Models of Clustered Photolithography Tools for Fab-Level Simulation: From Affine to Flow Line

Jung Yeon Park, Kyungsu Park, and James R. Morrison, *Member, IEEE*

***Abstract*—Fab-level discrete-event simulation is an important practical tool for the analysis and optimization of semiconductor wafer fabricators. In such facilities, a clustered photolithography tool (CPT) is by far the most expensive tool and often the capacity bottleneck. In this paper, we consider linear, affine, flow line, and detailed models of CPTs for use in fab-level simulation. We develop extensions to affine and flow line models and demonstrate exactly how to convert raw CPT data into the various models. Using a detailed CPT model based on industry data as the baseline, numerical experiments are conducted to test the models' fidelity for cycle time, lot residency time, and throughput. We also compare the computational burden of each model class. Further simulations are conducted to test the models' robustness to changing fab conditions, e.g., when lot size or train size changes. Flow line models are shown to be more accurate and robust than linear or affine models and require approximately 200 times less computation than detailed models.**

***Index Terms*—Fab-level simulation, clustered photolithography tools, affine models, flow line, throughput and cycle time models.**

I. Introduction

DUE TO their high cost, wafer fabrication facilities (fabs) must be well designed, operated efficiently and any changes to operating practices should be carefully considered. Fab-level simulation is an essential decision support technology to help pursue these objectives, see [1], [2], and has been used in many contexts. In particular, fab simulation with detailed AMHS models has been considered in [1] and [3]–[5]. Studies of fab behavior in relation to changes in the number of wafers per lot (lot size) were conducted in [6] and [7]. Efforts to reduce cycle time were pursued in [8] and [9]. There are many others that focus on wafer release policies, production control policies, batching, setups, product mix, etc.

Though fab-level simulation comprises many components, our focus is on equipment models for clustered photolithography tools (CPTs). These tools can cost as much as U.S. $120 million [10]. CPTs are typically the fabricator bottleneck and contribute significantly to fab cycle time.

Manuscript received December 2, 2015; revised January 28, 2017; accepted August 30, 2017. Date of publication September 15, 2017; date of current version October 27, 2017. This work was supported by KAIST High Risk High Return Project under Grant N10150064. *(Corresponding authors: Kyungsu Park; James R. Morrison.)*

The authors are with the Department of Industrial and Systems Engineering, KAIST, Daejeon 34141, South Korea (e-mail: jpark0@kaist.ac.kr; kyungsu@kaist.ac.kr; james.morrison@kaist.edu).

Digital Object Identifier 10.1109/TSM.2017.2752755

A. Equipment Models of CPTs

Numerous CPT models have been considered for fab simulation and optimization. We study the following.

- *Linear models* assume the per wafer production rate is constant (possibly a function of the wafer class).
- *Affine models* incorporate the so called first wafer delay into linear models. The first wafer production rate of each lot is considered separately from the other wafers.
- *Flow line models* include some details of the tool behavior but ignore wafer handling robots.
- *Detailed models* include process modules, wafer buffers, setups, wafer handling robots, and robot control policies.

Linear models have been used to study recipe dedication in CPTs in an ASIC fab model [11] and for litho machine scheduling [12]. Affine models are the basic equipment model provided in the industry standard fab simulation software *AutoSched AP* and have been used in optimization studies for CPTs in [13]. Flow line models of CPTs have been used for optimization, e.g., [14]–[17], simulation, e.g., [18] and [19] and analysis, e.g., [20] and [21]. Detailed models of CPTs have been used for wafer transport robot scheduling, e.g., [22] and [23].

There are other models that could be applied to the modeling of CPTs. Aggregation or lumped parameter models subsume unknown or random events into a few parameters, see [19], [24]–[27]. As our focus is on CPTs, whose internal workings are largely deterministic relative to the issues considered in the lumped parameter models, we will not consider such models here. There are numerous random events that occur internal to CPTs; we will model some of them. Note that we are focusing on equipment models of CPTs and their comparative performance, and not on tool scheduling.

B. Features of Equipment Models

In all models, there is a fundamental tradeoff between fidelity and complexity, see [1], [28]. Simulation models with greater detail, such as flow line models, may be more expressive and provide greater accuracy but require more modeling effort and longer computation times.

Obtaining and transforming the data needed to parameterize a model is not trivial. As the training data is reflective of

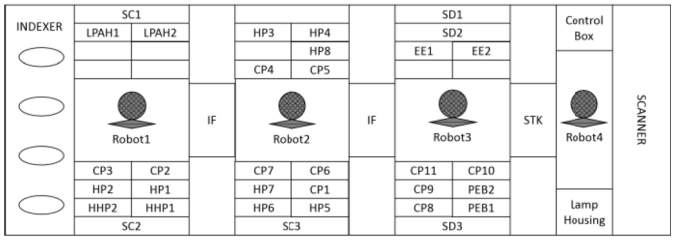

Fig. 1. Layout of a CPT.

a particular instance of wafers per lot and product mix, models can be inaccurate when used outside of the training conditions.

Equipment models should be accurate yet expressive, computationally tractable, and robust to changes in input data.

C. Contribution and Organization

In this paper, we investigate equipment models of clustered photolithography tools (CPTs) for use in fab-level simulation. We conduct simulations to compare their fidelity, computation times, and robustness. While the models considered in this paper are not new (we do extend them), to our knowledge, a detailed comparison of such models has not been conducted in the literature. For CPTs, we

- develop generalized affine models;
- propose a method to compute module processing times for flow line models when they are unknown;
- characterize exactly how to parameterize each model;
- conduct simulations on the fidelity, computation times, and robustness of the models; and
- compare the models.

We hope the results will be of use when selecting which CPT model to use for fab-level simulation and optimization.

The paper is organized as follows. In Section II, we describe our CPT system and our detailed baseline simulation model. Linear and affine models are discussed in Section III. Flow line models are discussed in Section IV. The numerical experiments are described in Section V. Results are provided in Section VI. Concluding remarks are provided in Section VII.

Some of these results appeared in conference form [29]. There are many new concepts, details and simulations here.

II. CPT System Description

The goal of a CPT is to transfer a pattern from a patterned mask (reticle) onto the surface of wafers. This is accomplished via three logical sections within the tool: the pre-scan processes, the scanner, and the post-scan processes. Here we describe the CPT model we will use for our studies.

A. Process Flow

Fig. 1 depicts a CPT based on actual data from the semiconductor industry [22]. Each of the four load ports can hold a lot of wafers, which enter and exit the CPT via the wafer indexer. This CPT consists of four clusters, each with its own single-armed wafer transport robot. There are several processes: hot plates (HP/HHP), low-pressure adhesions (LPAH), cold plates (CP), spin coaters (SC), post exposure bake hot plates (PEB), edge exposures (EE), and spin developers (SD). Some operations have multiple dedicated modules. In between the first three clusters are interface buffers (IF) which can hold at most one wafer. There is a pre-scan buffer (labeled STK for stacker) between the third cluster and the scanner, which can hold up to 16 wafers.

TABLE I
Process Flow for Three Classes of Wafers in Our CPT

	TARC #1		TARC #2		BARC	
	Process module	PT	Process module	PT	Process module	PT
Op 1	INDEXER		INDEXER		INDEXER	
Op 2	HHP1/HHP2	80	HHP1/HHP2	80	HHP1/HHP2	80
Op 3	LPAH1/LPAH2	90	LPAH1/LPAH2	90	LPAH1/LPAH2	90
Op 4	CP2/CP3	60	CP2/CP3	60	CP2/CP3	60
Op 5	SC1/SC2	65	SC1/SC2	65	SC3	50
Op 6	SC3	50	HP1/HP2	90	HP5/HP8	90
Op 7	HP5/HP6	90	CP6/CP7	60	CP1/CP6	60
Op 8	CP10/CP11	60	SC3	50	SC1/SC2	65
Op 9	Scanner	100	Scanner	100	HP6/HP7	90
Op 10	PEB1/PEB2	90	PEB1/PEB2	90	CP10/CP11	60
Op 11	CP8/CP9	60	CP8/CP9	60	Scanner	100
Op 12	EE1/EE2	90	EE1/EE2	90	PEB1/PEB2	90
Op 13	SD1/SD2/SD3	130	SD1/SD2/SD3	130	CP8/CP9	60
Op 14	HP3/HP4	90	HP3/HP4	90	EE1/EE2	90
Op 15	CP4/CP5	60	CP4/CP5	60	SD1/SD2/SD3	130
Op 16	INDEXER		INDEXER		HP3/HP4	90
Op 17					CP4/CP5	60
Op 18					INDEXER	

The process flows for three different classes of wafers are provided in Table I as TARC #1, TARC #2, and BARC. TARC and BARC stand for top anti-reflective coating and bottom anti-reflective coating, respectively. Wafers proceed from one operation in their flow to the next using the wafer transport robot, which has a pick/place time of one second and move time of three seconds. At each operation, wafers may be served by any one of the listed process modules with the respective process time (PT). The buffers IF and STK are used as required. The indexer is modeled as a single module process with zero process time. Inside the tool, the robot actions are dictated by the longest waiting pair (LWP) policy; see [16], [23]. This robot scheduling policy achieves optimal steady-state throughput for the process flows.

We use a previously constructed detailed discrete-event simulation model (DS) of this CPT configuration from [16]. This model contains all of the details of the CPT that we have described above, such as the redundancy of process modules, recipe data, robot move times, and the robot policy.

B. Lot Description

As many as 25 wafers are grouped into batches called lots. Each lot consists of wafers of the same class (this can be easily generalized). Lots enter the tool in a FIFO manner (it is easy to reorder them in the queue to model different dispatch policies). Wafers are admitted to the tool as soon as the tool is ready. All process modules for each operation may serve only one wafer class at a time. This prevents overtaking and contamination. The STK may hold several wafer classes.

Setups may be required between different lots. There are many possible types of setups, including full track, reticle alignment, pre-scan track, post-scan track, rolling setups and

so on. We focus on two cases: 1) reticle alignment setups only and 2) separate pre-scan track and reticle alignment setups, which will be described subsequently. We ignore post-scan track setups as the scanner setup times are often longer than the post-scan setup time (and post-scan setups may not be required due to similarities in post-scan processes).

A reticle alignment at the scanner may be required for the first wafer of a lot. This ensures that the pattern is properly aligned. While it may be conducted when changing lot class, reticle alignment may also be conducted for every lot to ensure quality. Based on industry data, this setup is assumed to be uniformly distributed in the range [210, 260] for every lot.

The pre-scan track setup is conducted only when the next lot is of a different class. The lot of a new class, upon arrival to the tool, must wait until all pre-scan processes are empty. The pre-scan track setup then commences. Once it is complete, the first wafer of the new lot enters the first process. This setup is assumed to be uniformly distributed in the range [240, 420].

C. Metrics

Our primary metrics of interest are lot cycle time (CT), lot residency time (LRT), lot throughput time (TT), and computation time. For lot i, let CT_i, LRT_i, and TT_i denote the first three, respectively. We define a_i, S_i, and C_i as the time instants at which lot i arrives at the tool queue, starts processing on the tool, and completes processing on the tool, respectively. For lots 1, 2,. . . , L, define

$$\begin{aligned} CT_i &= C_i - a_i \\ LRT_i &= C_i - S_i \\ TT_i &= min(C_i - S_i, C_i - C_{i-1}) \end{aligned}$$

with the initial condition $C_0 = -\infty$. TT_i is the time between two consecutive lot exits from the tool, not including idle time. Computation time is the time needed to calculate the start and completion times (not including model parameter extraction).

D. Comments on Detailed Simulation Model

We assume that the detailed model (DS) is exact and will be used as the benchmark for all of our simulations. Hereafter, we refer to the data obtained from this detailed model (DS) as our "true data" and call this the "true system".

To properly validate the DS model, it should be compared to industry data from a production CPT. However, it is difficult to gain access to data containing detailed aspects of the CPT configuration, including module process times, setup times, recipes, robot move times, and so on. Further, such data would represent a single realization of the CPT parameters. To our knowledge, we do not know anywhere in the literature that provides such detailed industry data. Therefore, we use a detailed model constructed using information from [22], which in turn is based on data from the industry.

With that said, we were able to obtain tool log data for CPT operation from an industry partner. Tool log data is simply the advancement of wafers from process to process and does not include recipe, robot motion, process times and so forth. Note that this data is reflective of only one set of operating conditions. As a sanity check, we use this tool log data to assess the relative performance of the models used in this paper; the results are shown in Section VI. In comparison to true industry data, the models behave similarly as they do in our simulations. These results give us some comfort that the detailed simulation may serve as a baseline.

We do not directly include tool availability in our study. All of the models considered can readily include the two common tool failure models used in fab-level simulation: non-preemptive and preemptive. Non-preemptive tool down events are often used to model preventive maintenance (PM) events. In this case, a high priority customer is used to model the PM event. Preemptive tool down events are used to model unanticipated failures. In this case, the event is modeled by a complete cessation of production on that tool until the event is complete. The general effects of both can be predicted in our studies by increasing the system loading.

Note that we are modeling a single CPT and comparing the equipment models for this single tool. For a simulator of a fab, a fleet of such models would be required for a tool group of CPTs. In that context, the lots may be dedicated to specific CPTs for process quality and yield prediction purposes.

III. Linear and Affine Models

Linear models (LM) and affine models (AF) are intuitively simple and widely used for fab simulation. We use $\tilde{S}_i$, $\tilde{C}_i$ with a tilde as the start and completion times for lot i obtained from the model (as opposed to the values obtained from the true system). Models are first parameterized and then used for simulation.

LM and AF were not initially intended for tools that allow multiple lots to receive processing simultaneously in a tool. However, some studies have extended the basic models to determine the total production time for a batch of lots (equivalent to the sum of TT for lots processed consecutively). In [6], the overlap (which we refer to as parallelism) is explored. They developed a model allowing for constant or variable overlap between lots in a single cluster tool for total production time of a batch. Multi-cluster tools are not considered, nor do they explicitly study LRT for randomly arriving lots. Here, we use LM and AF models without overlap, explicitly give expressions for CT, LRT and TT, and detail exactly how to parameterize them. However, these models are inherently handicapped by their inability to properly address parallel processing of lots.

A. Notation

Before describing the equipment models, we provide the key notation for the true data and abbreviations in Table II. Most are self-explanatory; an explanation may help for $\Omega(i, w)$, the overall wafer index of the w-th wafer in lot i. For example, if every lot has 25 wafers, the 7^{th} wafer in lot 10 has $\Omega(10, 7) = 257$; it is the 257^{th} wafer processed on the tool.

B. Description and Parameterization

The parameters for the models are extracted from the start and completion times of lots (LM) or wafers (AF) obtained

TABLE II
LIST OF NOTATION AND ABBREVIATIONS

Notation	Definition
L	Total number of lots
W_i	Number of wafers in lot i (lot size)
k, k'	Indices for lot classes, $k, k' \in \{1, \ldots, K\}$
k_i	Class of lot i
$k(w)$	Class of wafer w
$L(k)$	Set of lot indices for all lots in class k, $L(k) = \{i \mid k_i = k\}$
$L(k, k')$	Set of lot indices for all pairs of lots where $L(k, k') = \{i \mid k_i = k, k_{i-1} = k'\}$
$\Omega(i, w)$	Overall wafer index of w-th wafer in lot i
B	Index of bottleneck process (scanner)
$R(w, m)$	Number of redundant modules for process m for wafer w
a_i	True arrival time instant of lot i
$X_{w,m}$	True entry time of wafer w into process m
S_i	True start time instant of lot i
C_i	True completion time instant of lot i
$B_{\Omega(i,w)}$	True start time instant of w-th wafer in lot i (instant that wafer begins processing at the first process)
$F_{\Omega(i,w)}$	True completion time instant of w-th wafer in lot i (instant that wafer finishes processing at the last process)
DS	Detailed CPT model
LM	Linear models
AF	Affine models
PFL	Parametric flow line models
EFL	Empirical flow line models

from the true data. Table III "Parameters" row provides the equations for parameter extraction. Our method uses the industry standard approach (averaging over all available data) which is preferred as it preserves mean TT values when the model well matches the system being studied.

The class of lot i is defined as $k_i \in \{1, \ldots, K\}$ where K is the number of lot classes. The number of wafers in lot i is W_i. LM calculates $\tilde{C}_i$ as a function of W_i. The parameter A^{k_1} denotes the per wafer throughput time for wafers of class k_1 (it does not include idle time between wafers similar to TT). $L(k_1)$ denotes the set of lot indices for all lots in class k_1.

AF extends LM to allow a first wafer delay. These $Ax + B$ models, are more expressive and accurate [30] than LM.

AF can be extended to incorporate setups by allowing the parameters A and B to depend on lot class. The parameter A^{k_1} is the wafer throughput time within a lot for class k_1 lots. The parameter B can be interpreted as the first wafer delay and may depend on the current lot class (k_1) and previous lot class (k_2). This can help to model reticle setups or pre-scan track setups. In [29], this generalized B^{k_1,k_2} was the most accurate of the affine models considered; we use it.

For AF, $L(k_1,\ k_2)$ is the set indices of lots of class k_1 whose predecessor on the same tool was a lot of class k_2. Similarly to C_i for lot completion times, define $F_{\Omega(i,w)}$ as the wafer completion time for the w-th wafer in lot i.

C. Simulation

Given the parameter values obtained as detailed in Table III and the arrival time a_i, wafers W_i, and class k_i for each lot i, the models estimate the start $\tilde{S}_i$ and completion times $\tilde{C}_i$ as in Table III, with $\widetilde{C}_0 = -\infty$. From these, the simulated $\widetilde{CT}_i$, $\widetilde{LRT}_i$, and $\widetilde{TT}_i$ are calculated for each lot i.

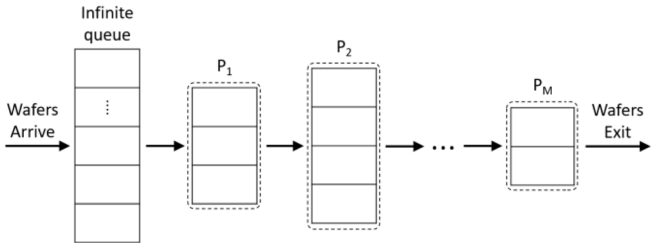

Fig. 2. Flow line.

IV. FLOW LINE MODELS

A. Overview

Flow lines (FL), see [31], consist of a series of M processes $P_1, \ldots, P_M$ at which wafers must receive service sequentially. Similarly to k_i for lot class, let $k_w \in \{1, \ldots, K\}$ denote the class of wafer w (all wafers within a lot have the same $k(w)$). There are $R(w,m)$ identical parallel servers dedicated to process m for a wafer w with class $k(w)$. Wafers may receive service from any of these identical servers. Each server can serve at most one wafer at a time. Wafers move to the next process as soon as it is available. There is an infinite buffer prior to the first process. Wafers are admitted in a first come first served (FCFS) manner. As discussed in [18] and [32]–[34], intermediate buffers IF and STK are modeled as parallel servers with zero process time. See Fig. 2.

A wafer w of class k arrives to the flow line at time a_w with $a_w \leq a_{w+1}$, similarly to a_i for lot arrival times (all wafers within a lot have the same a_w). Each server of process P_m has a deterministic processing time τ_m^k. After service is completed at a process, the wafer either moves to the next process or waits at its current position until the next process is available. After receiving service from all M processes, the wafer exits the flow line.

The fundamental assumption behind flow line models for multi-cluster tools is that the tool is process-bound. Cluster tools, especially CPTs, are often process-bound, see [35], [36].

We consider two types: a parametric flow line model (PFL) and an empirical flow line model (EFL). PFL assumes that the processing times τ_m^k are known. EFL is parameterized using true system wafer advancement data. The progression of wafers in PFL and EFL is characterized by the elementary evolution equations (EEEs), see [18], [34], which we extend to address features of our CPT system. The EEEs are recursive equations to calculate the entry times of the wafers into each process, which can then be used to compute the start and exit times of wafers. See Table IV.

Each class has its own process flow in our CPT; the number of processes for each class may be different. We add dummy modules with zero process times in front of the first process so that the number of processes for each class is equal. This is

TABLE III
LINEAR AND AFFINE MODEL EQUATIONS

	Linear Model (LM)	Affine Model (AF)
Lot Indices	$L(k_1) = \{i \mid k_i = k_1\}$	$L(k_1, k_2) = \{i \mid k_i = k_1, k_{i-1} = k_2\}$
Parameters	$A^{k_1} = \frac{\sum_{i \in L(k_1)} (C_i - max(a_i, C_{i-1}))}{\sum_{i \in L(k_1)} W_i}$	$A^{k_1} = \frac{\sum_{i \in L(k_1)} (C_i - F_{\Omega(i,1)})}{\sum_{i \in L(k_1)} (W_i - 1)}$
		$B^{k_1,k_2} = \frac{1}{\lvert L(k_1,k_2) \rvert} \sum_{i \in L(k_1,k_2)} \left(F_{\Omega(i,1)} - \max(a_i, C_{i-1}) \right)$
Start	$\tilde{S}_i = max(a_i, \tilde{C}_{i-1})$	
Completion	$\tilde{C}_i = \tilde{S}_i + A^{k_i} \times W_i$	$\tilde{C}_i = \tilde{S}_i + A^{k_i} \times (W_i - 1) + B^{k_i, k_{i-1}}$

TABLE IV
PARAMETRIC AND EMPIRICAL FLOW LINE MODEL EQUATIONS

	Parametric Flow Line (PFL)	Empirical Flow Line (EFL)
Modified Redundancy	$R'(w,m) = \begin{cases} R(w,m), & k(w) = k(w-1) \text{ or } m \in MBC(k(w)) \\ 1, & otherwise \end{cases}$	
Modified or Calculated Process Times	$S(k,m) = \begin{cases} \tau_{PB}^k + 3\delta + 4\varepsilon, & m = PB \\ \tau_B^k + 2\delta + 4\varepsilon, & m = B \\ \tau_m^k + \delta + 2\varepsilon, & otherwise \end{cases}$	$S(k,m) = \begin{cases} \frac{\sum_{i \in L(k)} \sum_{w=2}^{W_i} (X_{w,B+1} - X_{w-1,B+1})}{\sum_{i \in L(k)} (W_i - 1)}, & m = B \\ \min_{\{w \mid k(w)=k\}} (F_w - X_{w,M}), & m = M \\ \min_{\{w \mid k(w)=k\}} (X_{w,m+1} - X_{w,m}), & otherwise \end{cases}$
EEEs	$\tilde{X}_{w,1} = max\{a_w, \tilde{X}_{w-R'(w,1),P(w)+1}, \tilde{X}_{w-1,1}\} + \tau_s'(w,m)$ $\tilde{X}_{w,m} = max\{\tilde{X}_{w,m-1} + S(k(w), m-1) + \tau_R'(w,m), \tilde{X}_{w-R'(w,m),m+1}, \tilde{X}_{w-1,m}\}$ $\tilde{X}_{w,M} = max\{\tilde{X}_{w,M-1} + S(k(w), M-1), \tilde{X}_{w-R'(w,M),M} + S(k(w), M), \tilde{X}_{w-1,M}\}$	
Start	$\tilde{S}_i = \tilde{X}_{\Omega(i,1),d(k_i)+1}$	
Completion	$\tilde{C}_i = \tilde{X}_{\Omega(i,W_i),M} + S(k_i, M)$	

done to simplify the EEEs. Let *d(k)* be the number of dummy modules added to the process flow of class k wafers.

In our system, a process may only serve one class of wafer at a time, regardless of its redundancy. However, this restriction does not apply to the buffers. Let *MBC(k)* be the set of process indices corresponding to the buffer stages for process flow k. Depending on *MBC(k)* or a change in wafer class, we define *R'(w,m)* equal to either the redundancy of process m or 1, to model this single class processing restriction. See Table IV.

B. Setups

Our CPT conducts a reticle alignment setup of duration τ_R for the first wafer of every lot. We model it as an extension of this first wafer's process time in the scanner. We define

$$\tau_R'(w,m) = \begin{cases} \tau_R, & w = \Omega(i,1) \text{ and } m = B+1 \\ 0, & otherwise \end{cases}$$

This is used to calculate the entry time into process $B+1$.

Our CPT can also conduct a pre-scan track setup of duration τ_s for the first wafer of a lot when lot class changes, depending on which setup case is used. Let

$$\tau_s'(w,m) = \begin{cases} \tau_s, & k(w) \neq k(w-1) \\ 0, & otherwise \end{cases}$$

Note that the condition $k(w) \neq k(w-1)$ ensures that wafer w is the first wafer of a new lot. Let P be the last pre-scan process and set *P(w)* equal P if $k(w) \neq k(w-1)$, and 1 otherwise. When a pre-scan track setup is performed, the wafer can only enter the first process once the previous wafer has exited the last pre-scan process.

If there are no pre-scan track setups, then $\tau_s'(w,m) = 0$ and $P(w) = 1$ for all wafers w and process m.

Throughout, when we simulate with an FL model, the setup durations for each lot are provided as input to that model.

C. Parametric Model

Although the processing times (PT) are given in a PFL, they must be modified to account for the robot overhead. We provide a method to incorporate these essential overheads.

For maximum throughput, a robot must supply a wafer to the scanner via the pre-scan buffer as soon as possible. The rate at which the pre-scan buffer is fed is determined by the penultimate bottleneck process (that is, the slowest process before the bottleneck). Denote the bottleneck and penultimate bottleneck process indices as B and PB, respectively. When a wafer has completed service at P_{PB}, the robot then picks the wafer, moves to P_{PB+1}, places the wafer, moves to P_{PB-1}, picks the next wafer, moves to P_{PB} and places it into P_{PB}. This is the minimum robotic workload. It maximizes the throughput of the tool and consists of three moves and four pick/places. At the bottleneck process (scanner), there is a dedicated robot and there is one less move time. The robot move time and pick/place time are denoted as δ and ε, respectively. We use 3 s and 1 s, respectively.

For the other processes, when a wafer completes service, the robot must pick the wafer from its location, move to the next

TABLE V
PROCESS TIMES FOR PARAMETRIC FLOW LINE

Process #	TARC #1			TARC #2			BARC		
	Process	R	PT	Process	R	PT	Process	R	PT
1	Dummy	1	0	Dummy	1	0	Op 1	1	5
2	Dummy	1	0	Dummy	1	0	Op 2	2	85
3	Dummy	1	0	Dummy	1	0	Op 3	2	95
4	Dummy	1	0	Dummy	1	0	Op 4	2	65
5	Op 1	1	5	Op 1	1	5	IF	1	5
6	Op 2	2	85	Op 2	2	85	Op 5	1	63
7	Op 3	2	95	Op 3	2	95	Op 6	2	95
8	Op 4	2	65	Op 4	2	65	Op 7	2	65
9	Op 5	2	70	Op 5	2	70	IF	1	5
10	IF	1	5	Op 6	2	95	Op 8	2	70
11	Op 6	1	63	IF	1	5	IF	1	5
12	Op 7	2	95	Op 7	2	65	Op 9	2	95
13	IF	1	5	Op 8	1	63	IF	1	5
14	Op 8	2	65	IF	1	5	Op 10	2	65
15	STK	15	5	STK	15	5	STK	15	5
16	Op 9	1	110	Op 9	1	110	Op 11	1	110
17	STK	1	5	STK	1	5	STK	1	5
18	Op 10	2	95	Op 10	2	95	Op 12	2	95
19	Op 11	2	65	Op 11	2	65	Op 13	2	65
20	Op 12	2	95	Op 12	2	95	Op 14	2	95
21	Op 13	3	135	Op 13	3	135	Op 15	3	135
22	IF	1	5	IF	1	5	IF	1	5
23	Op 14	2	95	Op 14	2	95	Op 16	2	95
24	Op 15	2	65	Op 15	2	65	Op 17	2	65
25	IF	1	5	IF	1	5	IF	1	5
26	Op 16	1	0	Op 16	1	0	Op 18	1	0

R is the number of redundant modules.

process, and place the wafer into the module. This is the minimum robotic workload to transfer a wafer into the next process (one move and two pick/places). This overhead is added to the process times of the other processes, the buffers, and the front indexer. Dummy modules are kept at their original zero process times. Table V shows the modified process times after the robot overhead has been incorporated into the PFL.

Note that PFL are parameterized only by their given process time parameters. They do not otherwise require training data.

D. Empirical Model

EFL can be used when the module process times are unknown and the tool log data is given. The difference between the start times of wafers in consecutive processes contains process time, robot time and delays within a module. Processing times are calculated as the minimum possible such difference for all wafers of class k. (One could also use a 10^{th} percentile or similar.) We define $X_{w,m}$ as the true entry time of the w-th wafer into process m. To calculate the last module's processing time, F_w is used instead of $X_{w,m+1}$.

The bottleneck process time is treated differently. It is set as the average wafer throughput time from the bottleneck for each class k over wafers not facing a reticle setup. Expressions for these estimated process times *S(k,m)* are provided in Table IV.

Note that EFL may have large errors at high loading levels. Wafers may always be delayed in certain modules within the training data. Thus, the correct processing times cannot be calculated. This can be avoided if each lot class occasionally is processed on a nearly empty tool (so that the minimum occupancy times for a wafer are near to the process time plus robotic overhead). Such a problem was not observed in our simulations, however.

E. Elementary Evolution Equations

In the true CPT system, each of the two IF buffers can hold at most one wafer. These buffers are shared between the pre-scan and post-scan processes. This restriction is relaxed for both flow line models. We assume that there is one slot for the inlet buffer (pre-scan track) and one slot for the outlet buffer (post-scan track). For the stacker (STK) which has a total of 16 buffer slots, we allocate 15 slots for the inlet buffer and one for the outlet buffer. Let $\tilde{X}_{w,m}$ be the estimated entry time of the w-th wafer into process m. With the above proposed modifications and allowing for setups, the EEEs are used to calculate $\tilde{X}_{w,m}$ for $w = 1, \ldots, \Omega(L, W_L)$. The last term within the max function of the EEEs in Table IV is used to prevent overtaking. The initial conditions are $\tilde{X}_{w,m} = -\infty$ for $w < 1$ and $m = 1, \ldots, M$. The start and completion times of the lots are then calculated accordingly.

V. DESIGN OF SIMULATION EXPERIMENTS

A. Simulation Overview

The detailed CPT model (DS) is assumed to be exact and will be used as the baseline for our simulation studies. The results of the various models will be compared to the DS on CT, LRT, TT, and computation time. Simulations are run in JAVA on a computer with an i5-3570 CPU and 16GB of RAM.

We use a Poisson arrival process (but any arrival process can be used). The class of the next lot is determined based on the specified train size T, which is the average number of lots of the same class to be processed in a row. The class of each lot is determined by a simple Markov chain, whose state transition probability matrix is

$$P = \begin{bmatrix} \frac{T-1}{T} & \frac{1}{2T} & \frac{1}{2T} \\ \frac{1}{2T} & \frac{T-1}{T} & \frac{1}{2T} \\ \frac{1}{2T} & \frac{1}{2T} & \frac{T-1}{T} \end{bmatrix}$$

where T is the average train size.

We consider two types of setups: 1) reticle alignment only, and 2) pre-scan track + reticle alignment setups, which were detailed previously. We simulate with 15,000 lots, which gives over a year's run on the CPT and conduct 30 independent replications for each simulation case. The tool is initially empty. There are no tool failures. For all data collection, including parameter extractions, we discard the first and last 10% of lots.

We determine the lot arrival rate for a given simulation case by first finding the CPT's JIT (just in time) lot throughput. The detailed model is run five times with $a_i = 0$ for all lots. We find the average JIT lot throughput. We set the interarrival times of lots based on a predetermined loading level (with the JIT throughput set as the 100% loading case).

For each replication, the detailed model is run first to generate initial input data. This data is used to parameterize the equipment models. The detailed model is then run again and these results are used for comparison against the rest of the

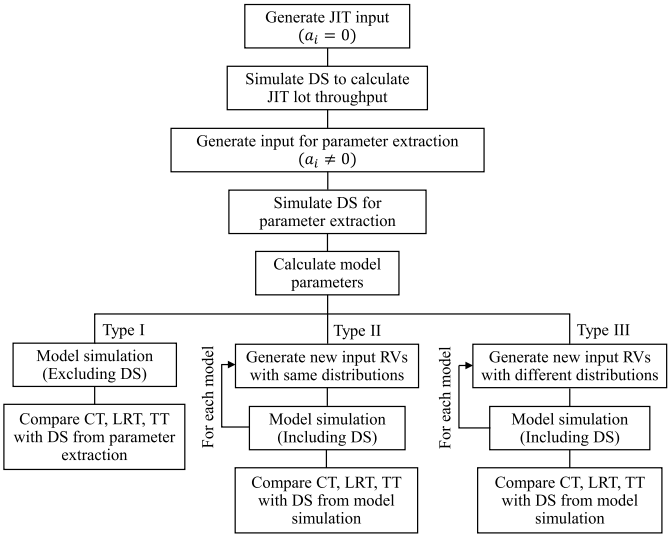

Fig. 3. Simulation process flow diagram.

models. These two phases are called parameter extraction and model simulation. The computation times do not include the time used for parameter extraction.

For model simulation, the start and completion times of all lots are generated and the average $\widetilde{CT}$, $\widetilde{LRT}$, and $\widetilde{TT}$ are found.

B. Simulation Types

We consider three types of simulations. See Fig. 3.

- *Type I simulations:* The exact same data sample used for parameter extraction is used for simulation.
- *Type II simulations:* The model parameters are extracted from one input sample and then we simulate using different realizations of the random input data (e.g., a_i) generated from the same distributions.
- *Type III simulations:* Model parameters are "trained" on one sample and then the underlying distributions for the random variables or some fundamental input parameter are changed. The models are then simulated using a random lot sample generated from these different conditions. The parameters changed are external to the tool.

Type I simulations allow us to assess the fidelity of the various models under the best conditions. Type II simulations are useful when trying to make predictions for the tool performance at its current state of operations. Type III simulations consider changes in operating conditions and address robustness.

We use Type I simulations at baseline conditions to assess computation times of the various models.

In Type I and II simulations, seven parameters are varied: train size, lot size, loading level, pre-scan buffer (STK) size, penultimate bottleneck process time, pre-scan track module process times, and different process time profiles. These are important parameters for CPTs.

In Type III simulations, we vary train size, lot size, and loading. We also vary the train size and lot size simultaneously. Reduced lot sizes are anticipated in the future; see [37].

TABLE VI
MODEL COMPUTATION TIMES

Model	Computation Time (ms)	Scaled Computation Time
DS	17372368	121273
LM	143	1
AF	177	1.24
PFL	83183	581
EFL	83596	584

Computation times have been scaled so that LM has a value of one in the third column.

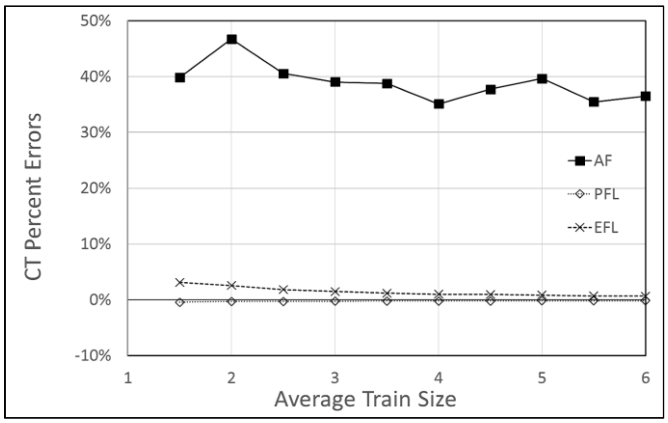

Fig. 4. Percent errors in mean cycle time for different train sizes (reticle alignment setup only).

VI. RESULTS AND DISCUSSION

We use average train size $T = 3$, lot sizes of 23, 24, 25 wafers per lot (with probability 0.25, 0.5, 0.25, respectively), and 0.95 loading as our *baseline case* parameters. These will be varied individually (mostly) in our studies. The setup types and IID setup durations are as detailed in the prequel.

A. Comparison of Computation Times

Table VI lists the average computation times in milliseconds for the various models using the baseline case parameters in Type I simulations. The results are very similar for both setup types. Both FL are very accurate and quite robust with approximately 500 times greater computation than LM and AF. The FL models are approximately 200 times faster than DS.

B. Type I Simulations

1) Train Size Cases: The average train size was varied from 1.5 to 6 in steps of 0.5. Figs. 4 to 6 provide the percent errors in average $\widetilde{CT}$, $\widetilde{LRT}$, and $\widetilde{TT}$ relative to the true data averages for the reticle alignment only setup cases. The results of the second setup case of reticle alignment + pre-scan track setups are shown in Figs. 7 to 9. LM is not shown as it appears overlapped with AF; it has slightly worse errors than AF.

For the reticle alignment only case, it can be seen that AF (and LM) is inaccurate for mean CT and LRT, with percent errors of approximately 37% and 55%, respectively. This is because these models do not allow parallelism (two or more

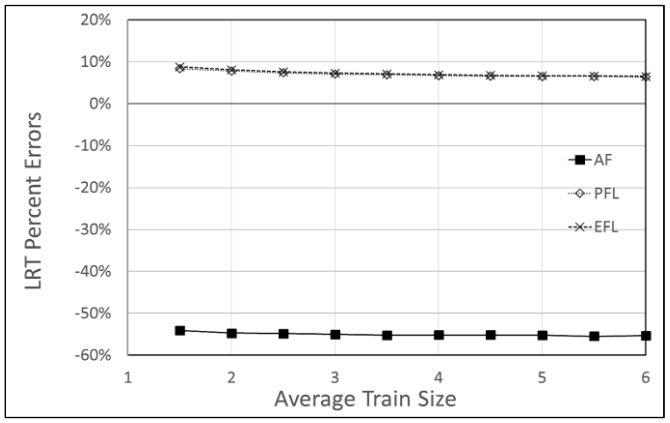

Fig. 5. Percent errors in mean lot residency time for different train sizes (reticle alignment setup only).

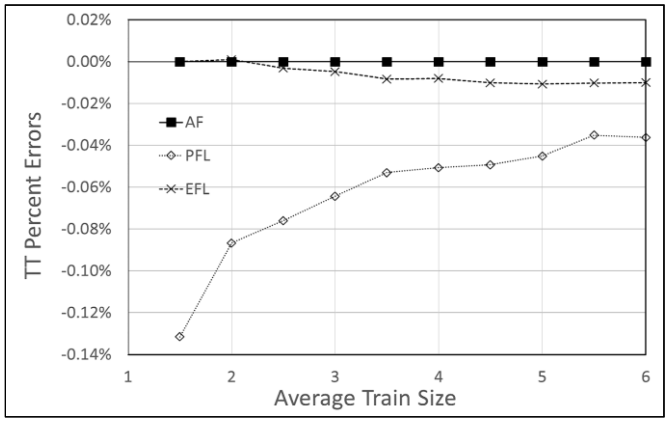

Fig. 6. Percent errors in mean throughput time for different train sizes (reticle alignment setup only).

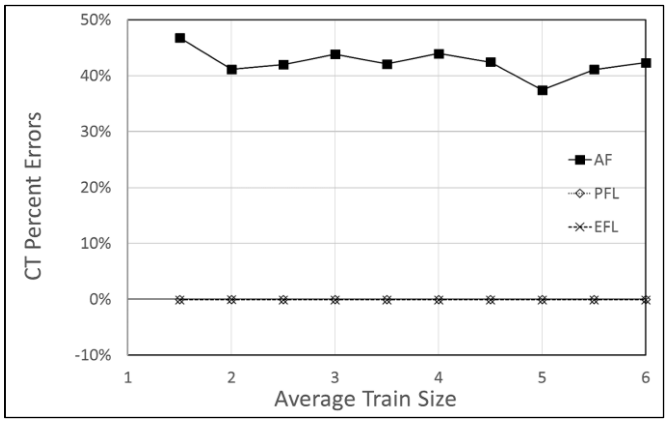

Fig. 7. Percent errors in mean cycle time for different train sizes (reticle alignment + pre-scan track setups).

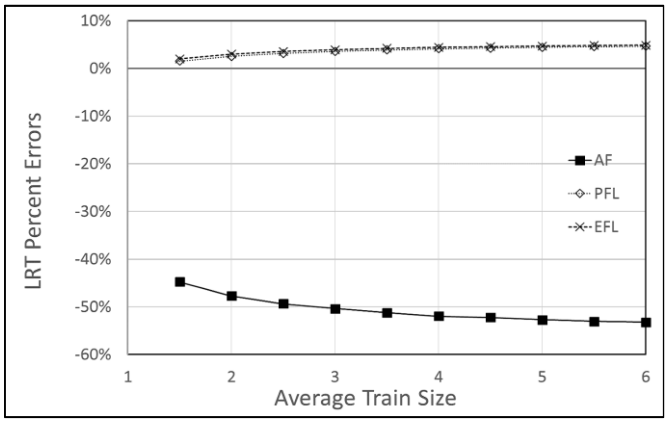

Fig. 8. Percent errors in mean lot residency time for different train sizes (reticle alignment + pre-scan track setups).

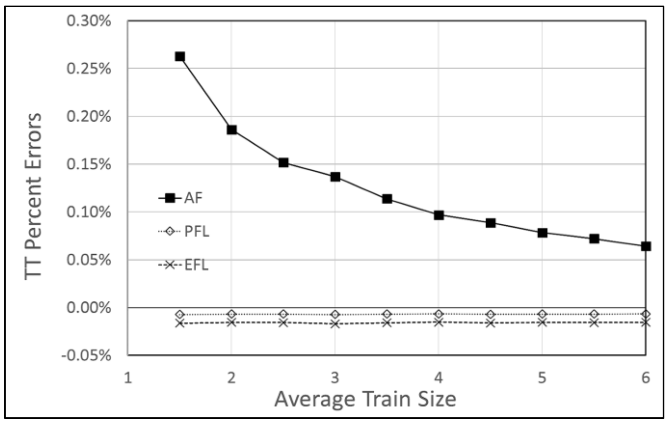

Fig. 9. Percent errors in mean throughput time for different train sizes (reticle alignment + pre-scan track setups).

lots in process on the tool simultaneously). The flow line models are more accurate, with errors up to 3% and 8% for mean CT and LRT, respectively. PFL is slightly more accurate than EFL for CT. For mean TT, AF and LM both have errors of practically zero. This is to be expected as Type I simulations use the same data sample for both training and simulation and the models are trained for throughput. This behavior continues for all Type I simulations for the reticle alignment only cases. More importantly, PFL and EFL perform extremely well for mean TT, with errors less than 0.14%.

For the second setup case, LM and AF continue to have high CT and LRT errors, while the FL models perform well on all three metrics; see Figs. 7 to 9. Throughout all simulation sets conducted in this paper, the second setup case shows similar results as the first setup case, for all Types I, II, and III. We omit the results of the second setup case for brevity; they show similar behavior as the first setup case where the FL models surpass LM and AF on all metrics.

2) Lot Size Cases: The lot sizes were varied from {1, 2, 3} to {23, 24, 25} in increments of 2 (with probabilities unchanged over the three values). Figs. 10 to 12 provide the percent errors in average CT, LRT, and TT of the various models, respectively. LM appears overlapped with AF; it has slightly worse errors than AF.

LM and AF seem to be very sensitive to lot sizes; they have very high errors of up to 280% for CT and 90% for LRT. As LM and AF are trained on throughput, they have zero errors for TT. PFL is the most accurate with errors of approximately 3% for CT and 10% for LRT. EFL fares slightly worse than PFL.

Both models experience a slight degradation in accuracy at very small lot sizes.

TABLE VII
SECONDARY METRICS FOR TYPE I – BASELINE LOT CONDITIONS

Model	CT				LRT				TT			
	Average	σ	|Error|	Error σ	Average	σ	|Error|	Error σ	Average	σ	|Error|	Error σ
DS	31005.10	25696.26			6505.59	703.30			2925.96	255.79		
LM	43114.53	36248.46	12486.36	15874.57	2925.96	86.29	3579.63	699.17	2925.96	86.29	99.52	243.93
AF	43103.04	36238.78	12475.13	15856.02	2925.96	78.45	3579.63	698.91	2925.96	78.45	99.32	243.79
PFL	30918.99	25685.66	88.23	154.80	6964.54	880.08	481.21	363.99	2924.07	240.39	23.56	48.99
EFL	31460.03	26136.73	521.89	753.64	6979.88	876.39	493.26	367.61	2925.82	237.23	25.80	54.26

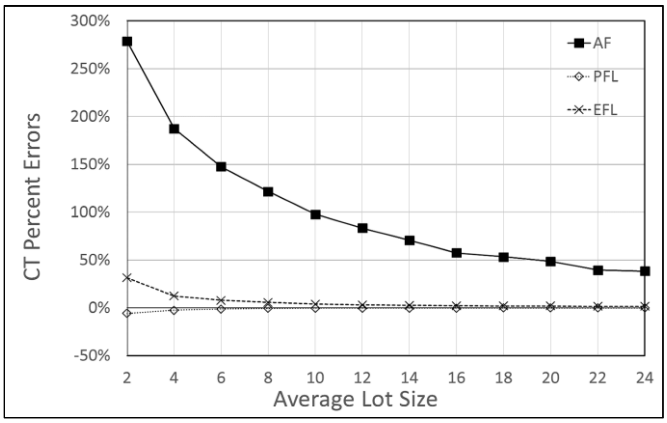

Fig. 10. Percent errors in mean cycle time for different lot sizes (reticle alignment setup only).

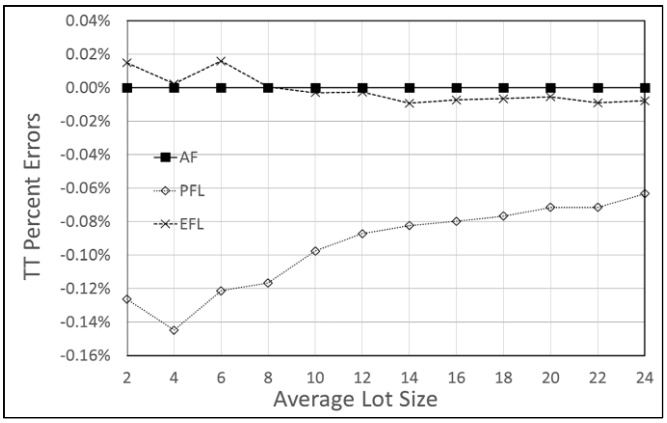

Fig. 12. Percent errors in mean throughput time for different lot sizes (reticle alignment setup only).

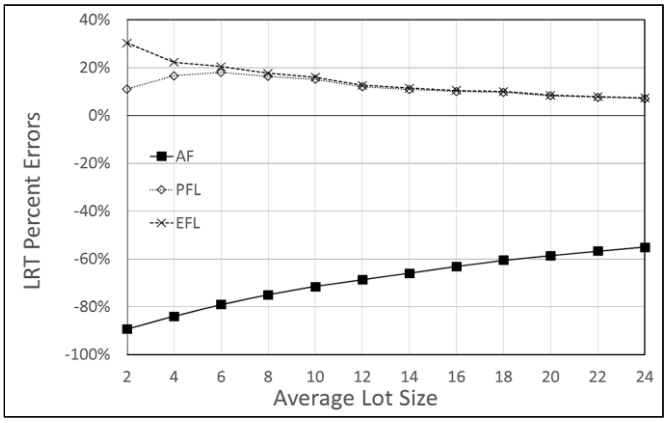

Fig. 11. Percent errors in mean lot residency time for different lot sizes (reticle alignment setup only).

3) Type I Simulations for Other Parameters: Similar results are obtained for other parameters: loading level, STK capacity, penultimate bottleneck process time, pre-scan track module process times, and different process time profiles. The loading level was varied from 0.90 to 0.99 and values of 0.80 and 0.85 were also simulated. The STK size (pre-scan buffer) was varied from 2 to 20. The penultimate bottleneck process time and pre-scan track module process times were scaled from 40% to 180% of their original values in increments of 20%. For the last parameter, three different process flows were simulated. The results paint a similar picture, where the FL models often dominate LM and AF.

4) Type I Simulation Performance (Average vs. Lot by Lot): Thus far, we have compared the *average* values. We now consider the standard deviation, the mean absolute error, and the error's standard deviation. Table VII shows these secondary metrics for our baseline case. One can see that the flow line models have very low mean absolute errors and error standard deviations. The standard deviation of each of the metrics are also close to that of the detailed model. For the linear and affine models, even for TT (for which they were trained on), the absolute error and standard deviation is relatively high. This behavior continues for all parameter sets considered.

C. Type II Simulations

Type II simulations study model performance when the simulation and training data use different realizations of the random variables (e.g., a_i) with all underlying parameters held constant. The results are similar to the Type I simulations with slightly higher errors and higher variability. For brevity, we do not provide the details.

D. Type III Simulations

In Type III simulations, the models are first trained on data obtained from one set of input parameters and then simulated with a different set of parameters. These allow us to assess the robustness of the models to changing conditions. For train size, the models were trained with train sizes of 1.5, 3, and 6 and simulated with varying train sizes from 1.5 to 6. For lot size, the models were trained with lot sizes {3, 4, 5}, {13, 14, 15}, {23, 24, 25} and simulated against varying lot sizes. In the third case, the models were trained on loading levels of 0.85, 0.95, and 0.99 and simulated at different loading levels. Finally, the models were trained on a train size of

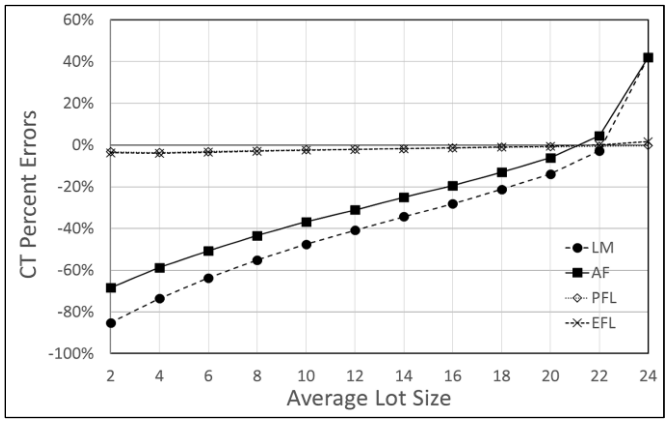

Fig. 13. Percent errors of mean cycle time for different lot sizes. Models are parameterized with lot sizes {23, 24, 25}.

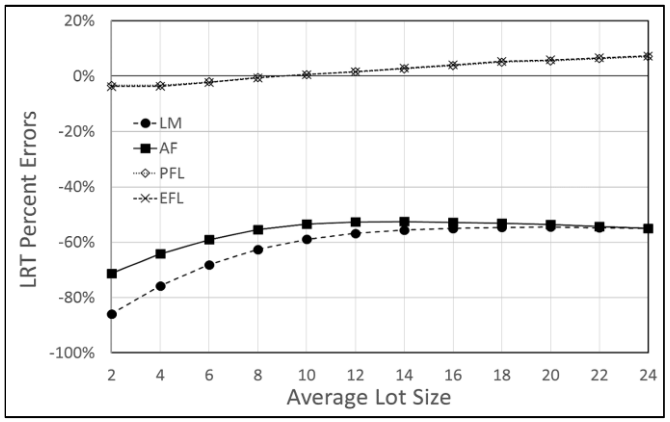

Fig. 14. Percent errors in mean lot residency time for different lot sizes. Models are parameterized with lot sizes {23, 24, 25}.

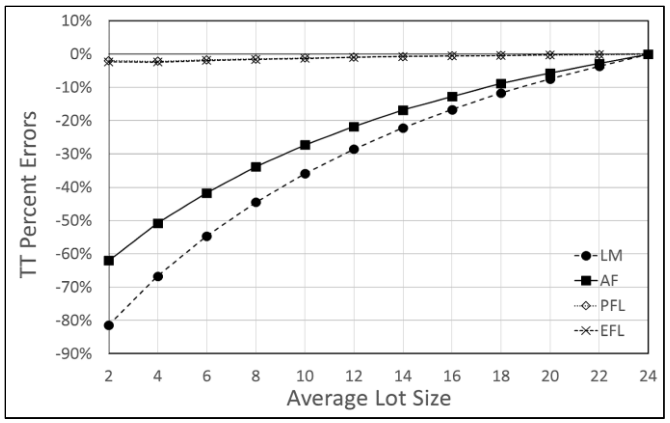

Fig. 15. Percent errors in mean throughput time for different lot sizes. Models are parameterized with lot sizes {23, 24, 25}.

6 and lot size of {23, 24, 25} and simulated with a train size of 3 and varying lot sizes of {1, 2, 3} to {23, 24, 25}. Note that the parameters adjusted are external to the tool. Again, we only show the results of the first setup case with only reticle alignment setups.

The Type III simulations reveal the limitations of LM and AF. Figs. 13 to 15 show the percent errors in average CT, LRT, and TT of the various models when they are trained at lot sizes

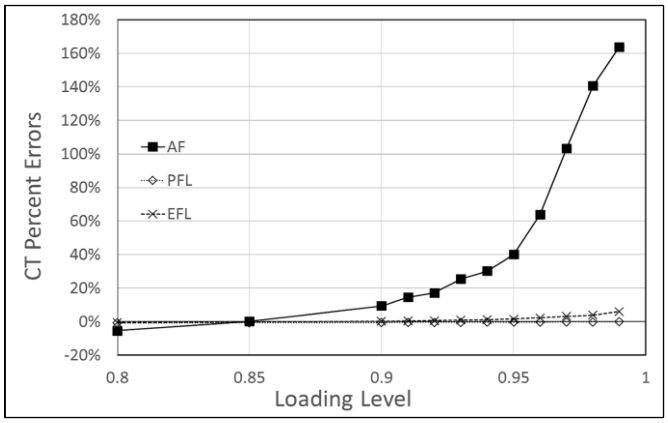

Fig. 16. Percent errors of mean cycle time for different loading levels. Models are parameterized with 0.95 loading.

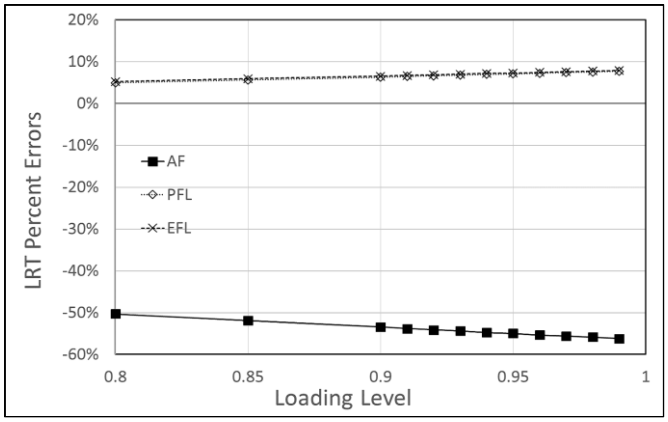

Fig. 17. Percent errors in mean lot residency time for different loading levels. Models are parameterized with 0.95 loading.

of {23, 24, 25}. Both LM and AF have very high errors on all three metrics: up to 80% for LM and 70% for AF. Even for TT, LM and AF have errors up to 60% and 80%, respectively. PFL and EFL predict all three metrics accurately, with errors less than 4% for CT, 7% for LRT, and 2.5% for TT.

Figs. 16 to 18 show the percent errors in mean CT, LRT, and TT when the models are trained at 0.95 loading. LM is not shown for ease of illustration; it has slightly worse errors than AF. LM and AF again perform poorly, with errors of up to 160% for CT, 55% for LRT, and 5% for TT.

FL models are clearly more accurate and robust to changes than the LM and AF models on all three metrics.

The other Type III simulations are similar.

E. Detailed Model Validation

As discussed in Section II, the models were compared against tool log data obtained from industry, to serve as a sanity check for our detailed model. The industry data consists of wafer advancement from process to process inside the CPT. It includes random reticle setup times and possibly other random events experienced between advancement from process to process. LM, AF, and EFL were trained on this tool log data and then simulated against the same data, as in our Type I simulations. The results are shown in Table VIII. PFL is not included

TABLE VIII
MODEL PERFORMANCE WITH REAL CPT DATA (TYPE I)

Model	LRT (seconds/lot)					TT (seconds/lot)				
	Average	σ	% Error	|Error|	Error σ	Average	σ	% Error	|Error|	Error σ
CPT Data	3957.825	602.111				1992.881	505.364			
LM	1992.881	270.432	-49.647%	1964.944	687.570	1992.881	270.432	0.000%	290.234	461.274
AF	1992.881	282.276	-49.647%	1964.944	767.094	1992.881	282.276	0.000%	215.124	403.900
EFL	3839.506	632.559	-2.989%	118.319	86.199	1981.984	479.814	-0.547%	39.997	52.015

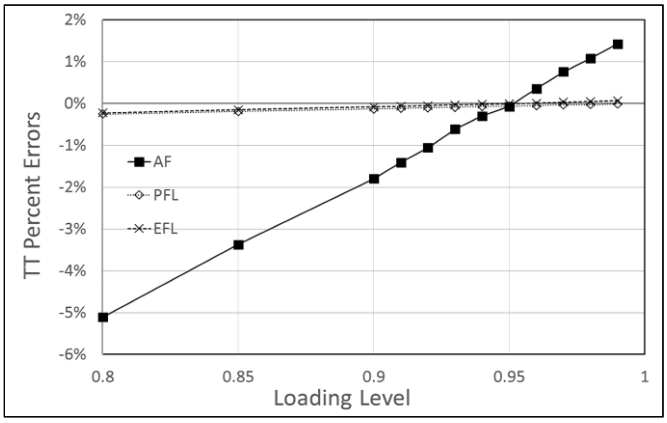

Fig. 18. Percent errors in mean throughput time for different loading levels. Models are parameterized with 0.95 loading.

as it uses parametric data derived from tool parameters. CT is also not used as we do not have lot arrival data. The units are seconds/lot, as before. Since the data is from a CPT with a different configuration than our DS, the data values are naturally different from those in the previous studies. We focus on errors and other secondary metrics.

The Table VIII results are similar to our Type I results. LM and AF have LRT errors of 50%, while EFL has errors of 3%. For TT, LM and AF have close to zero errors (as they are trained on throughput) and EFL has errors of approximately 0.5%. The secondary metrics also paint a similar picture, with EFL having lower absolute errors and error standard deviations.

When trained on the industry data, the LM, AF and EFL models exhibit errors similar to those of our previous Type I studies. We consider this encouraging in the sense that our DS model and the industry data give the same results. It serves to provide some validation for our DS model.

VII. CONCLUSION

We have developed extensions to, provided detailed parameterization and simulation equations for, and assessed the behavior of linear, affine, and flow line models of CPTs for use in fab-level simulation. We explored fidelity, robustness, expressive capability and computation times.

In particular, we developed an extension to the affine model and detailed how to extract process time parameters for flow line models given raw data from a CPT. We proposed a method to construct flow lines from tool log data and consider both parametric and empirical flow lines for our studies. The model predictions for cycle time, lot residency time and throughput time were compared to those from a detailed CPT model. Linear and affine models – which are trained intentionally to match throughput time – do not well model the cycle time and lot residency time (even when these models are fed the exact same data used for their parameter extraction). Flow line models were accurate on all metrics considered, frequently with less than 2% error for CT, 9% for LRT, and 0.5% for TT.

We explored the robustness of the models. The accuracy of linear and affine models degrades when used in conditions that deviate from their training data. Flow line models are much less sensitive to such changes. We recommend using parametric flow lines if possible and empirical flow lines if processing times are unavailable.

The tradeoff between fidelity and computation times was explored. Flow lines are about 200 times less computationally demanding than the detailed model. Linear and affine models are about 500 times less computationally complex than flow line models.

In the future, extending the work of [6] to the case of multi-cluster tools by incorporating parallel lot processing into those models would improve their accuracy (but robustness could still be a concern). In addition, instead of using computation times, the algorithmic complexity of the models could be analyzed to prove their exact computational requirements. Further, one could develop new non-linear models that improve upon the current affine models. Is it possible to develop a model that is nearly as accurate as the flow lines while retaining the excellent computational requirements of affine models and simultaneously be robust to changing fab conditions?

REFERENCES

[1] J. A. Jimenez, G. T. Mackulak, and J. W. Fowler, "Levels of capacity and material handling system modeling for factory integration decision making in semiconductor wafer fabs," *IEEE Trans. Semicond. Manuf.*, vol. 21, no. 4, pp. 600–613, Nov. 2008.

[2] J. W. Fowler, L. Mönch, and T. Ponsignon, "Discrete-event simulation for semiconductor wafer fabrication facilities: A tutorial," *Int. J. Ind. Eng. Theory Appl. Pract.*, vol. 22, no. 5, pp. 661–682, Jan. 2015.

[3] D. Fandel and R. Wright, "300 mm productivity detractors mitigation cost analysis," in *Proc. IEEE/SEMI Adv. Semicond. Manuf. Conf.*, Cambridge, MA, USA, 2008, pp. 222–227.

[4] J.-E. Kiba, G. Lamiable, S. Dauzère-Pérès, and C. Yugma, "Simulation of a full 300mm semiconductor manufacturing plant with material handling constraints," in *Proc. Win. Simulat. Conf.*, Austin, TX, USA, 2009, pp. 1601–1609.

[5] C.-H. Hsieh, C. Cho, T. Yang, and T.-J. Chang, "Simulation study for a proposed segmented automated material handling system design for 300-mm semiconductor fabs," *Simulat. Model. Pract. Theory*, vol. 29, pp. 18–31, Dec. 2012.

[6] K. Schmidt, J. Weigang, and O. Rose, "Modeling semiconductor tools for small lot size FAB simulations," in *Proc. Win. Simulat. Conf.*, 2006, pp. 1811–1816.
[7] C.-N. Wang and C.-H. Wang, "A simulated model for cycle time reduction by acquiring optimal lot size in semiconductor manufacturing," *Int. J. Adv. Manuf. Technol.*, vol. 34, nos. 9–10, pp. 1008–1015, Oct. 2007.
[8] K. Schmidt and O. Rose, "Development and simulation assessment of semiconductor fab architectures for fast cycle times," Ph.D. dissertation, Forum at SimVis, Magdeburg, Germany, 2007.
[9] E. Zarifoglu, R. Wright, C. Bubela, and J. Preece, "Modeling semiconductor factories for equipment and cycle time reduction opportunities, Part II," *Future Fab Int.*, vol. 25, pp. 54–59, Apr. 2008.
[10] M. Lapedus, *EUV Tool Costs Hit $120 Million*, *EE Times*, San Francisco, CA, USA, Nov. 2010. [Online]. Available: http://www.eetimes.com/document.asp?docid=1257963
[11] K. E. Kabak, C. Heavey, V. Corbett, and P. J. Byrne, "Impact of recipe restrictions on photolithography toolsets in an ASIC fabrication environment," *IEEE Trans. Semicond. Manuf.*, vol. 26, no. 1, pp. 53–68, Feb. 2013.
[12] B. Yan, H. Y. Chen, P. B. Luh, S. Wang, and J. Chang, "Litho machine scheduling with convex hull analyses," *IEEE Trans. Autom. Sci. Eng.*, vol. 10, no. 4, pp. 928–937, Oct. 2013.
[13] A. Bitar, S. Dauzère-Pérès, and C. Yugma, "On the importance of optimizing in scheduling: The photolithography workstation," in *Proc. Win. Simulat. Conf.*, Savanah, GA, USA, 2014, pp. 2561–2570.
[14] M.-C. Wu and C.-W. Chiou, "Scheduling semiconductor in-line steppers in new product/process introduction scenarios," *Int. J. Prod. Res.*, vol. 48, no. 6, pp. 1835–1852, Mar. 2010.
[15] C.-W. Chiou and M.-C. Wu, "Scheduling of multiple in-line steppers for semiconductor wafer fabs," *Int. J. Syst. Sci.*, vol. 45, no. 3, pp. 384–398, 2014.
[16] K. Park and J. R. Morrison, "Controlled wafer release in clustered photolithography tools: Flexible flow line job release scheduling and an LMOLP heuristic," *IEEE Trans. Autom. Sci. Eng.*, vol. 12, no. 2, pp. 642–655, Apr. 2015.
[17] Y. J. Park and H. R. Hwang, "Minimization of total processing time in semiconductor photolithography process," *Appl. Mech. Mater.*, vols. 325–326, pp. 88–93, Jul. 2013.
[18] J. R. Morrison, "Multiclass flow line models of semiconductor manufacturing equipment for fab-level simulation," *IEEE Trans. Autom. Sci. Eng.*, vol. 8, no. 1, pp. 81–94, Jan. 2011.
[19] A. A. A. Kock, C. P. L. Veeger, L. F. P. Etman, B. Lemmen, and J. E. Rooda, "Lumped parameter modelling of the litho cell," *Prod. Plan. Control Special Issue Novel Models Approaches Semicond. Manuf.*, vol. 22, no. 1, pp. 41–49, 2011.
[20] J. R. Morrison and D. P. Martin, "Performance evaluation of photolithography cluster tools," *OR Spectr.*, vol. 29, no. 3, pp. 375–389, Jul. 2007.
[21] W.-S. Kim and J. R. Morrison, "The throughput rate of serial production lines with deterministic process times and random setups: Markovian models and applications to semiconductor manufacturing," *Comput. Oper. Res.*, vol. 53, pp. 288–300, Jan. 2015.
[22] H. J. Yoon and D. Y. Lee, "Deadlock-free scheduling of photolithography equipment in semiconductor fabrication," *IEEE Trans. Semicond. Manuf.*, vol. 17, no. 1, pp. 42–54, Feb. 2004.
[23] H. N. Geismar, C. Sriskandarajah, and N. Ramanan, "Increasing throughput for robotic cells with parallel machines and multiple robots," *IEEE Trans. Autom. Sci. Eng.*, vol. 1, no. 1, pp. 84–89, Jul. 2004.
[24] E. Lefeber and D. Armbruster, "Aggregate modeling of manufacturing systems," in *Planning Production and Inventories in the Extended Enterprise*. New York, NY, USA: Springer US, 2011, pp. 509–536.
[25] A. A. A. Kock *et al.*, "Performance measurement and lumped parameter modeling of single server flow lines subject to blocking: An effective process time approach," *Comput. Ind. Eng.*, vol. 54, no. 4, pp. 866–878, May 2008.
[26] C. P. L. Veeger *et al.*, "Predicting cycle time distributions for integrated processing workstations: An aggregate modeling approach," *IEEE Trans. Semicond. Manuf.*, vol. 24, no. 2, pp. 223–236, May 2011.
[27] R. J. Brooks and A. M. Tobias, "Simplification in the simulation of manufacturing systems," *Int. J. Prod. Res.*, vol. 38, no. 5, pp. 1009–1027, Nov. 2000.
[28] J. Van der Eerden, W. Walbrick, H. Niesing, T. Saenger, and R. Schuurhuis, "Litho area cycle time reduction in an advanced 300mm semiconductor manufacturing line," in *Proc. IEEE/SEMI Adv. Semicond. Manuf. Conf.*, Boston, MA, USA, 2006, pp. 114–119.
[29] J. R. Morrison, "On the fidelity of the Ax+B equipment model for clustered photolithography scanners in fab-level simulation," in *Proc. Win. Simulat. Conf.*, Phoenix, AZ, USA, 2011, pp. 2034–2044.
[30] S. Radloff *et al.*, "First wafer delay and setup: How to measure, define and improve first wafer delays and setup times in semiconductor fabs," in *Proc. IEEE/SEMI Adv. Semicond. Manuf. Conf.*, Berlin, Germany, 2009, pp. 86–90.
[31] T. Altiok, *Performance Evaluation of Manufacturing Systems*. New York, NY, USA: Springer-Verlag, 1997.
[32] B. Avi-Itzhak, "A sequence of service stations with arbitrary input and regular service times," *Manag. Sci.*, vol. 11, no. 5, pp. 565–571, Mar. 1965.
[33] H. D. Friedman, "Reduction methods for tandem queuing systems," *Oper. Res.*, vol. 13, no. 1, pp. 121–131, Feb. 1965.
[34] J. R. Morrison, "Deterministic flow lines with applications," *IEEE Trans. Autom. Sci. Eng.*, vol. 7, no. 2, pp. 228–239, Apr. 2010.
[35] J.-H. Kim, T.-E. Lee, H.-Y. Lee, and D.-B. Park, "Scheduling analysis of time-constrained dual-armed cluster tools," *IEEE Trans. Semicond. Manuf.*, vol. 16, no. 3, pp. 521–534, Aug. 2003.
[36] Q. Zhu, N. Wu, Y. Qiao, and M. Zhou, "Petri net modeling and one-wafer scheduling of single-arm multi-cluster tools," in *Proc. IEEE Int. Conf. Autom. Sci. Eng.*, Madison, WI, USA, 2013, pp. 862–867.
[37] D. Pillai, "The future of semiconductor manufacturing," *IEEE Robot. Autom. Mag.*, vol. 13, no. 4, pp. 16–24, Dec. 2006.

Jung Yeon Park received the B.S. and M.S. degrees in industrial and systems engineering from KAIST, South Korea, in 2014 and 2016, respectively. He is currently a Systems Engineer with Samsung Electronics, DS Division, South Korea. His research interests include semiconductor equipment modeling and simulation and flow lines.

Kyungsu Park received the B.S. degree in information and industrial engineering and the B.S. degree in chemical engineering from Yonsei University, South Korea, in 2008, and the Ph.D. degree from the Department of Industrial and Systems Engineering, KAIST, South Korea, in 2014.

He was a Senior Researcher with the Defense Agency of Technology and Quality and the Busan Institute of S&T Evaluation and Planning, South Korea. He is currently a Post-Doctoral Researcher with the Department of Industrial and Systems Engineering, University of Wisconsin-Madison. His research interests include modeling, analysis, and control of manufacturing.

James R. Morrison (S'97–M'00) received two B.S. degrees in electrical engineering and in mathematics from the University of Maryland, College Park, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Illinois at Urbana-Champaign. He was with the Fab Operations Engineering Department, IBM Corporation from 2000 to 2005. He is currently an Associate Professor with the Department of Industrial and Systems Engineering, KAIST, Daejeon, South Korea. In 2009, he became a Co-Chair of the IEEE Robotics and Automation Society Technical Committee on Semiconductor Manufacturing Automation. His research interests include semiconductor wafer fabrication, systems of UAVs, and engineering of service systems.