Characterizing Wafer Delays in Cluster Tools for K-Cyclic Schedule

KAIST
Dong-Hyun Roh and Tae-Eog Lee

2015.10.16
ISMI 2015
Table of Contents

1. Introduction
2. Preliminaries
3. Wafer Delays in Cluster Tools
4. Wafer Delay Regulation Strategies
5. Conclusion
Introduction

- **Cluster tools** are widely used **semiconductor manufacturing tools**. It usually repeats a lot of tasks cyclically. If K timing patterns appear during a work cycle of a schedule, then the schedule is called a K-cyclic schedule. **Usually, the tool has a K-cyclic schedule.**

- Some chambers in a cluster tool have **wafer residency time**, or **wafer delay** constraints. If the residency time is longer than the constraints, then the wafers cannot be used in real industries.

- **In a K-cyclic schedule, it is hard to recognize the wafer delays of PMs** because of the complexity of the schedule. Therefore it usually cannot assure the feasibility of the schedule for tight constraints.

- In this research, **we give explicit formulas for wafer delays in cluster tools.** We consider two types of cluster tools: single-armed cluster tools and dual-armed cluster tools. And we introduce two wafer delay regulation methods, which are a workload balancing and a feedback control.
Introduction

• There have been numerous works on wafer delays in cluster tools.
  1. Schedulability analysis against an upper limit on wafer delays which is waiting times within processing chambers [1], [3], [5], [6], [7], [8], [9]
  2. Stabilization and regulation of wafer delays [2], [6], [9], [10], [12], [13], [15], [17]
  3. Modelled and analyzed by TEGs and max-plus algebra [2], [4], [11], [14], [17], [18]

• However, most works consider 1-cyclic schedules.
• There were less works for identifying wafer delays itself.
• Although Baccelli et al. [16] provides a recursion for token sojourn times at places, we yet need a more direct insight on wafer delays.
Cluster Tools

- Cluster tools are widely used semiconductor manufacturing tools.
- It consists of several process modules (PMs), a wafer handling robot at the center of the tool, and loadlocks.
- The only one robot operation can perform at a time.
- We assume that the robot operation times are identical.
Petri-nets and Event Graphs

• **Petri-net**: a modelling framework for discrete event systems which consists of arcs, tokens, places, and transitions.

• **Event graph**: a kind of Petri-nets which every place has exactly a single input and output transition. Since it has no choice problem, the graph is called a *decision-free* graph.

• **Timed event graph (TEG)**: a kind of event graphs whose places have required token holding times.

• The behavior of a cluster tool can be regarded as a discrete event system, so it can be modelled by a Petri-net. **If the robot task sequence is fixed, the behavior can be modelled by a TEG.**
Timed Event Graphs

- Circuit ratio: $\sum_{\text{circuit}} \text{Token Holding Times} \div \sum_{\text{circuit}} \# \text{ of Tokens}$
- Critical circuit ratio: $\max_{\text{circuits}} \text{Circuit ratio}$
- Critical circuit: the circuit which has the critical circuit ratio.
- Example)

In cluster tools, the critical circuit means the work cycle of the bottleneck PM and the critical circuit ratio means the average cycle time ($\lambda$) of the tool.
K-cyclic Schedule

- In this research, the schedule of a cluster tool means the firing epochs of transitions of the TEG representing the behavior of the tool.

**Definition**) *K-cyclic schedule*

Let $x_{i}^{r}$ denote a $r$-th firing epoch of transition $i$. If $x_{i}^{r+k} - x_{i}^{r} = d\lambda$ is a constant but $x_{i}^{r+k} - x_{i}^{r}$ is not $\forall k, 1 \leq k \leq K$, this schedule $\{x_{i}^{r} | \forall i, r\}$ is named a *K-cyclic schedule*. And $K$ is called the *cyclicity*.

- In a cluster tool, the cyclicity $K$ is the # of parallel chambers of the bottleneck process. And in a K-cyclic schedule, each PM has $K$ values of wafer delays.

**Definition**) *Time difference of a K-cyclic schedule*

$\delta \in R^{K}$ whose $\delta_{i} := x_{j}^{i+1} - x_{j}^{i}$ where transition $j$ is one of transitions in the critical circuit is called the *time difference* of a K-cyclic schedule.

- Since the critical circuit has no delay, the firing epochs of the critical circuit or the bottleneck process determines the K-cyclic schedule. The wafer delays in PMs also dependent on the time difference.

- Since it is K-cyclic schedule, $\sum_{i=1}^{K} \delta_{i} = K \ast \lambda$ where $\lambda$ is the average cycle time of the tool.


**Time Difference**

\[ T_1 \rightarrow T_2 \rightarrow T_3 \rightarrow T_4 \rightarrow T_5 \rightarrow T_6 \rightarrow T_7 \rightarrow T_8 \]

\[ K = 3 \]

1\(^{st}\)-cycle

\[ x_4^1 \]

2\(^{nd}\)-cycle

\[ x_4^2 \]

3\(^{rd}\)-cycle

\[ x_4^3 \]

\[ \delta_1 := x_4^2 - x_4^1 \]

\[ \delta_2 := x_4^3 - x_4^2 \]

\[ \delta_3 := x_4^4 - x_4^3 \]

\[ \sum_{i=1}^{K} \delta_i = K \times \lambda = 150 \]
Single-armed and Dual-armed Cluster Tools

- The cluster tools are classified according to the number of robot arms. And each tool has its own optimal robot sequence for a serial-parallel wafer flow pattern.
- **Single-armed cluster tool**: Backward sequence
- **Dual-armed cluster tool**: Swap sequence
- In this research, we consider two robot sequences and suppose there are no parallel chambers except the bottleneck process.
TEG for the Backward Sequence

- The timed event graph for the backward sequence.
- We define the workload of $PM_i$ ($WL_i$) in the backward sequence: $p_i + 2u + 2l + 3v$ where $p_i$ : the process time of $PM_i$, $u$, $l$, $v$ : a robot task time for unloading, loading, and moving/transporting, respectively.
- The workload of a PM is the time needed of the PM to produce a wafer.
- The workload of $PM_i$ is the same as the circuit ratio of $PM_i$ in a TEG.
Wafer Delays in Single-armed Cluster Tools

Theorem 1)
For a single-armed cluster tool with a single bottleneck $PM_{i^*}$ and a cyclicity $K$, suppose the tool follows the backward sequence. Then the followings are satisfied:

1. With satisfying $\sum_k \delta_k = K \ast \lambda$, time differences $\delta_k \in [WL_i, K \ast \lambda - (K - 1) \ast WL_i]$ where $WL_i$ is the maximum workload of $PM_i$ ($i \neq i^*$).
2. For the bottleneck $PM_{i^*}$, $d_{PM_{i^*}} = 0$.
3. For downstream PMs ($i > i^*$), $d_{PM_i} = 0$.
4. For upstream PMs ($i < i^*$), $d_{PM_i} = \delta_k - WL_k$, $\forall k = 1, \ldots, K$. 
TEG for Swap Sequence

- The timed event graph for the swap sequence.
- We define the **workload of** $PM_i$ ($WL_i$) in the swap sequence: $p_i + u + l + r$ where $p_i$ : the process time of $PM_i$, $u, l, s$ : a robot task time for unloading, loading, and swap, respectively.
- The workload of $PM_i$ is the same as the circuit ratio of $PM_i$ in a TEG.
- Since $WL_{i^*}^{Swap} < WL_{i^*}^{Backward}$ generally, a dual-armed cluster tool has more productivity than a single-armed cluster tool.
Theorem 2)
For a dual-armed cluster tool with a single bottleneck $PM_i^*$ and a cyclicity $K$, suppose the tool follows the swap sequence. Then the followings are satisfied:
1. With satisfying $\sum_k \delta_k = K \cdot \lambda$, time differences $\delta_k \in [WL_R, K \cdot \lambda -$
Wafer Delays in a Single-armed Cluster Tool (Example)

- 3 PM, $K = 2$, $PM_2$ is bottleneck: Backward Sequence

1) $P_1 > P_3 \Rightarrow \delta_1 \in [WL_1, 2\lambda - WL_1]$ ($\delta_2 = 2\lambda - \delta_1$)

$$d_{PM_1} = \delta - WL_1, 2\lambda - WL_1 - \delta$$ and $$d_{PM_3} = 0, 0$$
Wafer Delays in a Single-armed Cluster Tool (Example)

- 3 PM, \( K = 2 \), \( PM_2 \) is bottleneck: Backward Sequence

2) \( P_1 \leq P_3 \Rightarrow \delta_1 \in [WL_3, 2\lambda - WL_3] \) \( (\delta_2 = 2\lambda - \delta_1) \)

\[ d_{PM_1} = \delta - WL_1, 2\lambda - WL_1 - \delta \] and \( d_{PM_3} = 0, 0 \)
Wafer Delays in a Dual-armed Cluster Tool (Example)

• 3 PM, $K = 2$, $PM_2$ is bottleneck: Swap Sequence

1) $P_1 > P_3 \Rightarrow \delta_1 \in [WL_R, 2\lambda - WL_R]$ and suppose that $WL_R \ll WL_1, WL_3$.

Note: The range between $WL_R \sim WL_3, WL_1$ is relatively large.
• 3 PM, $K = 2$, $PM_2$ is bottleneck: Swap Sequence

2) $P_1 \leq P_3 \Rightarrow \delta \in [WL_R, 2\lambda - WL_R]$ and suppose that $WL_R \ll WL_1, WL_3$.

Note: The range between $WL_R \sim WL_3$, $WL_1$ is relatively large.
Wafer Delays Regulation Strategies

- We introduce two approaches to control wafer delays: **feedback control** and **workload balancing**.

**Feedback Control**

**Workload Balancing**
Feedback Control

• Kim et al. [12][13] studied a specialized feedback control for cluster tools. By using their methodology, wafer delays of a certain PM can be controlled not to exceed the predetermined threshold regardless of the cyclicity.

• We use feedback control in a different way: control time difference $\delta$.

For example, suppose that we want to equalize wafer delays of PMs. Since delays are $\delta_k - WL_i$, the equal values of wafer delays can be obtained by equalizing $\delta_k$ as $\lambda$, for all $k \left( \sum_{j=1}^{K} \delta_j = K \times \lambda \right) \Rightarrow d_{PM_i}^k = \lambda - WL_i$

• Idea: Add an arc with a single token which has the critical circuit ratio.
  • Since $K$ is l.c.m.{$g.c.d.$(# of tokens of adjacent critical circuits)}, $K = 1$. 
Workload Balancing

- Lee et al. [18] studied about the workload balancing strategies for cluster tools.
- The key idea is to reduce the imbalance of circuit ratios by postponing robot tasks appropriately, i.e., increasing token holding times of robot task places in the TEG.

Case 1)

Since there are no delays in the critical circuits, \( PM_1 \) has no delays.

Case 2)

Again \( K \) becomes 1, so wafer delays of \( PM_1 \) are equalized.
Conclusion

- We have characterized the wafer delays in cluster tools for a given K-cyclic schedules.

- The closed form formulas describe how the delays occur. Also, it is known that the backward sequence is better than the swap in respect of wafer delays.

- We also have suggested two wafer delay regulation methods: feedback control and workload balancing.

- Our further works include to generalize this study for general series-parallel PMs.

- Analyses for worst-case delay are also included.

- We may improve wafer delay regulation methods with different and practical constraints.
Question?


Reference


