* **34**(2), 167-177.

Kumar, P. R. (1994), Scheduling Semiconductor Manufacturing Plants, *IEEE Control Systems* 14(6), 33-40.

Lee, G.-C., Kim, Y.-D., Kim, J.-G., Choi, S.-H. (2003), A Dispatching Rule-based Approach to Production Scheduling in a Printed Circuit Board Manufacturing System, *Journal of the Operational Research Society* **54**(1), 1038-1049.

Lee, Y. H., Park, J., and Kim, S. (2002), Experimental Study on Input and Bottleneck Scheduling for a Semiconductor Fabrication Line, *IIE Transactions* **34**(2), 179-190.

Li, S., Tang, T., Collins, D. W. (1996), Minimum Inventory Variability Schedule with Applications in Semiconductor Fabrication, *IEEE Transactions on Semiconductor Manufacturing* **9**(1), 145-149.

Liao, D.-Y., Chang S.-C., Pei, K.-W., and Chang, C.-M. (1996), Daily Scheduling for R&D Semiconductor Fabrication, *IEEE Transactions on Semiconductor Manufacturing* **9**(4), 550-561.

Lu, S. C. H., Ramaswamy, D., and Kumar, P. R. (1994), Efficient Scheduling Policies to Reduce Mean and Variance of Cycle-time in Semiconductor Manufacturing Plants, *IEEE Transactions on Semiconductor Manufacturing* **7**(3), 374-388.

Min, H.-S. and Yih, Y. (2003), Selection of Dispatching Rules on Multiple Dispatching Decision Points in Real-time Scheduling of a Semiconductor Wafer Fabrication System, *International Journal of Production Research* **41**(16), 3921- 3941.

Robinson, J. K. and Giglio, R. (1999), Capacity Planning for Semiconductor Wafer Fabrication with Time Constraints between Operations, *Proc. 1999 Winter Simulation Conference*, 880-887.

Scholl, W. and Domaschke, J. (2000), Implementation of Modeling and Simulation in Semiconductor Wafer Fabrication with Time Constraints between Wet Etch and Furnace Operations, IEEE Transactions on Semiconductor Manufacturing **13**(3) 273-277.

Sheen, G. J. and Liao, L. W. (2007), A Branch and Bound Algorithm for the One-machine Scheduling Problem with Minimum and Maximum Time Lags, *European Journal of Operational Research* **181**(1), 102-116.

Sloan. T. and Shanthikumar, J. G. (2002), Using In-line Equipment Condition and Yield Information for Maintenance Scheduling and Dispatching in Semiconductor Wafer Fabs, *IIE Transactions* **34**(2), 191-209.

Sourirajan, K., Uzsoy, R. (2007), Hybrid Decomposition Heuristics for Solving Large-scale Scheduling Problems in Semiconductor Wafer Fabrication, *Journal of Scheduling* **10**(1), 41-65.

Su, L.-H. (2003), A Hybrid Two-stage Flowshop with Limited Waiting Time Constraints, *Computers and Industrial Engineering* **44**(3), 409-424.

Upasani, A. A., Uzsoy, R., and Sourirajan, K. (2006), A Problem Reduction Approach for Scheduling Semiconductor Wafer Fabrication Facilities, *IEEE Transactions on Semiconductor Manufacturing* **19**(2), 216-225.

Wein, L. M. (1988), Scheduling Semiconductor Wafer Fabrication, *IEEE Transactions on Semiconductor Manufacturing* **1**(3), 115-130.

Weng, W. W. and Leachman, R. C. (1993), An Improved Methodology for Real-time Production Decisions at Batch-process Work Stations, *IEEE Transactions on Semiconductor Manufacturing* **6**(3), 219-225.

Wikum, E. D., Llewellyn, D. C., and Nemhauser, G. L. (1994), One-machine Generalized Precedence Constrained Scheduling Problems, *Operations Research Letters* **16**(2), 87-99.

Yang, D.-L. and Chern, M.-S. (1995), Two-machine Flowshop Sequencing Problem with Limited Waiting Time Constraints, *Computers and Industrial Engineering* **28**(1), 63-70.

Yildirim, M. B., Duman, E., and Duman, D. (2006), Dispatching Rules for Allocation of Component Types to Machines in the Automated Assembly of Printed Circuit Boards, *Lecture Notes in Computer Science* **4263**(1), 55-64.