Backward Sequence Analysis for Single-Armed Cluster Tools

Jun-Ho Lee,[1] **Hyun-Jung Kim**[2*]
[1]School of Business, Konkuk University, Seoul 05029, Republic of Korea
[2]Dept. of Systems Management Engineering, Sungkyunkwan University, Suwon 16419, Republic of Korea
junholee@konkuk.ac.kr, kim.hj@skku.edu

Abstract

This paper analyzes a backward sequence for single-armed cluster tools with processing time variations. A fundamental cycle is defined with the backward sequence, and a formula for the cycle time is derived by considering processing time variations. Then conditions for which the backward sequence is optimal are developed. The upper bound on the average cycle time from the backward sequence is also analyzed. We then show experimentally that the sequence performs well even with processing time variations.

Introduction

Cluster tools for semiconductor manufacturing processes such as oxidation, photolithography, or etching, consist of multiple processing modules (PMs), a transport robot, and loadlocks for loading and unloading wafer cassettes as illustrated in Fig. 1. More than 70 % of semiconductor tools in one of the largest semiconductor fabs in Korea have similar configurations to the cluster tools. When a wafer cassette with 25 wafers is loaded into the loadlock, the robot transports each wafer to PMs. After a wafer finishes processing, it is transported to the next PM or loadlock by the robot. The cluster tool scheduling problem is the same as a flow shop with identical jobs and a material handling robot, and its scheduling decision is to determine a robot task sequence.

The robot in a single-armed cluster tool usually follows a backward sequence in practice due to its simplicity, predictability, and traceability. The robot in the backward sequence, when there are n PMs, first unloads a wafer from PM_n, transports it, and loads it into the loadlock. It then goes to PM_{n-1}, unloads a wafer, transports it, and loads it into PM_n. The robot repeats the tasks until it unloads a wafer from the loadlock and loads it into PM_1. In this study, we assume that each wafer is moved from one PM to the next and one wafer is completed and loaded into the loadlock in a cycle. Hence, multiple cycles are not considered. If we assume identical PMs in each step, there can be multiple cycles, which then requires the modification of the backward sequence and its optimality conditions. Reactive

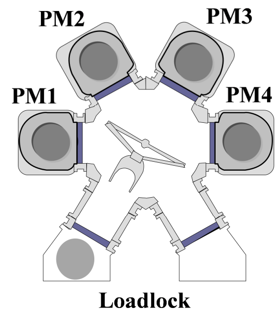

Figure 1: Single-armed cluster tool

policies for scheduling cluster tools in the presence of uncertainty can also be handled with a Markov decision process or other heuristic algorithms.

It is proved that the backward sequence provides an optimal cycle time in single-armed cluster tools with deterministic processing times under the condition of $\max_j p_j + 2u + 2l + 3t \geq (n+1)(u+l+2t)$ where p_j, u, l, t and n indicate the processing time in PM_j, unloading, loading, and transporting times of the robot, and the number of process steps, respectively. We note that the robot task times, u, l, and t are applied for not only a PM but also for the loadlock. In the condition, the left hand side, $\max_j p_j + 2u + 2l + 3t$, indicates the minimum time required for a PM to process a wafer in a cycle. After processing as much as p_j, the wafer is unloaded, transported, and loaded to the next PM ($u+l+t$), and then the robot moves to PM_{j-1}, unloads a wafer, transports it and loads it into PM_j ($u+l+2t$). The right hand side, $(n+1)(u+l+2t)$, is the minimum robot task time required in a cycle from the backward sequence. The robot should move a wafer from each PM as many as $n+1$. Hence, the condition, $\max_j p_j + 2u + 2l + 3t \geq (n+1)(u+l+2t)$, indicates the minimum time (or workload) required for a PM in a cycle is larger than or equal to the robot task time of the backward sequence. Hence, the cycle time, equal to the workload of PMs, is obtained.

However, its performance is not guaranteed with processing time variations. In practice, the processing durations of identical wafers in a PM are often different depending on the

*Corresponding author

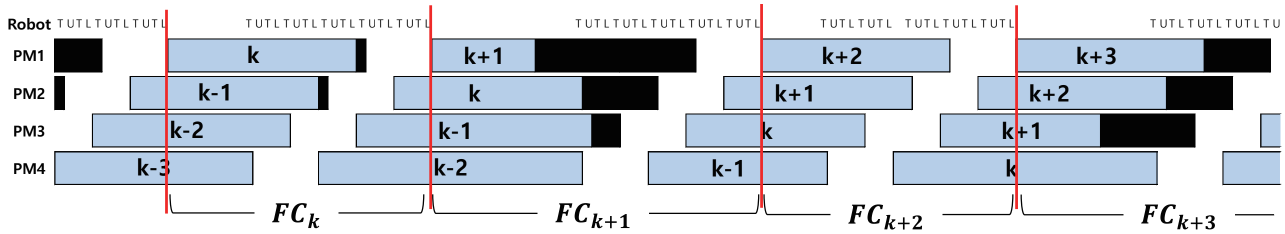

Figure 2: A schedule with time variations

machine or wafer states or process control parameters. Regarding to this issue, tool engineers want to be sure of using the sequence in fabs, and hence we analyze the performance of the backward sequence for the first time and provide analytical results on the sequence. This research can be very helpful for not only tool engineers but also researchers interested in scheduling automated manufacturing systems. We develop a formula for the cycle time, optimality conditions, and the upper bound on the average cycle time of the sequence. We then show experimentally that the backward sequence performs well even with processing time variations.

Literature Review

There have been numerous papers on cyclic scheduling of cluster tools. (Rostami, Hamidzadeh, and Camporese 2001) developed an optimal periodic scheduler for cluster tools with time window constraints. (Wu et al. 2008) and (Qiao et al. 2015) further analyzed the cluster tools with wafer residency time constraints by developing feasibility conditions and optimal robot task sequences. (Lee, Kim, and Lee 2015) examined concurrent processing of two different types of wafer lots by developing efficient robot task sequences. (Li and Fung 2016) considered two-cluster tools with residency time constraints and derived a mathematical programming model for optimal K-unit cyclic schedules. (Kim, Lee, and Lee 2016) examined the schedulability of given schedules with time window constraints and bounded time variations by modeling the problem with Petri nets. (Lee and Kim 2017) and (Kim and Lee 2019) analyzed the completion time of wafer lots in single-armed and dual-armed cluster tools, respectively. Most of the previous studies on single-armed cluster tools have assumed the backward sequence of the robot and then analyzed the given schedules. Some studies have addressed cyclic scheduling of robotic cells or hoist systems that have similar configurations to cluster tools (Che and Chu 2005), (Zarandi and Fattahi 2013). A survey on scheduling problems of semiconductor manufacturing tools can be found in (Mönch et al. 2011) and (Pan et al. 2018).

Even though there have been many papers on cyclic scheduling of cluster tools, the performance of the backward sequence in single-armed cluster tools with time variations is still an open question.

Problem Description

A cluster tool conducts n process steps, each of which is performed by PM_j where $1 \leq j \leq n$. A wafer visits each PM in a fixed order sequentially, and each PM performs different operations. The realized processing time for step j, p_j, is obtained between P_j^a and P_j^b; i.e., $p_j \in [P_j^a, P_j^b]$, $1 \leq j \leq n$. The symbol p_j^i is used for the processing time of the ith wafer in PM_j.

Fig. 2 shows a process schedule in a single-armed cluster tool with four PMs. The number in each bar is used for wafer identification, and the black bars indicate the wafer delays in PMs due to the late arrival of the robot. The symbols, U, L, and T, represent the robot unloading, loading, and transportation tasks, respectively. The backward sequence is used in the schedule. The processing times in Fig. 2 have variations. The red lines indicate the time when a new wafer is loaded into PM_1, and the length between two adjacent lines becomes a cycle time. We can observe that the cycle time is different from one another in Fig. 2.

We define a fundamental cycle k, FC_k, as the cycle that starts when the kth wafer is loaded into PM_1 and ends when the $k+1$th wafer is loaded into the same PM. The analytic methods and results of this paper can still be used even if the fundamental cycle is defined with different PMs. We use C_k as the cycle time of FC_k. We now analyze the cycle time for each fundamental cycle with the backward sequence. We do not consider revisiting or time window constraints.

Cycle Time Analysis of Fundamental Cycle

From the definition of the fundamental cycle, FC_k starts when the kth wafer is loaded into PM_1. At this point in time, the $k-1$th, $k-2$th, $\cdots$, $k-n+1$th wafers are in PM_2, PM_3, $\cdots$, and PM_n, respectively. With the backward sequence, the $n-1$ wafers are transported to the next PM in order, and FC_k ends when the $k+1$th wafer is loaded into PM_1. We denote $p_j^{'i}$ as the remaining processing time of the ith wafer in PM_j at the loading timing of the $i+j-1$th wafer into PM_1.

Theorem 1 *The cycle time for FC_k, C_k, from the backward sequence is obtained as follows:*

$$C_k = \max \left\{ \begin{array}{r} p_1^k + 2u + 2l + 3t, \\ \max\{\cdots\max\{\max\{\max\{\max(p_n^{'k-n+1}, t) \\ +u + l + 2t, p_{n-1}^{'k-n+2}\} + u + l + 2t, \\ p_{n-2}^{'k-n+3}\} + u + l + 2t, p_{n-3}^{'k-n+4}\}\cdots, \\ p_2^{'k-1}\} + u + l + 2t + 2u + 2l + 3t \end{array} \right\} \quad (1)$$

We first provide a simple example for computing a cycle time.

Example 1 *We compute the cycle time for FC_k in Fig. 2. In the schedule, $p_4^{'k-3}$, $p_3^{'k-2}$, and $p_2^{'k-1}$ are 9 s, 13 s, and 16 s, respectively. p_1^k is 20 s. If all the robot tasks are assumed to take 1 s, C_k is* $28(= \max(27, 20, 28) = \max(p_1^k + 2u + 2l + 3t, 5(u + l + 2t), p_4^{'k-3} + 5u + 5l + 9t))$.

From the formula for the cycle time, the optimality condition of the backward sequence with processing time variations can be obtained as follows:

Theorem 2 *The backward sequence provides an optimal cycle time for FC_k if $p_1^k \geq \max\{(n-1)(u + l + 2t) + t, \max_{1<j\leq n}\{p_j^{k-j+1}\}\}$.*

The optimality of the backward sequence should be verified with the condition in Theorem 2. To do that, we provide an example of computing the probability that p_1^k is the largest one among processing times in PMs.

Example 2 *Assume that n is 3 and $X_1 \sim U(30, 50)$, $X_2 \sim U(20, 40)$, and $X_3 \sim U(20, 50)$; the three random variables follow the uniform distribution with (P_j^a, P_j^b). We compute the probability of x_1 to be the maximum value so that the backward sequence for FC_k provides an optimal cycle time.* $P(\max(x_1, x_2, x_3) = x_1) = \int_{-\infty}^{+\infty} f_{X_1}(t)F_{X_2}(t)F_{X_3}(t)dt = \int_{30}^{40} \frac{1}{20}\frac{t-20}{20}\frac{t-20}{30}dt + \int_{40}^{50} \frac{1}{20}1\frac{t-20}{30}dt = \frac{1}{12000}[\frac{1}{3}t^3 - 20t^2 + 400t]_{30}^{40} + \frac{1}{600}[\frac{1}{2}t^2 - 20t]_{40}^{50} = 0.611$. *In this example, the backward sequence provides an optimal cycle time for FC_k with the probability of* 0.611 *when the processing times are larger than the robot workload.*

From Theorem 2, we can see that the backward sequence can be always optimal if the minimum processing time of PM_1 is larger than or equal to the maximum processing times of other PMs and the robot task times, $(n-1)(u + l + 2t) + t$.

Average Cycle Time Analysis

We now analyze the average cycle time when the number of wafers processed becomes large. We first derive a lower bound on the average cycle time.

Lemma 1 *When m identical wafers are processed consecutively, a lower bound on the average cycle time in a single-armed cluster tool is $\max_j \bar{p_j} + 2u + 2l + 3t$ where $\bar{p_j} = \lim_{m\to\infty} \frac{\sum_{k=1}^{m} p_j^k}{m}$.*

We now analyze the maximum average cycle time of the backward sequence with the processing time variations.

Theorem 3 *The upper bound on the average cycle time of the backward sequence with processing time variations is $\lim_{m\to\infty} \frac{\sum_{k=1}^{m} \max_{1\leq j\leq n} p_j^{k-j+1}}{m} + 2u + 2l + 3t$ with the assumption of $P_j^a \geq (n-1)(u + l + 2t)\ \forall j$.*

Example 3 *Assume that n is 3 and $X_1 \sim U(30, 50)$, $X_2 \sim U(20, 40)$, and $X_3 \sim U(20, 50)$. We first compute* $\lim_{m\to\infty} \frac{\sum_{k=1}^{m} \max_{1\leq j\leq n} p_j^{k-j+1}}{m}$ *as follows:* $E(Z) = \int_{-\infty}^{+\infty} tf_Z(t)dt = \int_{-\infty}^{+\infty} t(f_{X_1}(t)F_{X_2}(t)F_{X_3}(t) + F_{X_1}(t)f_{X_2}(t)F_{X_3}(t) + F_{X_1}(t)F_{X_2}(t)f_{X_3}(t))dt = \int_{20}^{30} t(0 + 0 + 0)dt + \int_{30}^{40} t(\frac{1}{20}\frac{t-20}{20}\frac{t-20}{30} + \frac{t-30}{20}\frac{1}{20}\frac{t-20}{30} + \frac{t-30}{20}\frac{t-20}{20}\frac{1}{30})dt + \int_{40}^{50} t(\frac{1}{20}1\frac{t-20}{30} + 0 + \frac{t-30}{20}1\frac{1}{30})dt = \frac{1}{12000}[\frac{3}{4}t^4 - \frac{140}{3}t^3 + 800t^2]_{30}^{40} + \frac{1}{600}[\frac{2}{3}t^3 - 25t^2]_{40}^{50} = 42.431$. *The upper bound on the average cycle time of the backward sequence is then 63.431 with the assumption of robot task times of 3 s. In this case, a lower bound from Lemma 1 is 61* $(= 40 + 6 + 6 + 9)$. *Hence, the largest difference of cycle times is 2.431.*

With the above theorem, we can obtain the upper bound on the average cycle time from the backward sequence even when there are processing time variations. Then the difference between the upper and lower bounds becomes the worst-case bound of the backward sequence. We now show that the backward sequence provides a good performance in practice with experiments.

Experimental Results

In this section, we numerically examine the performance of the backward sequence with processing time variations by comparing cycle times from simulation to lower bounds presented in Lemma 1. The processing time in PM_j is assumed to follow $U[P_j^a, P_j^b]$ where P_j^a and P_j^b are lower and upper bounds, respectively. The experiments were conducted on a personal computer with an Intel Core i7-4790 CPU with 4.00 GB RAM. For simulation, 1,000, 10,000, and 100,000, cycles are considered, and for each case first 30 % of cycles are considered to be warm-up periods, and the next 70% of cycles are examined to compute the average cycle time. In addition, the simulation was run repetitively for 30 times to evaluate the cycle time. The robot task times are assumed to be 3 s.

Table 1 shows experimental results with the cycle time (CT) and the gap from the lower bound (GAP). The gap is obtained by '(cycle time from simulation - lower bound)×100/lower bound'. Note that cases in which the backward sequence provides an optimal cycle time as indicated in Theorem 2 were intentionally excluded in our experiments. Each of instances (1)-(4) in Table 1 has the same processing time ranges for four PMs, but the gap between P_j^a and P_j^b becomes larger as the instance number increases. For example, in instance (4), the variance of processing times is about 33.3 which cannot be observed in practice, but it is also tested to see the worst-case performance. In instances (5)-(8), two PMs out of four PMs have large lower and upper bounds. We note that the numbers of PMs and wafers, processing times, and robot task times were obtained from real data in one of the largest semiconductor manufacturing fabs in Korea. Wafers are usually processed on two to four PMs in a cluster tool for quality risk reductions through earlier inspection or restrictions in gas supply piping to the tool PMs. The processing steps take from 60 to 500 s in most cases, and the robot tasks usually take 3 to 5 s.

As shown in Table 1, the gaps between the cycle time from simulation and the lower bound from Lemma 1 are very small when the variance of processing times is less than 10;

Table 1: Experimental results on various processing times that are uniformly distributed.

Instance No.	Processing Time Ranges $[P_1^a, P_1^b], [P_2^a, P_2^b], [P_3^a, P_3^b], [P_4^a, P_4^b]$	Number of Cycles 1,000 CT (s)	GAP(%)	10,000 CT (s)	GAP(%)	100,000 CT (s)	GAP(%)	LB
(1)	[60,61], [60,61], [60,61], [60,61]	81.76	0.32	81.77	0.33	81.77	0.33	81.5
(2)	[60,65], [60,65], [60,65], [60,65]	84.82	1.58	84.82	1.58	84.82	1.58	83.5
(3)	[60,70], [60,70], [60,70], [60,70]	88.65	3.08	88.65	3.08	88.65	3.08	86
(4)	[60,80], [60,80], [60,80], [60,80]	96.29	5.82	96.30	5.82	96.30	5.82	91
(5)	[70,71], [70,71], [60,61], [60,61]	91.67	0.18	91.67	0.18	91.67	0.18	91.5
(6)	[70,75], [70,75], [60,65], [60,65]	94.32	0.88	94.33	0.89	94.33	0.89	93.5
(7)	[70,80], [70,80], [60,70], [60,70]	97.69	1.76	97.66	1.73	97.67	1.73	96
(8)	[70,90], [70,90], [60,80], [60,80]	104.41	3.38	104.40	3.36	104.40	3.36	101
	Overall Results	Avg.	2.13	Avg.	2.12	Avg.	2.12	

any instances except for (4), and (8). Even for extreme cases, for example, instance (4), the gap is less than 6 %. We believe this result is highly acceptable by considering the fact that the cycle time from simulation is compared to the lower bound not to an optimal one. Furthermore, the average gap of 8 instances is about 2.1 % even though extreme cases are included. Therefore, from the results, we can conclude that the backward sequence provides near optimal cycle times when processing times are uniformly distributed.

Conclusion

We analyzed the performance of the backward sequence in single-armed cluster tools with processing time variations. The backward sequence is widely used in practice due to its simplicity, traceability, and high throughput rate under deterministic processing times. However, its performance has not been guaranteed with processing time variations. This work is the first one that provided analytical results of the backward sequence. We derived a formula for the cycle time of a fundamental cycle and optimality conditions for a certain cycle. We then developed the upper bound on the average cycle time of the sequence. We finally tested the sequence experimentally and showed that the gap from the lower bound is not large. Further research might include analyzing a swap sequence for dual-armed cluster tools with processing time variations.

Acknowledgments

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government (MSIT) (No. 2016R1D1A1B03930952, No. 2018R1C1B5045738, No. 2019R1C1C1004667).

References

Che, A., and Chu, C. 2005. A polynomial algorithm for no-wait cyclic hoist scheduling in an extended electroplating line. *Operations Research Letters* 33(3):274–284.

Kim, H.-J., and Lee, J.-H. 2019. Closed-form expressions on lot completion time for dual-armed cluster tools with parallel processing modules. *IEEE Transactions on Automation Science and Engineering* 16(2):898–907.

Kim, H.-J.; Lee, J.-H.; and Lee, T.-E. 2016. Schedulability analysis for noncyclic operation of time-constrained cluster tools with time variation. *IEEE Transactions on Automation Science and Engineering* 13(3):1409–1414.

Lee, J.-H., and Kim, H.-J. 2017. Completion time analysis of wafer lots in single-armed cluster tools with parallel processing modules. *IEEE Transactions on Automation Science and Engineering* 14(4):1622–1633.

Lee, J.-H.; Kim, H.-J.; and Lee, T.-E. 2015. Scheduling cluster tools for concurrent processing of two wafer types with PM sharing. *International Journal of Production Research* 51(12):3671–3687.

Li, X., and Fung, R. Y. K. 2016. Optimal K-unit cycle scheduling of two-cluster tools with residency constraints and general robot moving times. *Journal of Scheduling* 19(2):165–176.

Mönch, L.; Fowler, J. W.; Dauzère-Pérès, S.; Mason, S. J.; and Rose, O. 2011. A survey of problems, solution techniques, and future challenges in scheduling semiconductor manufacturing operations. *Journal of Scheduling* 14(6):583–599.

Pan, C. R.; Zhou, M. C.; Qiao, Y.; and Wu, N. Q. 2018. Scheduling cluster tools in semiconductor manufacturing: Recent advances and challenges. *IEEE Transactions on Automation Science and Engineering* 15(2):586 – 601.

Qiao, Y.; Wu, N. Q.; Zhu, Q.; and Bai, L. 2015. Cycle time analysis of dual-arm cluster tools for wafer fabrication processes with multiple wafer revisiting times. *Computers & Operations Research* 53:252–260.

Rostami, S.; Hamidzadeh, B.; and Camporese, D. 2001. An optimal periodic scheduler for dual-armed robots in cluster tools with residency constraints. *IEEE Transactions on Robotics and Automation* 17(5):609–349.

Wu, N. Q.; Chu, C.; Chu, F.; and Zhou, M. C. 2008. A Petri net method for schedulability and scheduling problems in single-arm cluster tools with wafer residency time constraints. *IEEE Transactions on Semiconductor Manufacturing* 21(2):224–237.

Zarandi, M. H. F., and Fattahi, H. M. M. 2013. Two-machine robotic cell scheduling problem with sequence-dependent setup times. *Computers & Operations Research* 5(40):1420–1434.