Motivation

- Cluster tools are an important class of tools in semiconductor wafer fabrication (including clustered photolithography scanners)
- Flow line models can be used if we ignore the wafer transport robot
- Existing results do not focus on transient performance, state-dependent failures and computationally tractable wafer-to-wafer behavior

System description: A flow line with regular process times

- Wafers must receive service from all modules in sequence to be complete
- Process times are deterministic (regular): $\tau_i$ in module $m_i$

Manufacturing blocking: A wafer advances as soon as its service is complete and the next module is vacant

Question: How do wafers advance? Where are they delayed?

Example flow line:

- Wafers must receive service from all modules in sequence to be complete
- Process times are deterministic (regular): $\tau_i$ in module $m_i$

Definition: Modules with process time $\tau_i > \tau_j$ for $i < j$ (greater process time than all preceding modules) are termed **successive bottlenecks**

Definition: The modules between (including) two successive bottlenecks form a **resettable monotone channel (RMC)**

Wafer progression in an RMC: Yellow indicates delay

Accomplishments

- For flow lines with regular service (process) times
  - Prove the manner in which wafers are delayed
  - Develop equations for setups and wafer lots
  - The new approach requires significantly less computation to characterize wafer lot progression with setups than traditional tools
  - Can be directly applied to the study of transient behavior in cluster tools

Theorem: Delay in a flow line

In RMC, with modules labeled as $m_1, ..., m_M$.
- Let $d(w)$ = delay wafer $w$ experiences in module $m_i$ (waiting for $m_{i-1}$)
- Let $Y(w) = d_i(w) + d_j(w) + ... + d_{i*}(w)$ if $w$ is a wafer delayed

If a wafer is delayed in an RMC, there is a module $m_j$ with $d_j(w) > 0$, $j < j^*$ (no delay before $m_j$)
- $d_j(w) = \tau_{j^*} - \tau_j > 0$ (after $m_j$, maximum possible delay is experienced)

$$Y(w + 1) = \max \left[ 0, \min \left( S_{j^*}, r_i(w) + \tau_{j^*} - \max \{\tau_i, \alpha_{j^*+1} - \alpha_i - d_{j^*}(w)\} \right) \right]$$

$$d_i(w + 1) = \max \left[ 0, \alpha_i + d_i(w) + \tau_i + d_{i+1}(w) - \alpha_i \right]$$

where $S_i := \sum \tau_j - \tau_i$ is the maximum in-channel delay in $m_0, ..., m_i$

Wafer delay in a flow line may be described by the delay experienced in each RMC formed by the successive bottlenecks

Application: Semiconductor wafer fabrication

Buffer modules are modeled as a module with $\tau_i = 0$

Redundant (parallel) modules

Setups: The tool may require a setup before it can process a wafer.
- Setup only begins after all modules, before a particular one, are empty
- This is a class of state dependent failure

Wafer lots: Wafers typically arrive as a group of identical wafers (a lot)

For these, significant reduction in computational complexity is possible

Future Directions

- Steady state throughput analysis with setups (which are a form of state dependent failure)
- Incorporate multiple products with different process (service) times
- Determine design principles/trade-offs
- Incorporate robots and transient behavior

Theorem: Computational Complexity

<table>
<thead>
<tr>
<th>Recursion</th>
<th>Max/Min Operations</th>
<th>Addition Operations</th>
<th>Multiplication Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic Evolution Equations</td>
<td>$O(M^2)$</td>
<td>$O(M)$</td>
<td>$O(1)$</td>
</tr>
<tr>
<td>Evolution of All Successive Bottlenecks</td>
<td>$O(2^M)$</td>
<td>$O(2^M)$</td>
<td>$O(1)$</td>
</tr>
<tr>
<td>Evolution of the Final Channel Alone with Batch Arrival</td>
<td>$O(2^M)$</td>
<td>$O(1)$</td>
<td>$O(1)$</td>
</tr>
</tbody>
</table>

Notation: There are $\alpha$ successive bottlenecks, W wafers to study. L wafers per lot, M modules

Contact Information
James R. Morrison, KAIST
E-mail: james.morrison@kaist.edu

KAIST Websites
- KAIST general: http://www.kaist.edu