36

A LOAD BALANCING METHOD FOR DEDICATED PHOTOLITHOGRAPHY MACHINE CONSTRAINT

Arthur Shr[1], Alan Liu[1], Peter P. Chen[2]
[1]*Department of Electrical Engineering, National Chung Cheng University*
Chia-Yi 621, Taiwan, R.O.C.
arthurshr@gmail.com, aliu@ee.ccu.edu.tw
[2]*Department of Computer Science, 298 Coates Hall, Louisiana State University*
Baton Rouge, LA 70803, U.S.A.
pchen@lsu.edu

The dedicated photolithography machine constraint in semiconductor manufacturing is one of the new issues of photolithography machinery due to natural bias. In this paper, we propose the heuristic Load Balancing (LB) scheduling approach based on a Resource Schedule and Execution Matrix (RSEM) to tackle this constraint. The LB method is to schedule each wafer lot at the first photolithography stage to a suitable machine, according to the load balancing factors among machines. We describe the proposed LB scheduling method and present an example to demonstrate the proposed method and the result of the simulations to validate the approach.

1. INTRODUCTION

Semiconductor manufacturing systems are different from the traditional manufacturing operations, such as a flow-shops manufacturing system in assembly lines or a job-shops manufacturing system. In a semiconductor factory, one wafer lot passes through hundreds of operations, and the processing procedure takes a few months to complete. The operations of semiconductor manufacturing incrementally develop an IC product layer by layer.

To solve the complex and challenging scheduling problems in the semiconductor manufacturing, many queuing network scheduling polices or methods have been published to formulate the complexity of semiconductor manufacturing problems. These scheduling policies deal with the buffer competing problem in the re-entrant production line (Kumar, 1993), wherein they pick up the next wafer lot in the queue buffers when machines are becoming idle. Two scheduling policies have been proposed to reduce the mean and variance of product cycle time (Kumar, 1994) (Lu, 1994). Wein's research used a Brownian queuing network model to approximate a multi-class queuing network model with dynamic control to the process in the semiconductor factory (Wein, 1988). SDA-F, a special family-based scheduling rule, uses a rule-based algorithm with threshold control and the least slack principle

to dispatch wafer lots in photolithography stages (Chern, 2003). A study proposed a stochastic dynamic programming model for scheduling new wafer lot release and bottleneck processing by stage in the semiconductor factory (Shen, 2003). One other research used the Petri Net approach to modeling, analysis, simulation, scheduling and the control of the semiconductor manufacturing system (Zhou, 1998).

Recently, a study concerning the load balancing issue developed a load balance allocation function by applying a dynamic programming method to the machine constraint in the photolithography machines (Miwa, 2005). Two approaches were reported to use simulations to model the photolithography process, and one of them proposed a Neural Network approach to the photolithography scheduling problem (Arisha, 2004). The approach followed a qualifying matrix and the lot scheduling criteria to improve the performance of the photolithography machines. The other is to decide the wafer lots assignment of the photolithography machines when the wafer lots were released to the manufacturing system to improve the load balancing problem in the photolithography area (Mönch, 2001).

Although these scheduling polices or methods have been developed and applied in the semiconductor factories, they did not concern the dedicated photolithography machine constraint, which will cause load unbalancing among the photolithography machines in the semiconductor factory. In fact, the wafer lots in a load unbalancing semiconductor factory usually need to be switched from the highly congested machines to the idle machines. This takes much time and relies on experienced engineers to manually handle alignment problems of the wafer lots with a different situation off-line. It is inefficient to determine, one lot at a time, which wafer lot and machine need to be switched. This method cannot meet the fast-changing market of the semiconductor industry.

In this paper, motivated by the issue described above, we propose a Load Balancing (LB) scheduling approach based on the Resource Schedule and Execution Matrix (RSEM) developed in our previous research work (Shr, 2006a) (Shr, 2006b) (Shr, 2006c) to tackle the dedicated machine constraint and load balancing issue.

The paper is organized as follows: Section 2 describes the dedicated photolithography machine constraint in detail. Section 3 presents our proposed LB scheduling method based on RSEM to apply to the dedicated constraint issues of semiconductor factory. Section 4 shows the simulation results that validated our approach. Section 5 discuses the conclusion.

2. DEDICATED PHOTOLITHOGRAPHY MACHINE CONSTRAINT

One of the challenges in the semiconductor manufacturing systems is the dedicated photolithography machine constraint which happens because of the natural bias of the photolithography machine. Natural bias will impact the alignment of patterns between different layers. The smaller the dimension of the IC products (wafers), the more difficult they will be to align between different layers, especially when we move on to a smaller dimension IC for high technology products. The wafer lots passing through each photolithography stage process have to be processed on the same machine. The purpose of the limitation is to prevent the impact of natural bias and to keep a good yield of the IC product. A research considered different process

control policies including the machine dedication policy in their simulation study for semiconductor manufacturing has reported that the machine dedication policy had the worst performance of photolithography process (Akcalt, 2001). The machine dedication policy is similar to the dedicated machine constraint we are discussing here. Figure 1 describes the circumstance of the dedicated machine constraint.

With this dedicated photolithography machine constraint, when wafer lots enter each photolithography operation stage and if the wafer lots have been dedicated to the machine ***X***, they need to be processed and wait for ***X***. They cannot be processed by other machines, e.g., the machine ***Y***, even if ***Y*** is idle. On the other hand, when wafer lots enter into other stages, without any machine constraints, the wafer lots can be scheduled to any machine of ***A***, ***B*** or ***C***.

The photolithography process is the most important process in semiconductor manufacturing since the yield of IC products is always dependent on a good photolithography process, while at the same time the process can also cause defects. Not surprisingly, the performance of the factory will rely on the photolithography machines. Therefore, the dedicated photolithography machine constraint is the most important challenge to improve productivity and fulfill the request for customers as well as the main contributor to the complexity and uncertainty of semiconductor manufacturing.

This load balancing issue is derived mainly from the dedicated photolithography machine constraint. This is because once the wafer lots have been scheduled to one of the machines at the first photolithography stage, they must be assigned to the same machine in the subsequent photolithography stages until they have passed the last photolithography stage. Therefore, any unexpected abnormal events or breakdown of one machine will cause a pile-up of many wafer lots waiting for the machine and make it critical to the factory. Some of the photolithography machines will become idle and remain so for a while, due to the fact that no wafer lots can be processed, and the other will always be busy while many wafer lots bound to this machine are awaiting processing. As a result, some wafer lots will never be delivered to the customer on time, and the performance of the factory will have been decreased and impacted.

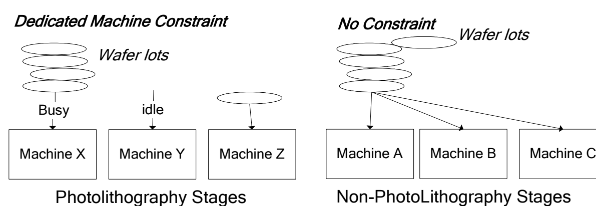


Figure 1–Dedicated photolithography machine constraint

3. LOAD BALANCING SCHEDULING APPROACH

In this section, we apply the Load Balancing (LB) scheduling approach to the dedicated machine constraint of the photolithography machine in semiconductor

manufacturing. The LB approach uses the RSEM as a tool to represent the temporal relationship between the wafer lots and machines during each scheduling step. We use an example to demonstrate the LB approach in the following.

The RSEM construction process consists of three modules including *Task Generation*, *Resource Calculation*, and *Resource Allocation* module. The first module is to model the tasks for the scheduling system and it is represented in a two-dimensional matrix. We generate the two-dimension matrix for the tasks that are going be processed by machines. One dimension is reserved for the tasks $t_1, t_2, \ldots, t_n$, the other is to represent the periodical time events (or steps), $s_1, s_2, \ldots, s_m$. Each task has a sequential *Process Pattern* to represent the resources it needed during the process sequence from a raw material to a product and we put the process pattern in an array. We define each type resource as r_k, $k=1$ *to* o, e.g., the process pattern, $r_1, r_2, \ldots, r_o$, means that a particular task needs the resources in the sequence of r_1 first and r_2 following that until r_o is gained. Therefore, the matrix looks as follows:

	s_1	s_2	.	.	.	.	.	s_j	.	.	s_m
t_1	r_1	r_2	r_3	..	..	..	r_k.	..	..	r_o	..
t_2		r_3	r_4	..	..	r_k	..	..	..	..	..
.							..	..			
t_i					r_3	r_4	..	..	r_k	..	
.							.	.			
t_n					..	..	r_k	..	..	..	

The symbol r_k in the matrix entry $[t_i, s_j]$ is to represent the task t_i needs the resource r_k at the time s_j. If t_i starts to the process at s_j in the system and the total numbers of steps needed for t_i is p, we will fill its process pattern into the matrix from $[t_i, s_j]$ to $[t_i, s_{j+p-1}]$ with r_k. All the tasks, $t_1, \ldots t_n$, follow the illustration above to form a task matrix in the task generation module. To represent the dedicated machine constraint, the symbol r_k^x, a replacement of r_k, is to represent that t_i has been dedicated to a particular instance x of a resource type r_k at s_j. One more symbol w_k represents the wait situation when the r_k cannot serve t_i at s_j. We will insert this symbol in the *Resource Allocation* module later.

The *Resource Calculation* module is to summarize the value of each dimension as the factors for the scheduling rules of the *Resource Allocation*. For example, we can determine how many steps t_i is processed by counting task pattern of the row, t_i row in the matrix. We can also realize how many wait steps t_i has had by counting w_k from the starting step to the current step in that row of the matrix.

We need to generate the task matrix, obtain all the factors for the scheduling rules, and build up the scheduling rules before starting the execution of the *Resource Allocation* module. The module is to schedule the tasks to the suitable resource according to the factors and predefined rules. To represent the situation of waiting for r_k, i.e., when the resource of r_k is not available for t_i at s_j, then we will not only insert w_k in $[t_i, s_j]$ of the matrix, but also shift one step for the process pattern following of t_i.

After obtaining the process flow for customer product from the database of semiconductor manufacturing, we can use a simple program to transform the process flow into the process pattern and task matrix representation. For a typical factory, there are thousands of wafer lots and hundreds of process steps. There is an example to transform the process pattern of wafer lots into a task matrix. We let r_2 represent

the photolithography machine and the others are to represent non-photolithography machines. The symbol r_2^x in the matrix entry $[i,j]$ is to represent t_i need of the photolithography machine x at s_j with dedicated machine constraint, while the r_k ($k \neq 2$) in $[i,j]$ is to represent t_i needing the machine type k at s_j without dedicated machine constraint. There is no assigned machine number for the photolithography machine before the wafer lot has passed first photolithography stage.

Suppose that $r_1r_3r_2r_4r_5r_6r_7r_2r_4r_5r_6r_7r_8r_9r_1r_3r_2r_4r_5r_6r_7r_3r_2r_8r_9$ is the process pattern of t_1 and it starts to release to the factory at s_1, we will fill its process pattern into the task matrix from $[t_1,s_1]$ to $[t_1,s_{25}]$, which indicates that t_1 needs the resource r_1 at the first step, resource r_3 at the second step, and r_9 at the last step. The photolithography process, r_2, in this process pattern has not dedicated to any machine yet and total of the steps for t_1 is 25. The wafer lot t_2 in the following task matrix has the same process pattern as t_1 has but it starts at s_3. The wafer lot t_i in the task matrix starts from s_8 and it requires the photolithography machine, but the machine is different from the machine t_2 needed at s_{10}, i.e., t_2 needs the machine m_1, while t_i has not been dedicated to any machine yet. Moreover, at s_{11}, t_2 and t_i might compete for the same resource r_4 if r_4 is not enough for them at s_{11}.

	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9	s_{10}	s_{11}	s_{12}	s_{13}	s_{14}	s_{15}	s_{16}	s_{17}	s_{18}	s_{19}	s_{20}	s_{21}	s_{22}	s_{23}	s_{24}	s_{25}	..	..	..	s_m
t_1	r_1	r_3	$\underline{r_2}$	r_4	r_5	r_6	r_7	$\underline{r_2}$	r_4	r_5	r_6	r_7	r_8	r_9	r_1	r_3	$\underline{r_2}$	r_4	r_5	r_6	r_7	r_3	$\underline{r_2}$	r_8	r_9				.
t_2			r_1	r_3	$\underline{r_2}$	r_4	r_5	r_6	r_7	$\underline{\mathbf{r}_2}^1$	$\mathbf{r}_4$	r_5	r_6	r_7	r_8	r_9	r_1	r_3	$\underline{r_2}$	r_4	r_5	r_6	r_7	r_3	$\underline{r_2}$	r_8	r_9		
t_i								r_1	r_3	$\underline{\mathbf{r}_2}$	$\mathbf{r}_4$	r_6	r_5	..	..	..	..	..	..	..	..	..	..	..	..	..	..	..	
..												..	..																

We need to obtain some factors for our proposed LB scheduling approach. The definitions and formulae of these factors are as follows:

W: wafer lots in process,
P: numbers of photolithography machines,
O: types of machine (resource)

(1) How many wafer lots will need the k type machine (photolithography machine, k = 2) at s_j:

$$RR(r_k^x, s_j) = \sum_{t_i \in W} \begin{cases} 1 & \text{if } [t_i, s_j] = r_k^x \\ 0 & \text{otherwise} \end{cases}, \quad 1 \le x \le P$$

(2) Step Count:
(2.1) How many wait steps t_i has had before s_j:

$$WaitStep(t_i) = \sum_{j=start}^{current\ step} \begin{cases} 1 & \text{if } [t_i, s_j] = w_k \\ 0 & \text{otherwise} \end{cases}, \quad 1 \le k \le O$$

(2.2) How many steps t_i will have.

$$Steps\,(t_i) = \sum_{j=start}^{end\ step} \begin{cases} 1 & \text{if } [t_i, s_j] \neq null \\ 0 & \text{otherwise} \end{cases}$$

(3) The load factor of the machine m_x, wafer lots × remanding photolithography stages

$$Load\,(m_x, s_j) = \sum_{t_i \in W} \{ t_i * R(t_i) \mid pm(t_i) = m_x \}$$

(3.1) How many remaining photolithography stages of t_i:

$$R(t_i) = \sum_{j=current}^{end_step} \begin{cases} 1 & \text{if } [t_i, s_j] = r_2^x \\ 0 & \text{otherwise} \end{cases}, \quad 1 \le x \le P$$

(3.2) $pm(t_i)$: the photolithography machine number that t_i is dedicated to.

We will use these factors in the third module. In this example, load is defined as the wafer lots limited to the machine m multiply their remaining photolithography layers stage. The larger load factor means that the more required service from the wafer lots which have been limited to this machine. The LB uses these factors to schedule the wafer lots to a suitable machine at the first, unconstrained, photolithography stage.

Suppose we are currently at s_j, and the LB scheduling system will start from the photolithography machine. First, we check if there is any wafer lot which requires the resource of the photolithography machine at the first stage. LB will assign the m_x with smallest Load(m_x, s_j) (formula (3)) for them one by one. After that, these wafer lots have been dedicated to a photolithography machine. For each m_x, the LB will select one of the wafer lots dedicated to m_x which has the largest WaitStep(t_i) for it. Load(m_x, s_j) of m_x will be updated after these two processes. The other wafer lots dedicated to each m_x which can not be allocated to the m_x at this step s_j will insert a w_2 for them in their process pattern. For example, at s_{10}, t_i has been assigned to m_1, therefore, t_{i+1} will have a w_2 inserted into at s_{10}, and then all the following required resource of t_{i+1} will shift one step. The following matrix shows the situation. All the other types of machines will have the same process, without the need of being concerned with the dedicated machine constraint. The LB will schedule one of the wafer lots with the largest WaitStep(t_i), then the second largest one, and so on, for each machine. Similarly, the LB will insert a w_k for the wafer lots not assigned to machines r_k.

	s_9	s_{10}	s_{11}	s_{12}	s_{13}	s_{14}	..	..	s_j	..	s_m
					..	..					
t_i	..	r_2^1	r_4	r_5	r_6	r_7	..	..			
t_{i+1}	..	w_2	r_2^1	r_4	r_6	r_5	..	..	..	..	
..		↑	→	→	→	→					

To better understand our proposed scheduling process, the flowchart of the RSEM is shown in Figure 2.

The process of using the RSEM starts from the *Task Generation* module, and it will copy the predefined task patterns of tasks into the matrix. Entering the *Resource Calculation* module, the factors for the tasks and resources will be brought out at the current step. This module will update these factors again at each scheduling step. The execution of the scheduling process is in the *Resource Allocation* module.

When we have done the schedule for all the tasks for the current step, we will return to check for new tasks and repeat the whole process again by following the flowchart. We will exit the scheduling process when we reach the final step of the last task if there is still no new task appended to the matrix. After that, the scheduling process will restart immediately when the new tasks arrive in the system.

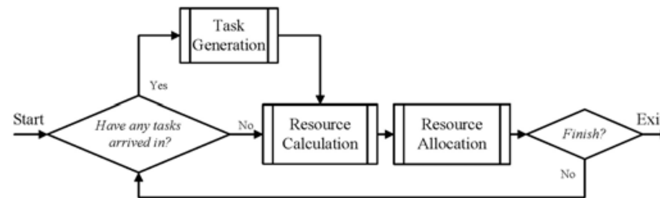


Figure 2–Flowchart of the RSEM

We assume that all the resource types for the wafer lot will have the same process time in this example, i.e. all the steps will have the same time duration. The assumption simplifies the real semiconductor manufacturing system, but could be focused on the issue of the dedicated machine constraint. However, it is not difficult to approach the real world on a smaller scale time step. Most scheduling polices or methods can provide neither the exact allocation in accepted time, nor a robust and systematic resource allocation strategy. We use the RSEM to represent the complex tasks and to allocate resources by the simple matrix calculation. This reduces much of the computation time for the complex problem.

4. SIMULATION RESULT

We have done two types of simulations for a Least Slack (LS) time scheduling policy and our LB scheduling approach. The LS policy has been developed in the research, Fluctuation Smoothing Policy for Mean Cycle Time (FSMCT) (Kumar, 2001). FSMCT scheduling policy is for re-entrant production lines. The LS scheduling policy sets the highest priority to a wafer lot whose slack time is the smallest in the queue buffer of one machine. When the machine is going to idle, it will select the highest priority wafer lot in the queue buffer to service next. The entire class of LS policies has been proven stable in a deterministic setting (Kumar, 1994) (Lu, 1991) without the dedicated constraint issue. However, the simulation result shows that our proposed LB is better than the LS approach. For simplifying the simulation to easily represent the scheduling approaches, we have made the following assumptions:

1. *Each wafer lot has the same process steps and quantity.*
2. *All photolithography and other stages have the same process time.*
3. *There is no breakdown event in the simulations.*
4. *There is unlimited capacity for non-photolithography machines.*

We have implemented a simulation program in Java and run the simulations on the NetBeans IDE 4.1. Figure 2 depicts our simulations result. Our simulations are to set up six to eleven photolithography machines and 1000 wafer lots. Each wafer lot in the simulations has 85 steps, and 20 of them are photolithography stages. For example, the task pattern is as follows:

$r,r,r_2,r,r,r,r,r,r_2,r,r,r,r,r_2,r,r,r,r,r,r,r,r,r,r_2,r,r,r,r,r,r_2,r,r,r,r,r,r,r_2,r,r,r,r,r,r_2,r,r,r_2,r,r,$
$r_2,r,r,r_2,r,r,r_2,r,r,r_2,r,r,r_2,r,r,r_2,r,r,r_2,r,r,r_2,r,r,r_2,r,r,r_2,r,r,r_2,r,r,r_2,r,r$

We let the symbol r represent the non-photolithography stage and r_2 the photolithography stage. In the following task matrix, the wafer lot t_2 starts to process in the simulation when t_1 has passed two steps (s_3). The wafer lot t_3 starts when t_1 has passed three steps (s_4).

$s_1, \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots. s_m$
t_1: $r,r,r_2,r,r,r,r,r,r_2,r,r,r,r,r_2,r,r,r,r,r,r,r,r,r,r_2,r,r,r,r,r,r_2,r,r,r,r,r,r,r_2,r,r,r,r,r,r_2, \dots\dots$
t_2: $r,r,r_2,r,r,r,r,r,r_2,r,r,r,r,r_2,r,r,r,r,r,r,r,r,r,r_2,r,r,r,r,r,r_2,r,r,r,r,r,r,r_2,r,r,r,r,r,r_2, \dots..$
:
t_i $r,r,r_2,r,r,r,r,r,r_2,r,r,r,r,r_2,r,r,r,r,r,r,r,r,r,r,r_2..$
:
t_{1000}: :

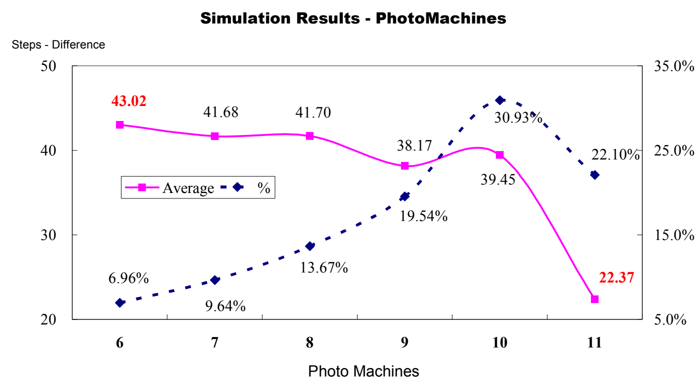
In the simulations, the wafer arrival rate between two wafer lots is a Poisson distribution. We ran ten-iteration for each simulation and the simulations result is shown in (a) and (b) of Figure 3.

We applied the LS and LB methods to select the next wafer lot to process in the simulations. When the wafer lot needs to wait for its dedicated photolithography machine, we will insert a w_r in the task pattern of the wafer lot to represent the situation. After finishing the simulations, we count the pattern of wafer lots to know how much time they have used. Although the simulations are simplified, they reflect the real situation we have met in the factory. After applying the LS method to the above simulations and counting the required resource (the formula $RR(r_k^x, s_j)$ in Section 3, $k = 2$, $x = 6$ to 11) for the photolithography machines at each step we can realize that the load of these machines becomes unbalanced during the simulations.

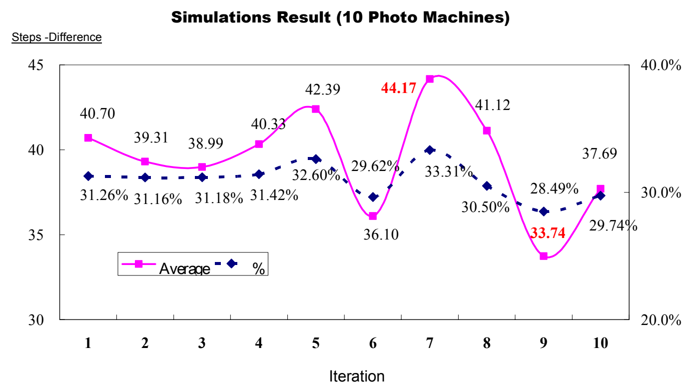
It is not difficult to extend the simulation with more wafer lots and stages of photolithography or non-photolithography. Moreover, we can use different numbers of r_2, e.g., r_2, r_2r_2, or $r_2r_2r_2r_2$, together for the task patterns to represent different process time for different photolithography stage. Both LB and LS scheduling approaches are applied to the same task matrix during each simulation generated by the Task Generation module described in Section 3. The result of simulation in Figure 3(a) shows that in terms of the outperformed steps and percentage, the LB method is better than the LS method set with 10 machines. While Figure 3(b) shows that in terms of the outperformed steps and percentage, the LB method is better than the LS method set with different number of machines.

By comparing the mean of cycle time, the LS has an average 39.45 steps more than the LB and that is the LB is better than the LS 30.93% on average in the simulation using 10 machines. The simulations result of different photolithography machines indicates that when the capacity of the photolithography area in the

manufacturing system is overloaded (the number of the photolithography machine for these simulations is less than or equal to 10), the more the photolithography machines, the better the LB method performs than LS method dose. On the other hand, when the capacity of the photolithography area in the manufacturing system is under loaded, the advantage of LB will decrease, but LB is still better than LS 22.37 steps and 20.10% in the simulations.



(a) 10 iterative simulations



(b) Simulations of different photolithography machines
Figure 3–Simulations results of LB and LS scheduling approaches

5. CONCLUSION

To provide the solution to the issue of dedicated photolithography machine constraint, the proposed Load Balancing (LB) scheduling method has been presented. Along with providing the LB scheduling method to the dedicated machine constraint, we have also presented a novel model–the representation and manipulation method for the task patterns. In addition, the simulations have also shown that the proposed LB scheduling method was better than the LS method. The

advantage of the LB method is that we could easily schedule the wafer lots by simple calculation on a two-dimensional matrix, RSEM.

6. ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their valuable comments. This research was partially supported by the U.S. National Science Foundation grant No. IIS-0326387. This research was also supported in part by the Ministry of Education under grant EX-91-E-FA06-4-4 and the National Science Council under grant NSC-94-2213-E-194-010 and NSC-92-2917-I-194-005. One of us, A. Shr, is grateful to Ms. Victoria Tangi for English proofreading.

7. REFERENCES

1. Akcalt, E, Nemoto, K, Uzsoy, R. Cycle-time improvements for photolithography process in semiconductor manufacturing. IEEE Transactions on Semiconductor Manufacturing 2001; 14, 1: 48-56.
2. Arisha, A, Young, P. "Intelligent Simulation-based Lot Scheduling of Photolithography Toolsets in a Wafer Fabrication Facility". 2004 Winter Simulation Conference 2004; 1935-1942.
3. Chern, C, Liu, Y. Family-Based Scheduling Rules of a Sequence-Dependent Wafer Fabrication System. In IEEE Transactions on Semiconductor Manufacturing 2003; 16, 1: 15-25.
4. Kumar, PR. Re-entrant Lines. In Queuing Systems: Theory and Applications, Special Issue on Queuing Networks 1993; 13, 1-3: 87-110.
5. Kumar, PR. Scheduling Manufacturing Systems of Re-Entrant Lines. Stochastic Modeling and Analysis of Manufacturing Systems, David D. Yao (ed.), Springer-Verlag, New York 1994: 325-360.
6. Kumar, S, Kumar, PR. Queuing Network Models in the Design and Analysis of Semiconductor Wafer Fabs. In IEEE Transactions on Robotics and Automation 2001; 17, 5: 548-561.
7. Lu, SCH, Ramaswamy, D., Kumar PR. Efficient Scheduling Policies to Reduce Mean and Variance of Cycle-time in Semiconductor Manufacturing Plants. In IEEE Transactions on Semiconductor Manufacturing 1994; 7, 3: 374-385.
8. Lu, SH, Kumar, PR. Distributed Scheduling Based on Due Dates and Buffer Priorities. In IEEE Transactions on Automatic Control 1991; 36, 12: 1406-1416.
9. Miwa, T, Nishihara, N, Yamamoto, K. Automated Stepper Load Balance Allocation System. IEEE Transactions on Semiconductor Manufacturing 2005; 18, 4: 510- 516.
10. Mönch, L, Prause, M, Schmalfuss, V. "Simulation-Based Solution of Load-Balancing Problems in the Photolithography Area of a Semiconductor Wafer Fabrication Facility". 2001 Winter Simulation Conference 2001: 1170-1177.
11. Shen, Y., Leachman, R.C. Stochastic Wafer Fabrication Scheduling. In IEEE Transactions on Semiconductor Manufacturing 2003; 16, 1: 2-14 .
12. Shr, AMD, Liu, A, Chen, PP. "A Load Balancing Scheduling Approach for Dedicated Machine Constraint". 8th International Conference on Enterprise Information Systems, Paphos, Cyprus, May 2006a (to appear).
13. Shr, AMD, Liu, A, Chen, PP. "A Heuristic Load Balancing Scheduling Method for Dedicated Machine Constraint". 19th International Conference on Industrial, Engineering & Other Applications of Applied Intelligent Systems (IEA/AIE'06), Annecy, France, June 2006b (to appear).
14. Shr, AMD, Liu, A, Chen, PP. "Load Balancing among Photolithography Machines in Semiconductor Manufacturing". 3rd International Conference on Intelligent Systems (IEEE IS'06), London, England, September 2006c (to appear).
15. Wein, L. M. Scheduling Semiconductor Wafer Fabrication. In IEEE Transactions on Semiconductor Manufacturing 1988; l, 3: 115-130.
16. Zhou, M. and Jeng, MD. Modeling, Analysis, Simulation, Scheduling, and Control of Semiconductor Manufacturing System: A Petri Net Approach. In IEEE Transactions on Semiconductor Manufacturing 1998; 11, 3: 333-357.