A three tier cooperative control architecture for multi-step semiconductor manufacturing proces

Mao, Ziqiang; Kang, Wei; Wang, Frank; Raulefs, Peter

Elsevier Ltd.

Mao, J. & Kang, W. 2008, "A three tier cooperative control architecture for multi-step semiconductor manufacturing process", Journal of Process Control, vol. 18, pp. 954-960.
http://hdl.handle.net/10945/57017

Contents lists available at ScienceDirect

Journal of Process Control

journal homepage: www.elsevier.com/locate/jprocont

A 3-tier cooperative control architecture for multi-step semiconductor manufacturing process

Ziqiang Mao [a,*], Wei Kang [b], Frank Wang [a], Peter Raulefs [a]

[a] *Intel Corporation, Santa Clara, CA 95052, USA*
[b] *Naval Postgraduate School, Monterey, CA 93943, USA*

ARTICLE INFO

Article history:
Received 27 June 2007
Received in revised form 29 January 2008
Accepted 6 April 2008

Keywords:
Process control
Optimization
Cooperative control
Target optimization
Quality control

ABSTRACT

In this paper, cooperative control is investigated and applied to chained processes with multiple steps and multiple tools in semiconductor manufacturing. A cooperative control architecture is proposed to optimize product quality, to improve yield, to achieve best tool performance, and to minimize throughput time. The architecture consists of three tiers: the top tier for target optimization and overall product performance, the middle tier for tool selection based on tool performance, throughput time and tool availability, and the bottom tier for tool level run-to-run control. Large data sets are collected from four individual process steps in a fabrication facility of a leading semiconductor manufacturer and the data sets are processed and lined up for the study of cooperative control. Monte Carlo simulations are carried out based on the real data to demonstrate a significant improvement for the end-of-line product quality.

1. Introduction

In a highly competitive CPU (central process unit) market, it is critical to manufacture high quality wafers with high yield and fast throughput time. The quality of wafers is determined in large by the manufactured circuits on wafers with hundreds of dies to be packaged and installed as CPUs in computers and other electronic products. The quality of circuits on wafers is measured by electrical testing measurement that catches the electrical characteristics on certain wafer structures, such as capacitances and resistances. These electrical features determine performance properties, such as CPU speed and power consumption. Therefore an important aspect of quality control in semiconductor manufacturing is to effectively control these electrical characteristics that lead to CPU speed and power efficiency. In addition, throughput time is important to the productivity of manufacturing. Semiconductor manufacturing is a complex process with hundreds of process steps. The idle time of individual processing tools is a major factor that causes manufacturing inefficiency.

A semiconductor manufacturing fabrication facility includes a large variety of equipment (or tools) used to process wafers in the following functional areas: diffusion, photolithography, etch, thin films, ion implant, and polish. In these functional areas, there are different types of process tools such as high temperature diffusion furnaces, wet cleaning stations, stepper tool, plasma etchers, and ion implanter etc.

Currently, every process step in manufacturing is controlled to meet a pre-defined target or a set of specification limits preset in the process development. SPC (statistical process control) and APC (advanced process control) are typical methods of process control applied in manufacturing. The SPC methods have been used for decades to monitor the process capability and stability, such as process drift trending and OOC (out-of-control) events. In many cases, the response to process problems is to shut down equipment, troubleshoot the problem and/or adjust the recipe. For the last decade, APC has been increasingly used in the semiconductor industry to improve yields. Run-to-run process control and multivariate fault detection and classification are two major techniques in APC used by today's semiconductor industry. A run-to-run controller is able to adjust recipe automatically with a feedback control mechanism. However, SPC and APC only work at each individual process step locally and multiple process steps do not work cooperatively to improve the product quality at the end of multiple process steps. In currently deployed control regimes, the deviation from the target at a process step is not automatically compensated in a subsequent process steps. The lack of cooperation results in deviations in final product quality such as the speed and the leakage current.

In other words, stand-alone controllers do not effectively use information from the multiple tools. On the other hand, if all related steps are taken into account, a deviation at one operation step

[*] Corresponding author. Tel.: +1 408 765 9584; fax: +1 408 765 9206.
E-mail address: John.mao@intel.com (Z. Mao).

0959-1524/$ - see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jprocont.2008.04.003

can be compensated in downstream operation steps. When we treat the complex system as a cooperative system, all tools communicate and cooperate with each other to achieve a common goal. Instead of controlling tools independently, we control the cooperative system collectively.

A cooperative system is defined to be multiple dynamic entities that share information or tasks to accomplish a common, though perhaps not singular, objective. Cooperative control has been widely used in a variety of engineering applications such as unmanned aircraft and satellite formations. In a complex cooperative system, all tools share information on process specification, recipes, measurements on wafer, and operational constraints. Readers are referred to [5] for background and literature of cooperative control.

In related earlier work, Qin [1] proposed fab-wide control to adjust targets at each process step dynamically so that downstream targets can be reset to compensate deviations at upstream steps. In [4], Qin et al. gave a detailed framework combining the run-to-run control and fault detection. Several approaches toward integrating individual control systems have started to emerge [2,3]. Multi-step EPC (electrical parameter control) control was proposed [4] to minimize an objective function that penalizes the difference between the desired electrical properties and the updated model output subject to constraints. The control mechanism is to re-target electrical parameters.

In this paper, we have a different perspective in dealing with this problem. Taking into the consideration of all interactions and communications among tools and control systems, this problem calls for a multiple-tier cooperative controller.

In this paper, we develop a 3-tier cooperative control architecture (Fig. 1) that not only optimizes product performance and quality goals, but also achieves optimal tool selection for the best tool performance and the minimal throughput time.

The 3-tier cooperative control architecture has the following components:

- The top tier is target optimization. It optimizes the targets at each module or functional area.
- The middle tier is the tool selection optimization based on the information of tool performance and tool availability. This tier uses operational information and tool performance information to decide the best tool to use for a process step.
- The bottom tier is the tool level run-to-run control.

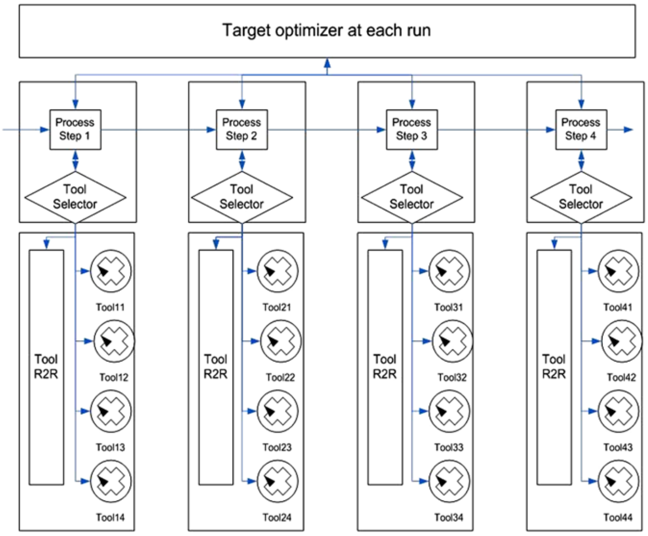

Fig. 1. 3-tier cooperative control architecture.

While we focus on the product performance, the control architecture is designed such that it is simple to take into account the waiting time, equipment downtime, and the impact of scheduled preventive maintenance. In addition, large data sets are collected from individual process steps in a manufacturing fabrication facility and the data sets are processed and lined up for the study of cooperative control. Monte Carlo simulations are carried out based on the real data to demonstrate a significant improvement for the end-of-line product performance.

The paper is organized as the following. In Section 2, we formulate the target optimization problem and derive the solution of dynamic targets to each process step. In Section 3, by utilizing dynamic targets and feedforward information, we construct the tool selection criteria to achieve the balance of best tool performance and minimal throughput time. In Section 4, the bottom tier is described as tool level control. Simulation results with production data from a leading semiconductor manufacturing company is demonstrated in Section 5.

2. Target optimization

The semiconductor product performance is measured by a set of key electrical test parameters. For a CPU product, the frequency of the circuit and the power consumption are important measures which have direct impact on its marketability. The frequency indicates the speed of the CPU and the power consumption is mainly affected by leakage current. These final performance parameters, denoted as $\mathbf{y}$, are related to many metrology and electrical testing parameters $\mathbf{x}$, an n-dimensional vector $\mathbf{x} = [x_1, x_2, x_3,\ldots,x_n]^{\mathrm{T}}$, such as the circuit gate length, threshold voltage, overall capacitance, etc. In the n-dimensional column vector $\mathbf{x} = [x_1, x_2, x_3,\ldots,x_n]^{\mathrm{T}}$, where T stands for the transpose of the vector, the sequence of positive integers represents the sequence of operations. So, operation step $k-1$ takes place before step k, where $k = 2,3,\ldots, n$.

Our first goal is to identify the key parameters $\boldsymbol{x}$ that are most influential to the product performance parameters $\mathbf{y}$. We use the physical knowledge to select relevant parameters and repeatedly validate their statistical impacts with statistical models. It is not practical to develop either a completely physical model, or a purely statistical model. In many cases, a physical model does not exist or the parameters in the model are not available, or only partially available. On the other hand, a purely statistical model may not provide adequate information about the underlying physical relationships, which are useful for controller design. In the approach that we adopt, we apply physical knowledge to capture parameters that are relevant to the model; and then recursively apply statistical models to validate the parameters and the correctness of the model. So $\mathbf{x}$ is identified as the key parameter set. We then build a model for the relationship between $\mathbf{x}$ and $\mathbf{y}$ from historical data. $\mathbf{y}(m)$ at run m does not depend on previous values $\mathbf{y}(m-1)$ at the previous run $m-1$, because the semiconductor equipment minimizes the autocorrelation between runs. A run is a lot, a batch of wafers, in the study. The relationship between $\mathbf{x}$ and $\mathbf{y}$ for the mth run is

$$\mathbf{y}(m) = \mathbf{f}(\mathbf{x}(m)) + \mathbf{w}(m), \tag{1}$$

where $\mathbf{w}$ is a noise vector and $\mathbf{f}$ is a vector function. In the model above, the function $\mathbf{f}$ is usually a linear function in a high volume manufacturing process because the process is running at a nominal operation zone.

A desired product performance parameter vector, denoted by $\mathbf{y}_{\mathrm{T}}$, is specified as a target. The optimization objective is to minimize the difference between $\mathbf{y}$ and $\mathbf{y}_{\mathrm{T}}$. At the operation step k ($k \leqslant n$), parameters [x_1, x_2, x_3, $\ldots$, x_k] for run m are known as

$\mathbf{z}_k(m) = [z_1(m), z_2(m), z_3(m),\ldots,z_k(m)]$ which may not perfectly meet the targets at these k steps. We can use the rest of n–k steps that will be processed to compensate the errors at previous k steps. The way to compensate is to find the right targets at the rest n–k steps that optimize the outcome for the run m. Mathematically, it is to minimize the following objective function:

$$J_{km} = \min\{||y_\mathrm{T} - f(z_k(m), \mathbf{x}_k(m))||^2 + \mu||\mathbf{x}_k(m) - \mathbf{x}_k(m-1)||^2\}, \quad (2)$$

where $\mathbf{x}_\mathrm{k}(m)$ is the vector $[x_{1+k}(m), x_{2+k}(m), x_{3+k}(m),\ldots,x_n(m)]$ and μ is a weight that helps to avoid jerky or irregular control sequences. To meet the physical specifications, the function $\mathbf{f}$ should be bounded by lower and upper control limits $\mathbf{y}_\mathrm{L}$ and $\mathbf{y}_\mathrm{U}$ of $\mathbf{y}$, and the parameters $\mathbf{x}_\mathrm{k}(m)$ = [$x_{1+k}(m)$, $x_{2+k}(m)$, $x_{3+k}(m),\ldots,x_n(m)$] should be bounded by their upper control limits (UCL) and lower control limits (LCL) which are used in statistical process control charts:

$$\mathrm{LCL}_i \leqslant x_i(m) \leqslant \mathrm{UCL}_i, i = k+1.k+2,\ldots,n \quad (3)$$

$$\mathbf{y}_\mathrm{L} \leqslant \mathbf{f}(\mathbf{x}(m)) \leqslant \mathbf{y}_\mathrm{U} \quad (4)$$

The solution to the optimization problem in Eq. (2) subject to the constraints Eq. (3)(4) can be computed by using nonlinear programming if function $\mathbf{f}$ is nonlinear, or quadratic programming if $\mathbf{f}$ is a linear function.

It is well known that it is a key challenge to create the model $\mathbf{f}$ in Eq. (1). There are two ways to create this model. One is to build a first principle model with engineering knowledge. However, we can only find limited first principle models for some local processes, but not for all the tools. Thus, finding first principle models for complex semiconductor manufacturing systems is an impractical approach. An alternative is to build empirical models from historical manufacturing data. Because manufacturing processes may change and drift, the empirical model must be updated in real-time to accommodate process changes. If $\mathbf{f}$ has a known structure, such as a linear function, we can use the recursive least square procedure to build the model using historical data in a given time window.

Once the optimal targets for the next n–k steps are found by solving the optimization problem defined by Eq. (2)(3)(4), the optimal target is then assigned to process step k+1. Now, it is the task of the middle tier controller to find the best tool from the available tool clusters to process this run of wafers.

3. Tool selection

When a target is optimized for a process step, we need to select a best tool out of a set of available tools to perform the task. There are two criteria for searching a best tool. One is the current tool performance that indicates the tool's possibility of meeting the updated target for run m at process step k. Tool performance can be assessed in real-time. The other criterion consists of the tool availability and tool condition indicating the predicted throughput time for run m at step k. Based on these two criteria that characterize the performance and throughput time we can find the best tool to perform the process.

It is an ideal and rare case that all tools perform identically and have the same throughput time. In this case, any tool can be equally used as long as the tool is available. In reality it is very common that many tools have the same throughput time because these tools are from one single vendor and run a similar recipe. Tools usually perform differently due to various reasons. Although factories constantly try to match all tools to the best, differences between tools still exist in performance. Tools usually do not have the same availability due to maintenance schedules. So we have to take these factors in the tool selection. Therefore, tool selection is influenced by tool performance, tool availability and throughput time. The controller design for this middle tier is described in the following three subsections.

3.1. Best tool performance

The tool performance is mathematically formulated as follows. Let x_p and x be the outputs at the previous process step $k-1$, and the current step k, respectively. Denote x_T as the target value for the process step k for run m, and Δ as the tolerance margin of the output. We use probability to define the performance indicator of tools. More specifically, the performance indicator of tool i is

$$\begin{aligned} J_p(i) &= P_i(x_\mathrm{p}, x_\mathrm{T}, \Delta) \\ &= \mathrm{Prob}\{x \in [x_\mathrm{T} - \Delta, x_\mathrm{T} + \Delta]|\text{past runs with}x_\mathrm{p}\text{in a given range around}x_\mathrm{p}(m)\} \end{aligned} \quad (5)$$

Therefore, $J_\mathrm{p}(i)$ is a number between 0 and 1. The goal is to select the tool with maximal probability $J_\mathrm{p}(i)$ among all tools. This probability is in fact a performance assessment of the equipment to produce wafer with desired output in the interval $[x_\mathrm{T}-\Delta, x_\mathrm{T}+\Delta]$ given the condition of previous operations.

In a semiconductor manufacturing process, x_p is the information from previous process steps that has been completed, which is treated as the input to the next process steps. For example, when x is the final inspection critical dimension (FICD) at an etching step, x_p may represent the development inspection critical dimension (DICD) on the wafers at the preceding lithography step. The quality of DICD definitely affects the quality of FICD. The margin Δ is selected such that the parameter x is in the interval $[x_\mathrm{T}-\Delta, x_\mathrm{T}+\Delta]$, where the output meets the performance requirement. Δ is usually selected as 1 to 1.5 standard deviations of the measurement x around x_T. In the following, we give two ways to compute J_p(i).

3.1.1. Method 1: model-based method

For the ith tool at process step k, we build a tool model M_{ik} with tool input, target and feedforward information

$$x = g(x_\mathrm{p}, x_\mathrm{T}, u) + \omega, \quad (6)$$

where u is the control parameter of step k, ω is a random noise and x_p is the information from previous process steps. More specifically, we can build a linear equation in most cases with an item c representing other known factors:

$$x = au + bx_\mathrm{p} + x_\mathrm{T} + c + \omega \quad (7)$$

For example, we can build a model for an etcher in which the etching output FICD is modeled as a function of etching time T_etch and *DICD*. Here x = FICD, x_p = DICD, and $u = T_\mathrm{etch}$, and x_T is the target FICD$_\mathrm{T}$. The item c represents the disturbance from incoming thin film thickness. The model in Eq. (7) is FICD = $a * T_\mathrm{etch}$ + $b *$ DICD + FICD$_\mathrm{T}$ + c + ω.

Given x_p and Δ, the actual x may or may not fall into the desired vicinity of the target $[x_\mathrm{T}-\Delta, x_\mathrm{T}+\Delta]$. The probability of x falling into the vicinity of the target is

$$P_i(x_\mathrm{p}, x_\mathrm{T}, \Delta) = \mathrm{Prob}(x \epsilon [x_\mathrm{T} - \Delta, x_\mathrm{T} + \Delta]|u \epsilon [u(m-1) - \delta, u(m-1) + \delta]) \quad (8)$$

The condition $u \in [u(m\text{-}1)\text{-}\delta, u(m\text{-}1)\text{+}\delta]$ is to ensure the control knob at run m will not move far away from run $m-1$.

3.1.2. Method 2: empirical data based method

This method does not depend on the control knobs at this step. Instead, this method uses the empirical data to define the correlation of the previous step output x_p and the desired output x_T. In the historical data of M observations of x_p and x, we divide all M records of x_p into three data sets, $\boldsymbol{G}_1$, $\boldsymbol{G}_2$, $\boldsymbol{G}_3$, to represent high, middle, and low value of previous step output x_p. For each group $\boldsymbol{G}_\mathrm{j}$ and each tool i, we can compute the probability of output x in the interval $[x_\mathrm{T}-\Delta, x_\mathrm{T} + \Delta]$ for a given $x_\mathrm{p} \in Gj$

$$P_{ij}(x_{\rm p}, x_{\rm T}, \Delta) = \mathrm{Prob}(x \ \epsilon \ [x_{\rm T} - \Delta, x_{\rm T} + \Delta] | x_{\rm p} \in G_j,) \tag{9}$$

For example, when at step k, the previous step output is in $\boldsymbol{G}_2$, the probability of x falling into the right target region is

$$P_{i2}(x_{\rm p}, x_{\rm T}, \Delta) = \mathrm{Prob}(x \ \epsilon \ [x_{\rm T} - \Delta, x_{\rm T} + \Delta] | x_{\rm p} \ \epsilon \ G_2) \tag{10}$$

3.2. Best throughput time and tool availability

The throughput time ($T_{\rm t}$) is mainly a sum of required process time ($T_{\rm p}$), setup time ($T_{\rm s}$), and waiting time ($T_{\rm w}$). The required process time is the time required for the tool to process a lot, a batch of wafers, and also include cure time for wafers needed in some process steps. The setup time accounts for setup between operation steps. The waiting time is the time when a lot is hold in idle status because a tool or an operator is not available.

For tools with same model from same supplier, process time $T_{\rm p}$ and setup time $T_{\rm s}$ of two different tools are expected to be close. Hence waiting time $T_{\rm w}$ becomes the major factor impacting the throughput time.

For heterogeneous tools with different models, process time $T_{\rm p}$ and setup time $T_{\rm s}$ can be much different. Hence we can make a wise decision on which tool to use to minimize the throughput time and also achieve the performance goal.

Tool availability is another constraint that needs to be included in the optimization. Every tool is subject to some scheduled preventative maintenance (PM) activities that take a tool down for some time and then recalibrate. When a tool is required to do PM, the tool will not be available for wafer processing for a certain period of time. When we decide which tool to use, we have to take the PM schedule into account. Suppose the time to the next scheduled PM event for tool i is $T_{\rm pm}(i)$, based on a maintenance schedule system or a predictive maintenance system, and the throughput time for tool i is $T_{\rm t}(i)$. Obviously, the tool is available only when the throughput time is less than the time $T_{\rm pm}(i)$ to the next scheduled PM for tool i. Then we have the following equations.

$$J_{\rm t}(i) = T_{\rm t}(i), \quad \mathrm{when} T_{\rm t}(i) < T_{\rm pm}(i), i = 1, 2, \ldots, N \tag{11}$$

$$J_{\rm t}(i) = +\infty, \quad \mathrm{when} T_{\rm t}(i) \geqslant T_{\rm pm}(i), i = 1, 2, \ldots, N \tag{12}$$

We want to select a tool that is available and has the shortest throughput time among all N tools. If $T_{\rm pm}(i)$ is less than $T_{\rm t}(i)$, tool i is not available to use. That is why it is defined as $+\infty$ in Eq. (12).

3.3. Best tool selection

Tool selection is a multi-objective task in which we sometimes have to compromise and find a good balance between performance and throughput time. To combine the two objectives $J_{\rm p}(i)$ and $J_{\rm t}(i)$, in one optimization procedure, we use the following objective function

$$J(i) = \eta^* J_{\rm t}(i) + (1 - \eta)^* (1 - J_{\rm p}(i)), \tag{13}$$

$$= \eta^* T_{\rm t}(i) + (1 - \eta^* (1 - P_i(x_{\rm p}, x_{\rm T}, \Delta)) \tag{14}$$

where η is a weighting factor. The optimization problem is to find the minimum Min{$J(i)$; i = 1,2,3,. . .,N}.

When throughput is more important than performance, η should be selected to be greater than 0.5. Otherwise η should be less than 0.5. There are two extreme cases. When η is 1, it corresponds to a scheduling optimization with an assumption that all tools are equal in performance. When η is 0, this corresponds to selecting a tool with best performance by assuming all tools are equally available and have the same throughput time.

In the optimization, there are at most N possible solutions. Hence solving the problem is very straightforward.

The approach in Eq. (13) is a systematic and implementable way to balance several factors including tool performance, throughput time, and tool availability for tools that are either homogeneous with same model or heterogeneous tools from different vendors. The optimization approach is superior to the current practice in manufacturing because, in many cases, available tools in fabs are just blindly used regardless of their performances. In some functional areas, the performance of a tool is taken into account in a qualitative way when it is found to perform worse than others more frequently. In this case, a tool with less satisfied performance is used only for non-critical operations steps or no other tools are available. Different from the current practice, the approach proposed in Eq. (13) is a quantitative way of managing multiple tools to balance and optimize multiple factors.

4. Tool level control

The third tier in the cooperative control architecture is the tool level control. After a best target is defined and a best tool is selected, our next task is to control the tool to reach the target.

In fact, semiconductor manufacturing equipment has to be controlled in two ways: internally and externally. The internal control mechanism is to make sure that the equipment can be operated at a desired setting. For example, when an equipment operator sets a furnace to process a batch of wafers at a desired temperature, the internal controller inside the furnace keeps furnace at that temperature for a specific time period. Internal controllers are provided by equipment manufacturers within the tool. The external control is usually done by operators in a fab by adjusting the recipe of the process step. Statistical process control has been widely used for the purposes of monitoring processes and making necessary adjustments in the settings. Automatic control of settings is performed by run-to-run (R2R) control.

The semiconductor industry has been using statistical process control (SPC) for several decades. SPC is set up to detect abnormal situations in processes based on output metrology parameters. When operators observe a drift, a shift, or an OOC (out-of-control) event on SPC control charts of metrology parameters, they have to decide if an adjustment of recipes or a preventive maintenance action is needed. Hence, SPC performs open-loop control, and is mainly a monitoring method where some events may even automatically trigger a shutdown of equipment.

R2R control has been used increasingly in the last decade in the semiconductor industry. Instead of just monitoring the process with SPC systems, R2R control actually changes recipe and control knobs automatically based on feedback from metrology results such as critical dimension or thickness, in order to ensure the desired metrology output. Some basics on R2R control can be found in [7].

The most commonly used R2R controller is the exponentially weighted moving average (EWMA) control method that assumes that each run is independent of the previous run. The EWMA controller is the optimal control when the process noise follows the Integrated Moving Average IMA(1,1) model.

We assume that the process model of tool level control at step k is formulated as follows:

$$x_m = \alpha u_m + \beta x_{\rm p} + x_{\rm T} + \omega, \tag{15}$$

where $x_{\rm T}$ is the desired target at process step k for run m, u_m and x_m are input and output of run m at step k, the coefficients α and β represent the slopes, $x_{\rm p}$ is the information from previous process steps as the feedforward signal, and ω is the noise and disturbance. In manufacturing, ω is assumed as an Integrated Moving Average IMA(1,1) model:

$$\omega_k = \omega_{k-1} + r_k + \gamma r_{k-1}, \tag{16}$$

where r_k is a sequence of white noises.

Let c be the estimate of ω, a the estimate of α, and b the estimate of β. Since c is an IMA (1,1) process, the following EWMA controller is an optimal controller [8] if the weight λ is properly tuned to 1 + γ:

$$u_m = (x_{\mathrm{T}} - bx_{\mathrm{p}} - x_{\mathrm{T}} - c_m)/a = (-bx_{\mathrm{p}} - c_m)/a \tag{17}$$

$$c_m = (1-\lambda)c_{m-1} + \lambda(x_{m-1} - au_{m-1} - bx_{\mathrm{p}} - x_{\mathrm{T}}), \tag{18}$$

In the above method, we hope the estimates of α, β and ω are good enough. The coefficients α and β can be estimated using historical data. However, the estimation becomes invalid as system drifts. Model mismatches always exist in reality. From time to time, we have to use adaptive modeling to ensure a correct model in controller [6]. Some results on the analysis of the impact of model mismatch and disturbances can be found in [9,10].

5. Simulations

Simulations were carried out based upon real manufacturing data. The data were used to build a system model that covers four sequential manufacturing process steps consisting of the gate oxidation process, the image pattern formation process in photolithography, the gate etching process, and the ion implantation process for shallow trench isolation. The transistor drive current leakage, a key electrical testing measurement, is used to characterize the product power consumption and it is treated as the output in the cooperative control system model. This sequence of processes is a part of standard semiconductor manufacturing lines. In modeling and simulations, some key metrology parameters in these processes were identified as model inputs such as gate length, gate oxide thickness, gate-to-source drain overlapping capacitance, junction area capacitance at gate edge and series resistance.

Different process technologies, different tools, different products, and different manufacturers have different manufacturing operational sequences in detail. Hence the relationship and the model of input and output may vary from one fab to another fab. A modeling process without engineering knowledge is neither possible nor meaningful. Thus, extensive time and effort were taken in this project to collect data for analysis, modeling, and simulations from a selected fab of a leading semiconductor manufacturer. Thanks to a close collaboration among a group of process engineers from this fab, a set of data consisting of 4015 runs was collected from a manufacturing line. In the following, this set of real data is called FabData4015. In the data analysis and simulations, we focus on four process steps: gate oxidation; photolithography; gate etching; and ion implantation. FabData4015 consists of five parameters, x_1, x_2, x_3, x_4, x_5 that are measured in these four process steps. Last two parameters x_4 and x_5 are measured at the 4th step. The data set also includes the final performance, the transistor drive current leakage y, of the 4015 runs. In fact, FabData4015 consists of several pieces of independently collected data sets. They were processed and lined up to form FabData4015 so that we can develop models that incorporate the multiple process steps in our study. A linear model is adopted for the function in Eq. (1). As a result, the model Eq. (1) has the following form

$$y = a_1x_1 + a_2x_2 + a_3x_3 + a_4x_4 + a_5x_5 + c \tag{19}$$

The error of the true output and predicted output of this model is shown in Fig. 2. The standard deviation of the error for 4015 runs is 0.02.

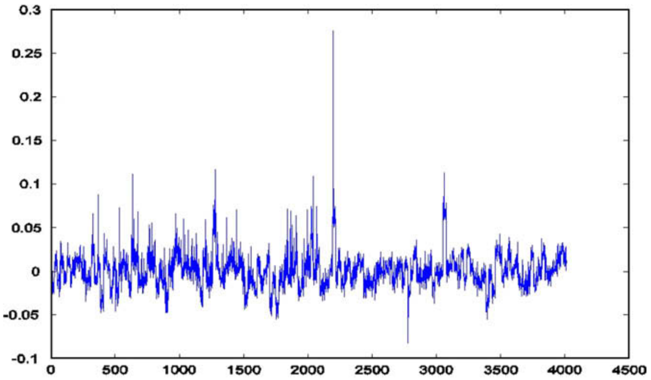

Fig. 2. Relative error of the wafer performance model.

In the simulations of target optimization, we tested the mechanism of cooperative control. For instance, suppose gate oxidation process on a wafer is already completed. As a result, suppose the value of x_1 is measured and denoted by z_1. Based on the value of x_1, the target value for the rest of individual process steps is re-computed and re-assigned so that, at the end of the entire manufacturing process, the performance of this particular lot is optimized. This task of target optimization requires a computational solution for the following constrained optimization problem

$$\min_{\{x_2,x_3,x_4,x_5\}} \{||y_T - a_1z_1 - a_2x_2 - a_3x_3 - a_4x_4 - a_5x_5 - c||^2 + \mu\sum_{\tau=2}^{5}||x_i - x_i(m-1)||^2\} \tag{20}$$

Subject to $\mathrm{LCL}_i \le x_i \le \mathrm{UCL}_i, i = 2,3,4,5$

$y_{\mathrm{L}} \le a_1z_1 + a_2x_2 + a_3x_3 + a_4x_4 + a_5x_5 + c \le y_{\mathrm{U}}$

where y_{T} is the target of transistor drive current leakage y, and y_{L} and y_{U} are the lower and upper control limits.

In this problem, z_1 is fixed and x_2, x_3, x_4, x_5 are optimization variables. After the completion of the second process step, their targets will be re-computed again using a quadratic programming similar to the one above except that z_1 and z_2 are fixed and x_3, x_4, x_5 are optimization variables. At the third step, z_1, z_2 and z_3 are fixed and we optimize the target of x_4 and x_5.

In the Monte Carlo simulation, the measured output from each completed process step is assumed to have a uniform distribution. The upper and lower limits of the uniform distribution are determined by FabData4015. Using 300 different random initial states, we found that when the first step at oxidation is off the target, the target optimization for the rest three process steps is able to make a 100% correction so that the final product still meet its performance target. If the first and second steps at oxidation and lithography are off the targets, then in the next 2 steps we can only have 45% chance in average to meet the final target if we use the etch process and the ion implantation to compensate the deviation in early steps. In the case that we have finished etching process step, we can only have 17% chance to reach the final performance goal. When we have finished all four steps, the chance to meet the performance goal is 4% that is almost equivalent to zero within the statistical error. This result shows that target optimization at the photolithography or upstream process steps are significantly more efficient than the optimization taking at downstream process steps. The simulation result is summarized in Table 1. This result also shows that the first two steps are very important. Otherwise, we will have only 45% or less chance to meet the goal of leakage current.

Table 1
Possibility to make a correction after each step

Current step	Possibility of complete correction
Step 1	∼ 100%
Step 2	∼ 45%
Step 3	∼ 17%
Step 4	∼ 4%

At the tool selection tier, we assume a manufacturing line with ten tools at each process step. We also assume that all tools are available and the difference of tools in throughput time is small enough to be negligible. The major reason to have these assumptions is that selecting tool with best throughput time is a much more straightforward task than performance based tool selection in Eq. (14). As long as we plug in the throughput time in Eq. (14), the final tool selection can be achieved. So in our simulations, we focus on performance based tool selection.

Let us take step 2, the lithography step, as an example in the tool selection procedure. The measurement x_1 from gate oxidation is treated as an input. Based on the range of x_1, the output error of each tool varies. The tool selection is calculated using Method 2, empirical data based method, introduced in Section 3, i.e. the tool with the highest probability of producing a satisfied output is selected. In the Monte Carlo simulation, we randomly assign 30 different equipment conditions for the set of 10 tools using the range and probability distribution from FabData4015.

For each set of equipment conditions, we run a simulation of 300 lots. Then, we compare the performance output y of the products using optimal tool selection with the performance of products using random tool selection. The calculation of y uses the model Eq. (19) developed from FabData4015.

In Fig. 3, the x-axis represents the 30 randomly assigned equipment conditions; and the y-axis represents the percentage of the lots, from a simulation of 300 runs, for which the leakage current y at the end of the four operation steps is between the limits y_L and y_U, which are set by 1.5 times the standard deviation of the corresponding data from FabData4015. The upper curve is the result with tool selection and the lower curve is the result without tool selection. Fig. 3 shows that, with tool selection, about 20% (95–75%) more lots fall inside the control limits than the operation without tool selection. Please note that the percentages 75% and 95% are based on simulation data that have some random equipment conditions. In real manufacturing process, we should have much better performance to reach control limits.

In this paper, we omit the R2R control simulations with optimized targets from the target optimization tier on a selected best tool, because the R2R control results are well known in the published literature. Readers are referred to [10] on proper selection of weights for dealing with model mismatch and disturbances. As always, during the implementation of a R2R controller, many engineering issues must be resolved. For instance, the metrology time and process time have to be synchronized to determine the lot order. In addition, a deadband is usually used to minimize the change of control signal such as an etch time in etch process or a setup of dosage and focus on lithography scanners when the output is in a tight range. Erroneous metrology readings must also be screened out. The calibration should be applied to measurements after preventive maintenance activities.

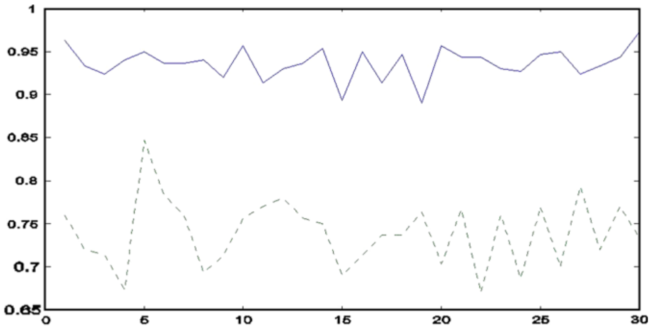

Fig. 3. Results with and without tool selection.

6. Conclusions

This paper presented the investigative work of the 3-tier cooperative control architecture for multi-step processes with multiple tools. The proposed architecture includes: target optimization, tool selection, and tool level control. Monte Carlo simulations on the target optimization and tool selection are carried out for hundreds of random states at each process step. Models, limits, and statistical characteristics in the simulations are based on a large data set collected from real manufacturing.

This study reveals some significant advantages of the cooperative control method over traditional individual tool controllers. The simulations based on the FabData4015 show that the output error of a process step can be corrected under a cooperative controller by as much as 45%∼100%, depending on the number of process steps available for the cooperative control. The result shows that target optimization at the photolithography or upstream process steps are significantly more efficient than the optimization taking at downstream process steps. Meanwhile, the middle tier of the controller, tool selection, is able to improve the probability of achieving satisfied output from a processing step. In the simulations with ten tools at each step, the optimization based tool selection tier is able to gain in average 20% improvement in the number of lots falling inside the tolerant limits than randomly selecting a tool. The 3-tier cooperative control architecture has shown a promising approach for the process control and yield improvement in semiconductor manufacturing.

Although the concept of cooperative control is not difficult to accept, it is a challenging task to make a fab-wide transformation from the current process control paradigm based on individual tool controls into a new paradigm of cooperative control. On the other hand, with the advance of computer aided manufacturing execution systems and the capability of massive manufacturing data farms, it is logic and advantageous to develop globally cooperative control methods that make effective use of the computation and communication capabilities, as well as massively available real-time manufacturing execution data.

References

[1] J.S. Qin, From chemical process control to semiconductor manufacturing control, in: AEC/APC Symposium, Snowbird, Utah, 2002.

[2] P. Raulefs, Z. Mao, F. Wang, Towards fab wide control systems, in: AEC/APC Symposium, Westminster, Colorado, 2004.

[3] K. Chamness, D. Kadosh, R. Good, Implementation of optimized MPC-based supervisory control in a semiconductor manufacturing environment, in: European AECAPC Symposium, Dublin, Ireland, 2005.

[4] S.J. Qin, G. Cherry, R. Good, J. Wang, C.A. Harrison, Semiconductor manufacturing process control and monitoring: A fab-wide framework, Journal of Process Control 16 (3) (2006) 179–191.

[5] R. Murphey, P.M. Pardalos, Cooperative Control and Optimization, Springer, 2002.

[6] W. Kang, Z. Mao, An adaptive model for the control of critical dimension in photolithography process, in: Proceedings of 43rd IEEE Conference of Decision and Control, 2004, pp. 4231–4236.

[7] J. Moyne, E. Castillo, A.M. Hurwitz, Run-to-Run Control in Semiconductor Manufacturing, CRC Press, 2001.

[8] G.G. Box, M. Jenkins, G.C. Reinsel, Time Series Analysis Forecasting and Control, Prentice-Hall International, Inc., 1994.

[9] W. Kang, Z. Mao, Robust control of lithographic process in semiconductor manufacturing, in: Proceedings of SPIE on data Analysis and Modeling for Process Control, 2005, pp. 29–36.

[10] Z. Mao, W. Kang, Benchmark study of run-to-run controllers for the lithographic control of the critical dimension, Journal 0f Micro/Nanolithography MEMS MOEMS 6 (2007).