Pheromone Propagation Controller: The Linkage of Swarm Intelligence and Advanced Process Control

Der-Shui Lee and An-Chen Lee, *Member, IEEE*

***Abstract*—Statistical process control (SPC) is traditionally used in advanced process control (APC). However, SPC, which treats measurements as a series of isolated statistical data, employs different methods to deal with different problems. In this paper, we present a new perspective on process control, which treats the intercepts of the process in different runs as a social insect colony. Our novel algorithm, called the pheromone propagation controller (PPC), is a meta-heuristic method based on the assumption that the intercepts of the linear regression model have their own behavior and affect others nearby on different runs. The pheromone basket is an environment initially filled with intercepts, and then the "intercepts pheromones" in the basket propagate according to the modified digital pheromone infrastructure. After propagation, the intercept in the next run can be forecast by extrapolating the last two entities of the pheromone basket. Consequently, a revised process recipe can be obtained from the forecast intercepts and the linear regression model. We also propose a workable scheme for adaptively tuning the PPC propagation parameter. We discuss the PPC stability region and the strategy for tuning the propagation parameter as well as the effects of size of pheromone basket, model mismatch on the performance. Our simulation results show that the standard deviation and the mean square error for PPC, whether fixed or self-tuning, are more consistent than that of the EWMA, the predictor corrector control (PCC), and the double EWMA for five types of anthropogenic disturbance. We also examined a hybrid disturbance obtained from semiconductor fabrication. When system drifts, the PPC was superior to the other candidate controllers for all values of the PPC propagation parameters and weightings of the other controllers, whether fixed or self-tuning.**

***Index Terms*—Digital pheromone infrastructure, pheromone basket, pheromone propagation controller, process control, swarm intelligence.**

I. Introduction

IN PROCESS control, combining statistical process control (SPC) and feedback control [1] is a popular technique. Quin *et al.* [2] divided semiconductor manufacturing process control into four levels: equipment control, run-to-run control, island control, and fab-wide control. The lowest level is equipment-level control, which holds the desired parameters of tools. Run-to-run control adjusts the recipe slightly based on in-line measurements to even out disturbances. Island control shares information among tools to achieve tool matching and feed-forward/feedback control. The highest level is fab-wide control, which optimizes the desired electrical properties by adjusting the target of the lower level controller. This study addresses advanced process control (APC), which is a run-to-run control.

The exponentially weighted moving average (EWMA) controller is widely used in semiconductor APC. EWMA weighs past data with an exponential discount factor [3]. The stability and sensitivity of EWMA have been analyzed [4]–[6] and some attempts have been made to solve the problem of tuning the discount factor of the EWMA controller in different ways [7]–[11]. Predictor corrector control (PCC) and double EWMA control have been proposed to improve the performance of EWMA when dealing with drifting processes [12], [13]. The stability of double EWMA and multiple-input multiple-output (MIMO) double EWMA has been demonstrated [14], [15], and the weightings for the double EWMA controller have been tuned using different methods [16], [17]. The initial intercept iteratively adjusted (IIIA) controller [18] is used to modify double EWMA with different update procedures to set initial intercepts under drifting disturbance conditions. However, the EWMA-based solvers reject only specific types of disturbance and cannot adapt to a complex environment.

In addition to EWMA and double EWMA, artificial neural networks (ANNs) [19] map the relationship between input and output directly. ANNs, ant colony optimization (ACO), and data mining have been integrated to illustrate the "black box" of ANNs in chemical mechanical polishing (CMP) processes [20]. However, ANNs require many parameters in a suitably constructed network with appropriate training data. The recursive least square (RLS) technique [21] can model a constant mean and a linear trend or random walk for online estimation, but the RLS controller may be unstable if the system gain varies with time.

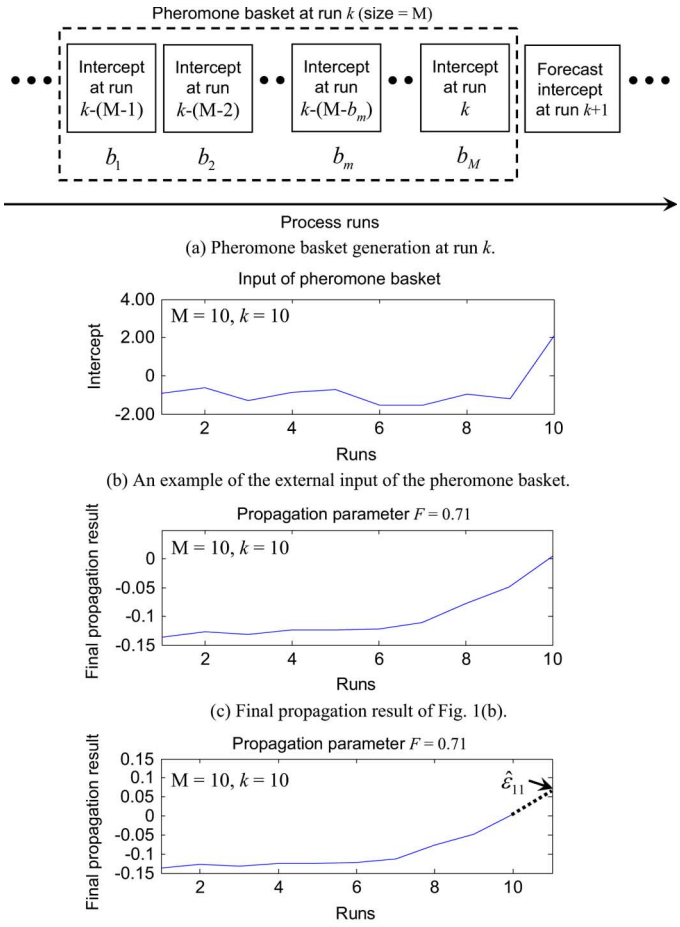
In this paper, we describe an algorithm that adapts easily to complicated disturbances and the uncertainty of the linear regression model. The concept comes from the observation that current errors are caused by previous errors, and the symptoms of future errors come from present errors in the real world. The new algorithm is a meta-heuristic method, which uses swarm intelligence by assuming that each intercept has its own behavior that affects nearby intercepts at different runs. Specifically, swarm algorithms or devices are inspired by the collective behaviors of social insect colonies and other animal societies [22]. ACO [23], for example, is motivated by the behavior of ants in finding paths from their colony to food. Particle swarm optimization (PSO) [24] is inspired by the social behavior of organisms such as birds in a flock or fish in a school. Stochastic

Manuscript received January 30, 2008; revised December 25, 2008; accepted May 04, 2009. First published July 07, 2009; current version published August 05, 2009.

The authors are with the Department of Mechanical Engineering, National Chiao-Tung University, Hsinchu City, Taiwan (e-mail: aclee@mail.nctu.edu.tw).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSM.2009.2024865



(a) Pheromone basket generation at run k.

(b) An example of the external input of the pheromone basket.

(c) Final propagation result of Fig. 1(b).

(d) Forecast intercept at run k+1, $\hat{\varepsilon}_{k+1}$, by extrapolating the last two entities of the final propagation result.

Fig. 1. Concept of the pheromone propagation controller (PPC).

diffusion search (SDS) [25] employs the tandem calling mechanism of ants to perform cheap, partial evaluations of a candidate solution to a search problem. In the semiconductor industry, Jiang *et al.* employed pheromone rules of ACO to solve dynamic scheduling problems for a fabrication line [26].

This paper presents the pheromone propagation controller (PPC) based on the digital pheromone infrastructure [27], which was inspired by the chemical dynamics of pheromone transition and was recently used to describe the decentralized self-organizing behavior of unmanned aerial vehicles (UAVs) and battlefield tactics [27]–[31]. However, the digital pheromone infrastructure must be modified to avoid the end effect. Under the modified digital pheromone infrastructure, the intercepts of a linear regression model in different runs are modeled as a social insect colony. The interaction among intercepts is modeled by a propagation mechanism, which means that an intercept affects others nearby.

Fig. 1 shows the overview of PPC. In Fig. 1(a), the pheromone basket is a moving window that is initially filled with intercepts or pheromones; b_m is the coordination within the pheromone basket and b_m can map to $k-(\mathrm{M}-b_m)$ run. The shape of the pheromone basket in Fig. 1(a) is a single-dimensional line, but could be different for other specific applications. Fig. 1(b) illustrates an example of the input of pheromone basket. Then, the intercepts in the pheromone basket propagate themselves into a steady state according to the modified digital pheromone infrastructure, which will be described in Section II. The final propagation result is shown in Fig. 1(c), which reflects the tendency of the external inputs (or intercepts) in the specific pheromone basket. Fig. 1(d) shows that the intercept for the next run can be forecast by extrapolating the last two entities of the final propagation result in the pheromone basket. Particularly, Figs. 1(a)–1(d) are all at the same time stamp. Finally, the process recipe for the next run can be obtained by the forecast intercept and the process model. We also propose a workable scheme for adaptively tuning the propagation parameter in Section III-F.

We conducted simulations to compare the performance of PPC with EWMA, PCC, and double EWMA. Five types of anthropogenic disturbance and semiconductor fabrication data were used as disturbances in the simulations with three performance indices (average, standard deviation and mean square error) to evaluate the performance. The controller parameters such as the PPC propagation parameter and the weights of the other controllers were fixed and were obtained from historical data with minimum sum square error. The proposed self-tuning PPC, the self-tuning EWMA [9], and the self-tuning PCC [17] were also included in the simulations.

The rest of this paper is organized as follows. Section II explains our modifications to the digital pheromone infrastructure. Section III illustrates the PPC structure including parameter tuning to improve performance. Section IV analyzes the PPC stability region and compares the controller structure of PPC with other controllers. Section V shows the simulation results for the proposed controller. The final section is the conclusion and describes related areas for future work.

II. The Modified Digital Pheromone Infrastructure for PPC

In nature, pheromone, a chemical substance, is released by an insect or animal which causes another individual of the same species to react. When pheromone is released, it evaporates from one position and propagates itself by wind. So, pheromone has different states and transition dynamics. The digital pheromone infrastructure [27] imitated the mechanism of pheromone by modeling the environment, states of pheromone and transition dynamics of a pheromone. While the natural pheromone is powered by wind and time, the digital pheromone infrastructure models the transition dynamics by transition parameters (evaporation and propagation parameters), transition functions where iterations for pheromone propagation are practiced. Therefore, as compared the power of natural with digital pheromones, wind maps to transition parameters and transition functions; time maps to iterations.

This section introduces the modified digital pheromone infrastructure. We define pheromone basket, mapping to the environment, as a moving time window in the manufacturing process. The pheromone states and transition parameters are the same as defined in the digital pheromone infrastructure. Because of the end effect and the energy balance in our one-dimensional pheromone basket, we modified the transition functions in [27].

A. Pheromone Basket

The pheromone basket is the pheromone propagation environment. The environment of the modified digital pheromone infrastructure is a tuple $\langle B, N\rangle$, where B is a finite set of positions $b_m \in B : m = 1, 2\ldots,$ M within the pheromone basket, M is the size of the pheromone basket, and b_{M} maps to the current run and b_1 maps to M runs beforehand. In addition, $N(b_m) \subseteq B$ is a finite set of neighbors of b_m and $|N(b_m)|$ is the size of $N(b_m)$. In addition, the modified pheromone infrastructure assumes that the propagation relationship between $N(b_m)$ and b_m is irreflective, which means that b_m will accept propagation inputs from $N(b_m)$ without preconditions.

From Fig. 1(a), the pheromone basket fills with "intercept pheromones" initially. Since the "intercept pheromones" propagate themselves into a steady state at a time stamp from Fig. 1(b) to Fig. 1(c), the measurement of the intercepts can be treated as the external impulse input to the pheromone basket. The external input is a finite set $\mathbf{R}(k,i) = \{r(k,b_m,i) \in (-L, L) : m = 1, 2, \ldots, \mathrm{M}\}$ where $k = 1, 2, \ldots,$ is the run number of the manufacturing process, $i \in \aleph$ is the number of iterations (or propagations) in the transition functions, and $L \in \aleph$ is the global limit of the external inputs in the environment $\langle B, N\rangle$. Note that iteration is executed to update transition functions but not means the process. We use the notation $\aleph$ for the set of natural numbers and $\Re$ for the set of real numbers. Since the external input is the initial condition for launching transition functions of pheromone propagation, $r(k,b_m,0)$ maps to the intercept at the $k-(\mathrm{M}-b_m)^{\mathrm{th}}$ run on the process and $r(k,b_m,i)$ is 0 when i is larger than 0.

B. Pheromone States

The states in the pheromone basket $\langle B, N\rangle$ are $\mathbf{Q}(k,i)$ and $\mathbf{S}(k,i)$, where $\mathbf{Q}(k,i) = \{q(k,b_m,i) \in \Re : m = 1, 2, \ldots, \mathrm{M}\}$ is a finite set of the propagated inputs at run k and iteration i, and $\mathbf{S}(k,i) = \{s(k,b_m,i) \in \Re : m = 1, 2, \ldots, \mathrm{M}\}$ is a finite set of the aggregated pheromones at run k and iteration i. Therefore, $q(k,b_m,i)$ is regarded as the propagated input from $N(b_m)$ to b_m at iteration i and run k. Similarly, $s(k,b_m,i)$ is regarded as the aggregated pheromone of b_m at iteration i and run k. In PPC, $\mathbf{Q}(k,i)$ refers to the effectiveness of one intercept affecting its nearby intercepts; $\mathbf{S}(k,i)$ refers aggregation result which involves the influence of external input $\mathbf{R}(k,i)$ and propagation effect $\mathbf{Q}(k,i)$.

In addition, $q(k,b_m,0) = 0$ and $s(k,b_m,0) = 0$ are assumed to be the default initial conditions. While launching transition functions as shown in the following sections, $\mathbf{Q}(k,i)$ disseminates pheromones and $\mathbf{S}(k,i)$ aggregates pheromones simultaneously.

C. Pheromone Transition Parameters

Two transition parameters of the digital pheromone infrastructure [27] are the evaporation parameter $E \in (0, 1]$ and the propagation parameter $F \in [0, 1)$. In PPC, the propagation parameter F describes the effect of a measurement on other nearby measurements, and the evaporation parameter E indicates the weakening property of the monitored data. In other words, E indicates that the importance of the measurement data will "evaporate" with time. Because measurements in a short period can be treated as a nondissipative system, the modified pheromone infrastructure uses $E = 1$ in a small pheromone basket (Proof is given in Appendix A).

D. Transition Functions

While defining the pheromone basket $\langle B, N\rangle$, the parameters E and F, the external input $\mathbf{R}(k,i)$, state $\mathbf{Q}(k,i)$, and $\mathbf{S}(k,i)$, Brueckner introduced two transition functions to describe the propagated inputs $\mathbf{Q}(k,i)$ and the aggregated pheromone $\mathbf{S}(k,i)$ [27]. The transition function of the propagated inputs $q(k,b_m,i) \in \mathbf{Q}(k,i)$ is

$$q(k,b_m,i+1) = \sum_{b_{m'} \in N(b_m)} \frac{F}{|N(b_{m'})|} \times (r(k,b_{m'},i) + q(k,b_{m'},i)) \quad \forall m = 1, 2, \ldots, \mathrm{M} \tag{1}$$

where $b_{m'} \in N(b_m)$ indicates that $b_{m'}$ is the neighbor of b_m, $|N(b_{m'})|$ is the number of neighbors of $b_{m'}$, and $F/|N(b_{m'})|$ indicates that $b_{m'}$ affects its neighbors equally. Because the shape of the proposed pheromone basket is a one-dimensional line, the basket has only one neighbor at the two ends of the basket and two neighbors in the other positions. Thus, (1) becomes (2), found at the bottom of the page.

In addition, the transition of the aggregated pheromone $s(k,b_m,i) \in \mathbf{S}(k,i)$ is defined as [27]

$$s(k,b_m,i+1) = E \times s(k,b_m,i) + r(k,b_m,i) + q(k,b_m,i),\ m = 1, 2, \ldots, \mathrm{M}. \tag{3}$$

Because b_m propagates the F ratio of $[r(k,b_m,i) + q(k,b_m,i)]$ out to $b_{m'}$ in (2), the F ratio of $[r(k,b_m,i) + q(k,b_m,i)]$ should be subtracted from (3) to maintain its balance. Thus, we modify (3) to be

$$s(k,b_m,i+1) = E \times s(k,b_m,i) + (1-F)\big[r(k,b_m,i) + q(k,b_m,i)\big]\ \forall m = 1, 2, \ldots, \mathrm{M}. \tag{4}$$

With E is 1, (4) becomes

$$s(k,b_m,i+1) = s(k,b_m,i) + (1-F)\big[r(k,b_m,i) + q(k,b_m,i)\big]\ \forall m = 1, 2, \ldots, \mathrm{M}. \tag{5}$$

$$q(k,b_m,i+1) = \begin{cases} \frac{F}{2}(r(k,b_{m+1},i) + q(k,b_{m+1},i)), & \text{if } m = 1 \\ \frac{F}{1}(r(k,b_{m-1},i) + q(k,b_{m-1},i)) + \frac{F}{2}(r(t,b_{m+1},i) + q(k,b_{m+1},i)), & \text{if } m = 2 \\ \frac{F}{2}(r(k,b_{m-1},i) + q(k,b_{m-1},i)) + \frac{F}{2}(r(k,b_{m+1},i) + q(k,b_{m+1},i)), & \text{if } 2 < m < \mathrm{M}-1 \\ \frac{F}{2}(r(k,b_{m-1},i) + q(k,b_{m-1},i)) + \frac{F}{1}(r(k,b_{m+1},i) + q(k,b_{m+1},i)), & \text{if } m = \mathrm{M}-1 \\ \frac{F}{2}(r(k,b_{m-1},i) + q(k,b_{m-1},i)), & \text{if } m = \mathrm{M} \end{cases} \tag{2}$$

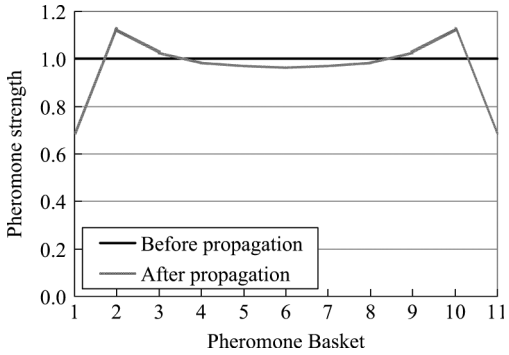


Fig. 2. Boundary effect of pheromone propagation without a modifier.

Moreover, the profile of the final transition result of $\mathbf{S}(k,\infty)$ should be flat when the pattern of the external input $\mathbf{R}(k,0)$ is flat. Unfortunately, the profile of the final propagation result obtained from (2) and (5) is not flat due to the end effect as shown in Fig. 2. To overcome this, we modify the propagation-out ratio in the two extremities from $F/|N(b_{m'})|$ to $F/(2-F)$ (See the proof in Appendix B) when the shape of the pheromone basket is a line and (2) and (5) become(6) and (7), found at the bottom of the page.

Fig. 3 illustrates the concept of (6) and (7). The final propagation results of (6) and (7) can be obtained analytically by (A9) in the Appendix C. For example, if M is 6, the final propagation results for $b_m = 5$ and $b_m = 6$ are (8) and (9), found at the bottom of the page.

III. Pheromone Propagation Controller

Fig. 4 illustrates the PPC block diagram for APC according to the modified pheromone infrastructure described in Section II. The PPC can be separated into four modules: plant, pheromone basket generator, intercept predictor, and recipe generator. This section describes each of these modules. In addition, a workable propagation parameter F tuner in the intercept predictor is also proposed.

A. Plant

The plant is the process model in a simulation or in a real system, which can be obtained by the linear regression model. Because fabrication recipes usually have several parameters, the demonstrated MISO plant, which has j recipes (inputs), is

$$Y_k = \alpha + \boldsymbol{\beta}_k \cdot \mathbf{X}_k + \upsilon_k \quad k = 0, 1, 2 \ldots, \tag{10}$$

where

$Y_k \in \Re$	measurement at the end of run k;
$\mathbf{X}_k \in \Re$	recipes (inputs) of run k and $\mathbf{X}_k$ is a $j \times 1$ matrix;
$\alpha \in \Re$	initial intercept of the process;
$\boldsymbol{\beta}_k \in \Re$	system gain of run k and $\boldsymbol{\beta}_k$ is a $1 \times j$ matrix;
$\upsilon_k \in \Re$	disturbance, which includes noise and uncontrolled terms of run k.

In (10), the intercept of run k, ε_k, becomes

$$\varepsilon_k = \alpha + \upsilon_k \quad k = 0, 1, 2 \ldots. \tag{11}$$

$$q(k,b_m,i+1) = \begin{cases} \frac{F}{2}(r(k,b_{m+1},i)+q(k,b_{m+1},i)), & \text{if } m=1 \\ \frac{F}{2-F}(r(k,b_{m-1},i)+q(k,b_{m-1},i)) + \frac{F}{2}(r(k,b_{m+1},i)+q(k,b_{m+1},i)), & \text{if } m=2 \\ \frac{F}{2}(r(k,b_{m-1},i)+q(k,b_{m-1},i)) + \frac{F}{2}(r(k,b_{m+1},i)+q(k,b_{m+1},i)), & \text{if } 2<m<\mathrm{M}-1 \\ \frac{F}{2}(r(k,b_{m-1},i)+q(k,b_{m-1},i)) + \frac{F}{2-F}(r(k,b_{m+1},i)+q(k,b_{m+1},i)), & \text{if } m=\mathrm{M}-1 \\ \frac{F}{2}(r(k,b_{m-1},i)+q(k,b_{m-1},i)), & \text{if } m=\mathrm{M} \end{cases} \tag{6}$$

$$s(k,b_m,i+1) = \begin{cases} s(k,b_m,i) + \left(1-\frac{F}{2-F}\right)(r(k,b_m,i)+q(k,b_m,i)), & \text{if } m=1,\mathrm{M} \\ s(k,b_m,i) + (1-F)(r(k,b_m,i)+q(k,b_m,i)), & \text{if } 1<m<\mathrm{M} \end{cases} \tag{7}$$

$$s(k,5,\infty) = \frac{1}{2(3F^2-4)(F+2)} \begin{bmatrix} -F^4 \\ F^4-2F^3 \\ F^4+2F^3-4F^2 \\ -F^4+4F^3+4F^2-8F \\ -F^4-4F^3+12F^2+8F-16 \\ F^4+6F^3-8F \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} r(k,1,0) \\ r(k,2,0) \\ r(k,3,0) \\ r(k,4,0) \\ r(k,5,0) \\ r(k,6,0) \end{bmatrix} \tag{8}$$

$$s(k,6,\infty) = \frac{F}{2(3F^2-4)(F^2-4)} \begin{bmatrix} -F^4 \\ -F^4+2F^3 \\ -F^4-2F^3+4F^2 \\ F^4-4F^3-4F^2+8F \\ F^4+4F^3-12F^2-8F+16 \\ -F^4+6F^3+12F^2-32F-16 \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} r(k,1,0) \\ r(k,2,0) \\ r(k,3,0) \\ r(k,4,0) \\ r(k,5,0) \\ r(k,6,0) \end{bmatrix} \tag{9}$$

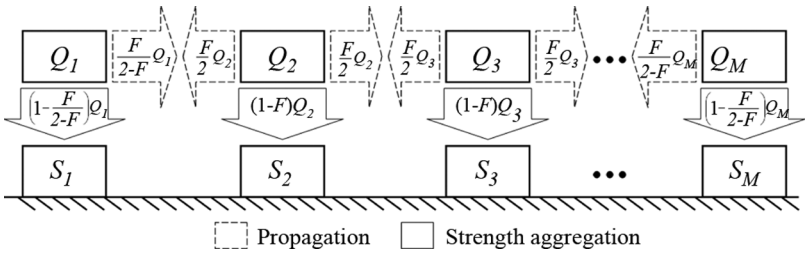


Fig. 3. Transitions with propagation parameter F and size M in a pheromone basket.

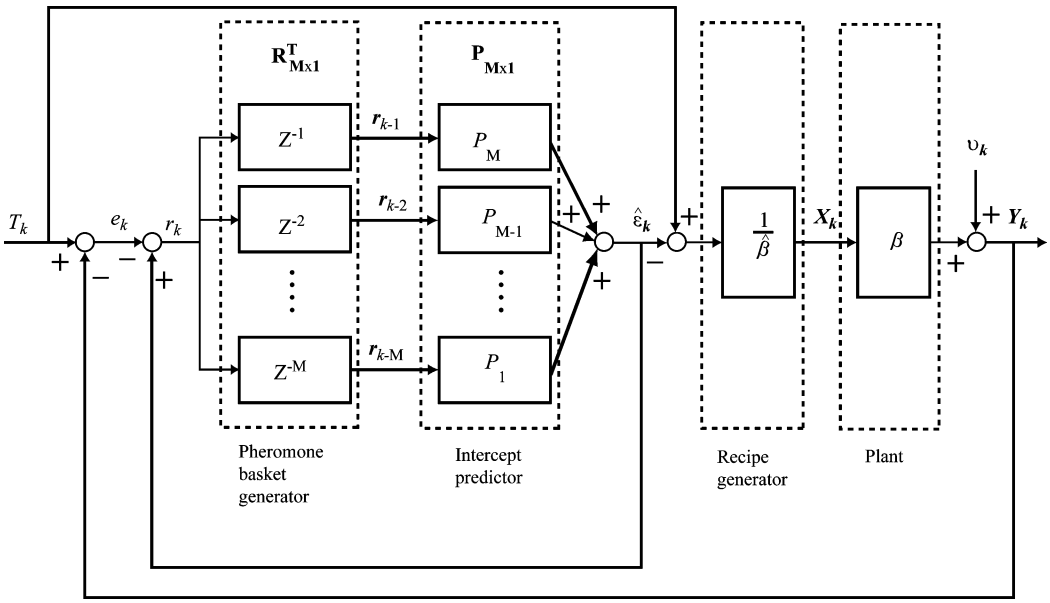


Fig. 4. Block diagram of a pheromone propagation controller.

Furthermore, this study considers two cases of system gain:

i) The fixed system gain with bias:

$$\boldsymbol{\beta}_k = \bar{\boldsymbol{\beta}} \quad k = 0, 1, 2 \ldots \tag{12}$$

where $\bar{\boldsymbol{\beta}}$ is the fixed shift of the process gain and size of $\bar{\boldsymbol{\beta}}$ is $1 \times j$.

ii) The system gain shifts and drifts simultaneously,

$$\boldsymbol{\beta}_k = \bar{\boldsymbol{\beta}} + \gamma \cdot k \quad k = 0, 1, 2 \ldots \tag{13}$$

where γ is the constant drifting rate of the process gain.

B. Pheromone Basket Generator

The concept of the pheromone basket is the most important part of the PPC. Treating the intercepts of the linear regression model for a sequence of runs as external inputs to the pheromone basket links process control with the modified digital pheromone infrastructure. The pheromone basket is initially filled with intercepts, which map the external input $\mathbf{R}(k, i)$ to the modified digital pheromone infrastructure. Then, states $\mathbf{S}(k, i)$ and $\mathbf{Q}(k, i)$ are updated simultaneously by the transition functions of the modified infrastructure. Because the intercept of run k can be obtained by the forecast intercept in the previous run and the error in the current run, the external input of b_m at run k becomes

$$\begin{aligned} r(k, b_m, 0) &= \hat{\varepsilon}_{k-(\mathrm{M}-m)} - e_{k-(\mathrm{M}-m)} \\ &\quad \forall m = 1, \ldots, \mathrm{M}, \text{ if } k > \mathrm{M} \\ r(k, b_m, 0) &= \hat{\varepsilon}_m - e_m \\ &\quad \forall m = 1, \ldots, k, \text{ if } k \leq \mathrm{M} \end{aligned} \tag{14}$$

where $\hat{\varepsilon}_k \in \Re$ is the forecast intercept of run k and $e_k \in \Re$ is the error of run k.

Equation (14) shows that the pheromone basket generator is a series of intercepts with a sequence of delays. The external input at b_m of run k maps the intercept of the previous $(M - m)$ runs.

In addition, the error of run k in (14) is defined as

$$e_k = T - Y_k \tag{15}$$

where $T \in \Re$ is a fixed process target.

C. Intercept Predictor

The intercept predictor propagates of the pheromone basket and the extrapolation of the last two entities of the aggregated pheromone $\mathbf{S}(k, \infty)$ to achieve one-step-ahead forecasting for the next run. In Fig. 4, the input to the intercept predictor function block is a series of intercepts with a sequence of time delays

and propagation parameter F, and the output of the intercept predictor is the forecast intercept at the next run.

The propagation of the pheromone basket obeys the modified transition functions in Section II. In addition, the final propagation result reflects the trend of the specific external input at run k. Then, the forecast intercept at run $k+1$ can be inferred simply by extrapolating the last two entities of $\mathbf{S}(k,\infty)$:

$$\hat{\varepsilon}_{k+1} = s(k, b_{\mathrm{M}}, \infty) - s(k, b_{\mathrm{M}-1}, \infty) + s(k, b_{\mathrm{M}}, \infty). \quad (16)$$

Using (A9), the analytic solution of (16) yields

$$\hat{\varepsilon}_{k+1} = 2 \cdot v_{\mathrm{M}}(k) - v_{\mathrm{M}-1}(k) = \mathbf{R}^{\mathbf{T}}(k,0) \cdot \mathbf{P}(k) \quad (17)$$

where

$$\mathbf{R}(k,i) = \begin{bmatrix} r(k,b_1,i) \\ r(k,b_2,i) \\ \vdots \\ r(k,b_m,i) \\ \vdots \\ r(k,b_{\mathrm{M}-1},i) \\ r(k,b_{\mathrm{M}},i) \end{bmatrix}_{\mathrm{M}\times 1}$$

is the external inputs of all of the positions at the pheromone basket. In addition, $\mathbf{P}(k)$, which varies with the size of the pheromone basket M, is an algebraic expression of F. For example, if M is 6, substituting (8) and (9) into (17) yields (18), found at the bottom of the page.

Figs. 5(a)–(c) shows an example of different forecast intercepts with an external ramp input and different propagation parameters F for a pheromone basket size $\mathrm{M} = 10$. Fig. 5(a) shows the external ramp inputs, and Fig. 5(b) illustrates the final propagation results of Fig. 5(a) with different values of F. Fig. 5(c) is the extrapolated result of Fig. 5(b). If F is 0, the final propagation result is directly equal to one of the last two extrapolated entities of the external inputs. However, the extrapolated result approaches the mean of the external inputs as F approaches unity.

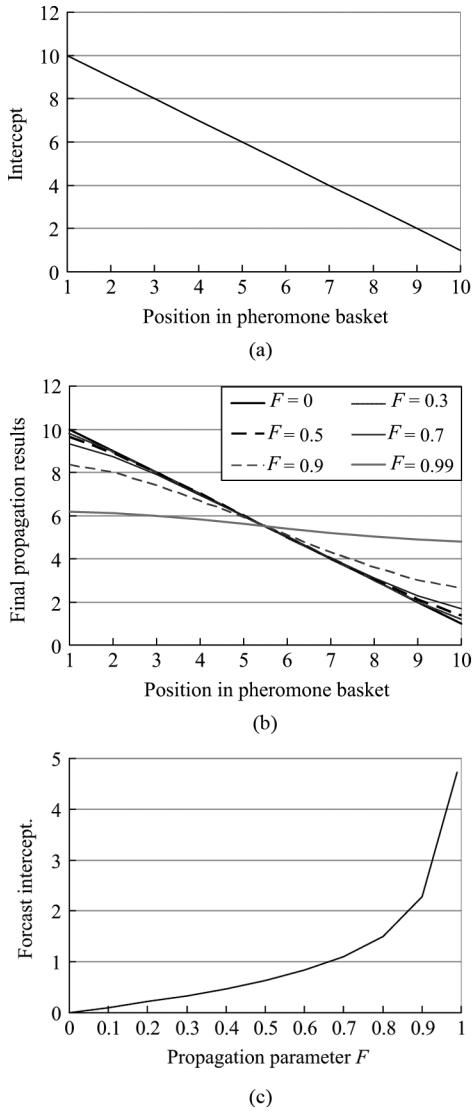


Fig. 5. Forecast intercept for different external inputs and pheromone parameters: (a) External inputs, (b) Final propagation results for different values of F, and (c) Forecast intercepts for different values of F.

D. Recipe Generator

The recipe generator generates the recipe of the process for the next run. We use the linear regression model to produce

$$T = \hat{\alpha} + \hat{\boldsymbol{\beta}} \cdot \mathbf{X}_{k+1} + \hat{\varepsilon}_{k+1} \quad (19)$$

where $\mathbf{X}_{k+1} \in \Re$, which is a $j \times 1$ matrix, is the recipe (input) of run $k+1$, $\hat{\alpha} \in \Re$ is the estimator of the initial intercept α, and $\hat{\boldsymbol{\beta}} \in \Re$, which is a $1 \times j$ matrix, is the estimator of the system $\boldsymbol{\beta}$ (model gain).

In (19), $\hat{\alpha}$ and $\hat{\boldsymbol{\beta}}$ are obtained from the linear regression model by the off-line DOE and are not equal to the process parameters α and $\boldsymbol{\beta}$. The parameter $\hat{\varepsilon}_{k+1}$ comes from the intercept predictor, and T is the given target value. Then the recipe for run $k+1$ is generated by

$$\mathbf{X}_{k+1} = \hat{\boldsymbol{\beta}}^{\mathrm{T}}(\hat{\boldsymbol{\beta}}\hat{\beta}^{\mathrm{T}})^{-1}(T - \hat{\alpha} - \hat{\varepsilon}_{k+1}). \quad (20)$$

$$\mathbf{P}(k) = \begin{bmatrix} p_1(k) \\ p_2(k) \\ p_3(k) \\ p_4(k) \\ p_5(k) \\ p_6(k) \end{bmatrix}_{6\times 1} = \frac{1}{2(3F^4 - 16F^2 + 16)} \begin{bmatrix} 3F^5 - 2F^4 \\ -3F^5 + 8F^4 - 4F^3 \\ -3F^5 - 4F^4 + 16F^3 - 8F^2 \\ 3F^5 - 14F^4 - 4F^3 + 32F^2 - 16F \\ 3F^5 + 10F^4 - 44F^3 + 64F - 32 \\ -3F^5 + 8F^4 + 36F^3 - 56F^2 - 48F + 64 \end{bmatrix}_{6\times 1}. \quad (18)$$

E. Propagation Parameter F Tuner

This section proposes two strategies for tuning the propagation parameter F. The first strategy uses historical data and examines all possible F values to obtain the best fixed propagation parameter, $\bar{F}$, with minimum mean square error:

$$\bar{F} \cong \min_{F} \left(\sum_{k'} e_{k'}^2 \right) \tag{21}$$

where k' is the index of the historical data. Then $\bar{F}$ is used in the testing data.

The second strategy employs $\hat{\varepsilon}_k$ and e_k available from the last run k for adaptively tuning F at run $k+1$, $\hat{F}_{k+1}$. The adjusted propagation parameter $\tilde{F}_k$ is given by

$$\tilde{F}_k \cong \min_{F}(|tmp_F - e_k|) \tag{22}$$

where all possible forecast intercepts tmp_F s are obtained after each run by (A9) and (17) to obtain $\tilde{F}_k$, which can be interpreted as the best parameter for run k.

Since noise will be included in (22), we conduct $\hat{F}_{k+1}$ with a moving average filter to avoid overcorrection and the perturbation of noise:

$$\hat{F}_{k+1} = \frac{1}{f} \sum_{i=0}^{f-1} \tilde{F}_{k-i} \tag{23}$$

where f is the size of the filter and its influence on performance is presented in Section V-A1.

F. Control Procedure

The PPC requires M historical intercepts, where M is the desired size of the pheromone basket. The whole control procedure is presented in the following sequence, including the case when the run number k is less than M at the beginning of the process.

- Step 1) Set the following initial conditions: $\hat{\varepsilon}_1$ and $\hat{\varepsilon}_2$ are zero; input $\mathbf{X}_1$ of run 1 is the initial recipe; if F is self-tuning, let $\tilde{F}_0$, $\tilde{F}_1$, and $\tilde{F}_2$ approach 1.
- Step 2) Get the process error e_1, set $\hat{\varepsilon}_2$ to e_1, and determine the recipe $\mathbf{X}_2$ using (20).
- Step 3) Get the process error e_2 and put e_1, e_2, $\hat{\varepsilon}_1$, and $\hat{\varepsilon}_2$ into (14) to generate external inputs to the pheromone basket for run 2.
- Step 4) Set $\hat{F}_3$ to $\bar{F}$ from (21) when F is fixed and selected by historical data. When F is self-tuning, calculate forecast propagation parameter $\hat{F}_3$ using (23).
- Step 5) Use (A9) with propagation parameter $\hat{F}_3$ to forecast the steady-state value of the aggregated pheromone at run 3, and predict the process intercept $\hat{\varepsilon}_3$ using (17).
- Step 6) Determine the recipe $\mathbf{X}_3$ using (20) and obtain process error e_3.
- Step 7) Put $\hat{\varepsilon}_3$ and e_3 into (14) to generate the external input of the pheromone basket for run 3.
- Step 8) Repeat steps 4–7 by replacing $\hat{\varepsilon}_3$ and $\hat{F}_3$ with $\hat{\varepsilon}_{k+1}$ and $\hat{F}_{k+1}$, respectively, for $k = 3, 4, 5, \ldots$. This produces the recipe $\mathbf{X}_{k+1}$ and the error e_{k+1} at run $k+1$.

If the run k is larger than the desired size of pheromone basket M, update the pheromone basket by (14).

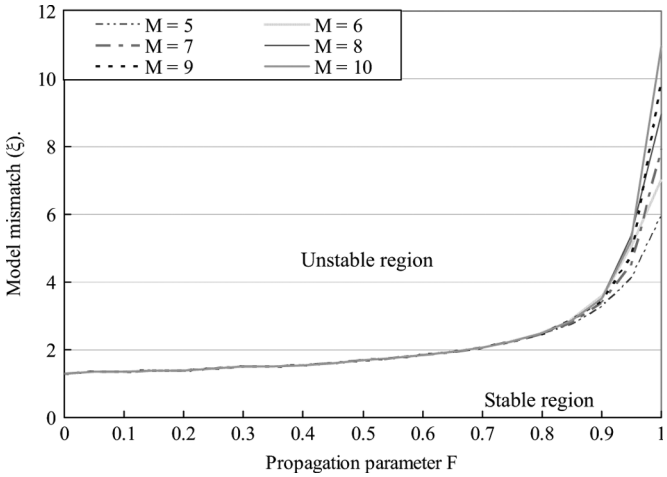


Fig. 6. PPC stability region for different sizes M of the pheromone basket.

IV. Stability Analysis

In this section, we analyze the stability of the SISO PPC. We first derive its transfer function from the block diagram and then discuss the stability region under the conditions of different model mismatch and propagation parameters.

A. Transfer Function

Because the analytic solution of pheromone propagation has been determined, the transfer function of the PPC can be derived from Fig. 4. The figure shows that the pheromone basket generator is modeled by a sequence of integer delays corresponding to a series of intercepts. The intercept predictor uses (17) to forecast the intercept of the next run, and the model gain is obtained using the linear regression model. Thus, the transfer function from target to output can be obtained by Mason's gain formula:

$$G_{\mathrm{TO}}(z) = \frac{\beta}{\hat{\beta} + \mathbf{R}^{\mathbf{T}}\mathbf{P}(\beta - \hat{\beta})} \tag{24}$$

where β and $\hat{\beta}$ are $\boldsymbol{\beta}$ and $\hat{\boldsymbol{\beta}}$ with j is one. In (24), the characteristic function has M poles, which can be used to check the stability region of the PPC as shown in the next section. In addition, (24) and Fig. 4 indicates that PPC is an Mth order controller. The higher order controller has the capacity of eliminating higher order disturbances.

B. Stability Region

The stability region of the PPC with a constant model and process gain can be obtained by checking the pole locations of the characteristic function in (24), where the order of the characteristic function varies with the size of the pheromone basket M. We examined the PPC stability region in terms of model mismatch ($\xi = \beta\hat{\beta}$) with different propagation parameters, F, and the size of the pheromone basket, M. If all of the pole locations are within the unit circle, the point (F, ξ) is a stable point. Fig. 6 shows the stability regions, which are located under each curve, for values of M in the range 5–10. Fig. 6 indicates that the stability regions increase as F increases, and the slight effect of M is observed when F is less than 0.7. As F approaches 1, the maximum stability value will converge to $\mathrm{M}+1$. For example,

TABLE I
CONTROLLER STRUCTURE COMPARISON OF EWMA, PCC, D-EWMA, AND PPC

Controller	Controller parameter	Pole number	Zero number (Zero positions)	Controller type	Allowable stable region for all controller parameters
EWMA	$0 \le W \le 1$	1	1 (1)	An integral controller	$0 < \xi \le 2$
PCC	$0 \le W_1 \le 1$ $0 \le W_2 \le 1$	2	2 (1, 1)	A double integral controller	$0 < \xi \le 1.3$
d-EWMA	$0 \le W_1 \le 1$ $0 \le W_2 \le 1$	2	2 (1, 1)	A double integral controller	$0 < \xi \le 1.3$
PPC	$F = 0$	2	2 (1, 1)	A double integral controller; same as PCC and d-EWMA with w1 = 1 and w2 = 1.	$0 < \xi \le 1.3$
	$0<F<1$	M	M (One of the zeros is 1.)	An M order controller which includes a single integral controller.	
	$F \rightarrow 1$	M	M (One of the zeros is 1.)	An M order controller which includes a single integral controller. The effect of PPC is equal to an M moving average filter of M intercepts.	

if M is 5, the maximum stability value is 6 as F approaches 1. It seems counter-intuitive that a higher order PPC has a larger stability region. Appendix D shows an example that a higher order moving average controller actually has a larger stability region than a lower order one.

C. Controller Comparison

This section compares PPC with EWMA, PCC and d-EWMA controller analytically. Table I lists the number of poles and zeros of the transfer function from disturbance to output, zero positions, controller type and allowable stable region for candidate controllers. A controller can reject step disturbance when zeros of the transfer function include one and reject ramp disturbance when they include two ones.

In Table I, controller type of PPC varies with the propagation parameter F. When F is zero, PPC has only two zeros locating all in one, which is the same as PCC and d-EWMA controller with $W_1 = 1$ and $W_2 = 1$. When F increases from zero, PPC has M zeros, and still has a zero at one. As F approaches to 1, the PPC is equal to a moving average filter of order M. Thus, PPC can deal with Mth order disturbance theoretically when F is not zero. Furthermore, PPC always has a zero at one for every F, which means that PPC is able to reject step disturbance, same characteristics as EWMA. Note that PCC and d-EWMA can reject ramp disturbance with every W_1 and W_2, while PPC can only deal with ramp disturbance when F is zero. In summary, PPC is a scalable Mth order controller and its properties are similar to EWMA, PCC and d-EWMA under certain conditions.

As for the stability region, it varies with the controller parameters. The minimum acceptable stability region of PPC is the same as PCC and d-EWMA, while the maximum acceptable stability region of PPC is smallest among all controllers.

V. SIMULATION RESULTS

This section compares the performance of PPC with EWMA, PCC, and double EWMA when subjected to different types of disturbance. This study used the training data to examine all possible propagation parameters or weightings of other controllers to find a fixed optimal propagation parameter or the weightings of the other controllers using the minimum mean square error. Then the parameter or weightings were applied to testing data. Furthermore, the proposed self-tuning PPC was compared with self-tuning EWMA [9] and self-tuning PCC [17] directly through the testing data. Notably, there has been no literature to date regarding adaptive tuning of the double EWMA controller. Thus, seven candidate controllers were used in our simulation; we applied five types of anthropogenic disturbance and the data from semiconductor fabrication.

A. Anthropogenic Disturbance

The simulation settings with anthropogenic disturbance included the following: the process model (α, β) was (1, 1.5), the 10% offset controller model $(\hat{\alpha}, \hat{\beta})$ was (1.1, 1.65), the initial input was 0, and the process target T was 0, respectively.

Because PPC is a meta-heuristic method, the performance can only be assessed by different cases. This study examined five types of disturbance with $\mathcal{B}$ as the backward shift operator and ω_k as the Gaussian noise and uncontrolled terms of run k:

(a) IMA(1, 1)

$$v_k = v_{k-1} + (1 - \theta_1 \mathcal{B})\omega_k \tag{25}$$

(b) ARMA (1, 1)

$$(1 - \phi_1 \mathcal{B})v_k = (1 - \theta_1 \mathcal{B})\omega_k \tag{26}$$

TABLE II
COEFFICIENTS OF THE ANTHROPOGENIC DISTURBANCE

υ_k	ε_t (Ave, Std)	ϕ_1	ϕ_2	ϕ_3	θ_1
IMA(1, 1)	(0, 1.5)	—	—	—	0.5
ARMA(1, 1)	(0, 1.5)	0.7	—	—	0.28
Random walk	(0, 1.5)	—	—	—	—
AR(3)	(0, 1.5)	0.9	–0.27	0.027	—
ARI(3, 1)	(0, 1.5)	0.9	–0.27	0.027	—

(c) Random walks

$$\upsilon_k = \upsilon_{k-1} + \omega_k \tag{27}$$

(d) AR(3)

$$(1 - \phi_1\mathcal{B} - \phi_2\mathcal{B}^2 - \phi_3\mathcal{B}^3)\upsilon_k = \omega_k \tag{28}$$

(e) ARI(3, 1)

$$(1 - \phi_1\mathcal{B} - \phi_2\mathcal{B}^2 - \phi_3\mathcal{B}^3)(\upsilon_k - \upsilon_{k-1}) = \omega_k. \tag{29}$$

The coefficients for the different types of anthropogenic disturbance are listed in Table II. IMA(1, 1) [5], [6], [8], [11], [13], [17], [21], ARMA(1, 1) [5], [8], [18], Random walk [1], [21] are typical disturbance models for the fabrication data in semiconductor manufacturing and the high order disturbance ARI(3, 1) is also observed in sputter deposition process [32]. Additionally, AR(3) is employed to check the performance of PPC under high order disturbance.

We produced 100 sets of training data and 100 sets of testing data using (25)–(29) with ω_k, $\mathrm{mean} = 0$, and standard deviation $\sigma = 1.5$. Each set of the training and testing data had 100 runs. In addition, this study employs three performance indices to evaluate the performance: average (Ave.), standard deviation (Std.) and mean square errors (MSE) of Y_{100}'s (10) obtained from 100 sets of test data.

1) Influence of PPC Parameters on Output Performance: This section shows the influence of PPC parameters on output performance: 1) the size of pheromone basket M; 2) the standard deviation of noise in (25)–(29); 3) the model mismatch; and 4) the filter size f in (23).

The influence of the size of the pheromone basket is shown in Figs. 7(a)–7(b). In Fig. 7(a), when propagation parameter is fixed, the size of pheromone basket almost has no effect on MSE. In Fig. 7(b), when the propagation parameter is self-tuned, a small size of pheromone basket has benefits in dealing with the higher order disturbance. However, selecting M is a trade-off between stability and performance. One selects $\mathrm{M} = 6$ by the following considerations: (1) The PPC with $\mathrm{M} = 6$ has slightly smaller MSE than that of $\mathrm{M} = 5$ for ARMA(1,1) and IMA(1,1) as shown in Figs. 7(a)–7(b); (2) Stability region of the PPC with $\mathrm{M} = 6$ is larger than that of $\mathrm{M} = 5$ as shown in Fig. 6.

For the minimum variance control [33], with a *priori* information about the disturbance structure, the Std. of the controlled

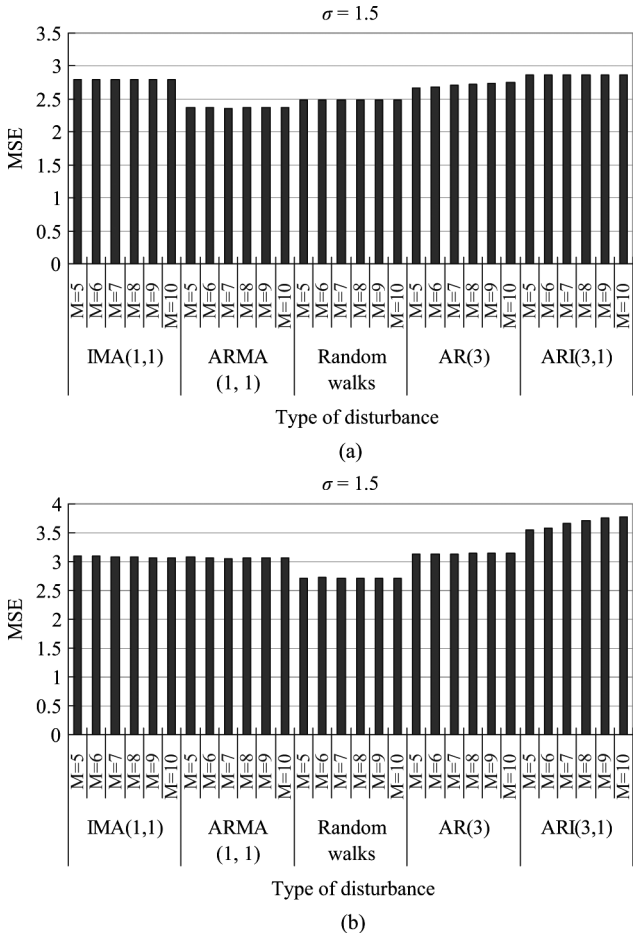


Fig. 7. Comparison of the size of pheromone basket with $\xi = 1.1$: (a) PPC with fixed optimum propagation parameters and (b) Self-tuning PPC.

output is always equal to or larger than σ. The influence of the standard deviation of noise in (25)–(29) on performance is shown in Fig. 8(a)–(b), which shows output Std. is slightly larger than the setting of σ. It also indicates that the performance approaches to that of minimum variance controller, which needs *a priori* information about the disturbance structure.

Fig. 9(a)–(b) examines the influence of different model mismatches within the stability region, where no particular trend can be concluded, but it will increase with the increase of model mismatch for high order disturbances. From our observation, MSE with different model mismatches in Fig. 9(a)–(b) is in form of a parabolic curve when model mismatch is further extended; the minimum MSE is not always at $\alpha = \hat{\alpha} = 1$ for different types of disturbance.

The influence of the filter size f in (23) on performance is shown in Fig. 10, where the size of pheromone basket of self-tuning PPC is 6. It shows that the perturbation of noise will appear when $f = 1$, i.e., no filtering. In addition, the result is slightly different when $f \in 3$ and choosing larger f needs more runs for initialization (it needs $f - 1$ runs in simulation and real case). So, the self-tuning PPC in this study employs $f = 3$ in the following simulations.

2) Performance Comparison of PPC With EWMA, PCC and Double EWMA: The fixed optimal propagation parameter or

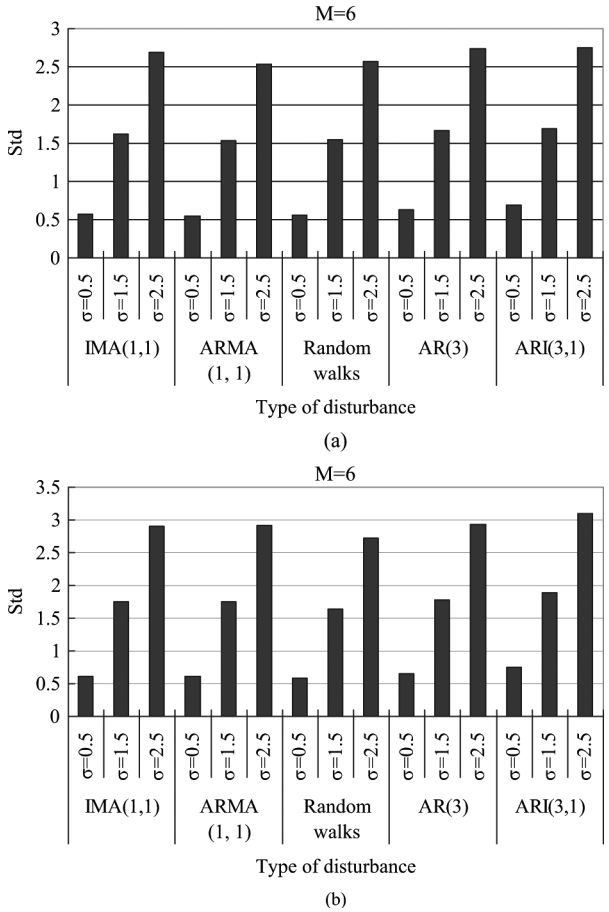


Fig. 8. Comparison of the magnitude of σ with $\xi = 1.1$ and $\mathrm{M} = 6$: (a) PPC with fixed optimum propagation parameters and (b) Self-tuning PPC.

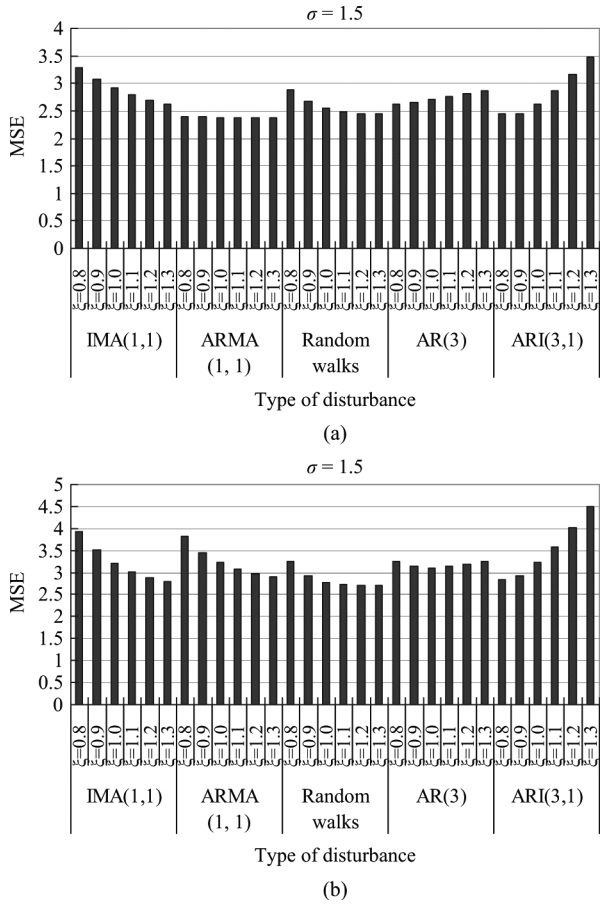


Fig. 9. Comparison of the magnitude of the model mismatch: (a) PPC with fixed optimum propagation parameters and (b) Self-tuning PPC.

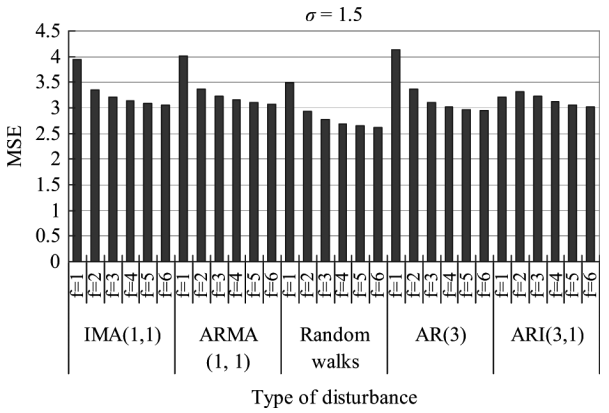


Fig. 10. The influence of the filter size f in (23).

weights of the other controllers listed in Table III were obtained using the minimum MSE from 100 sets of training data. Next, the fixed optimal controller parameters were applied to the testing data. The simulation results from the 100 sets of testing data with fixed optimum parameters controllers are listed in Table IV. Table IV also compares consistency of the simulation results of five anthropogenic disturbances by mean, range and variance. One observed that the fixed PPC has better output Std. and MSE for ARMA(1, 1), AR(3) and ARI(3, 1) disturbances and is consistent in Std. and MSE with smaller mean, range and variance. However, the Ave. of fixed PPC is worse than the other candidate controllers. Consequently, this study develops self-tuning PPC for improvement.

The three performance indices from the 100 sets of testing data with self-tuning controllers are listed at Table V. The Ave. index is improved over the one by fixed PPC. Table V also indicates that the self-tuning PPC, which is designed without *a priori* information about disturbance type, is superior to the other candidate self-tuning controllers in the sense of smallest variance in Std. and MSE. Summing up Table IV and 5, the self-tuning PPC is more consistent in output Std. and MSE than any other candidate controllers.

B. Fabrication Data

The test pattern, which was collected from semiconductor fabrication [32], is shown in Fig. 11, and is composed of ramp disturbance, step disturbance, and noise. Moreover, the test pattern has 197 runs, where the first 97 runs (runs 1–97) are designated as training data and the last 100 runs (runs 98–197) are the testing data.

The other simulation settings were $\alpha = \hat{\alpha} = 0$, $\boldsymbol{\beta} = [-0.7666, 0.5783]$, $\hat{\boldsymbol{\beta}} = [-0.8, 0.6]$, initial $\mathrm{input} = [0, 0]$, and $T = 0$; the upper and lower specification limits of the controlled outputs were set at 25 and -25, respectively. The simulation assumed that the process had two variations; one was a fixed gain with bias $\hat{\boldsymbol{\beta}} = \bar{\boldsymbol{\beta}} = [-0.8, 0.6]$, and the other

TABLE III
FIXED OPTIMUM CONTROLLER PARAMETERS OF THE ANTHROPOGENIC DISTURBANCE IN FIG. 10

Methods	Controller parameters	IMA(1, 1)	ARMA(1, 1)	Random walks	AR(3)	ARI(3, 1)
Fixed PPC	F	0.66	0.9	0.5	0.72	0.21
Fixed EWMA	W	0.56	0.46	0.99	0.97	1.00
Fixed PCC	W_1	0.54	0.46	0.99	0.97	1.00
	W_2	0.02	0.01	0.02	0.01	0.97
Fixed d-EWMA	W_1	0.56	0.51	0.99	0.98	1.00
	W_2	0.01	0.01	0.02	0.01	0.97

TABLE IV
PERFORMANCE COMPARISON OF THE ANTHROPOGENIC DISTURBANCES WITH DIFFERENT FIXED PARAMETER CONTROLLERS: (A) AVERAGE, (B) STANDARD DEVIATION, AND (C) MEAN SQUARE ERROR

Perf. index	Controller	Average results of anthropogenic disturbances					Consistency comparison of anthropogenic disturbances		
		IMA(1,1)	ARMA (1, 1)	Random walks	AR(3)	ARI(3,1)	Mean	Range	Variance
Ave.	PPC(M=6)	-0.19886	-0.16446	-0.12734	-0.07136	-0.01770	-0.11595	0.18116	0.07245
	EWMA	0.02856	0.00447	0.00636	-0.00001	-0.04663	-0.00145	0.07519	0.02757
	PCC	0.01800	0.00079	-0.00336	0.00116	-0.00211	0.00289	0.02136	0.00866
	d-EWMA	0.02148	0.00038	-0.00336	0.00116	-0.00211	0.00351	0.02485	0.01021
Std.	PPC(M=6)	1.62344	1.53257	1.55120	1.66175	1.69087	1.61196	0.15830	0.06862
	EWMA	1.49338	1.58086	1.51867	1.70158	2.29014	1.71693	0.79676	0.33038
	PCC	1.50549	1.58845	1.52959	1.71010	1.84332	1.63539	0.33782	0.14061
	d-EWMA	1.50012	1.58842	1.52959	1.70848	1.84332	1.63399	0.34320	0.14165
MSE	PPC(M=6)	2.79222	2.37506	2.48332	2.75683	2.86300	2.65408	0.48794	0.21231
	EWMA	2.24175	2.48698	2.32344	2.88278	5.46292	3.07958	3.22117	1.35498
	PCC	2.26004	2.51041	2.33596	2.91185	3.39199	2.68205	1.13195	0.47015
	d-EWMA	2.24792	2.51010	2.33596	2.90625	3.39199	2.67844	1.14407	0.47224

TABLE V
PERFORMANCE COMPARISON OF THE ANTHROPOGENIC DISTURBANCES WITH DIFFERENT SELF-TUNING CONTROLLERS: (A) AVERAGE, (B) STANDARD DEVIATION, AND (C) MEAN SQUARE ERROR

Perf. index	Controller	Average results of anthropogenic disturbances					Consistency comparison of anthropogenic disturbances		
		IMA(1,1)	ARMA (1, 1)	Random walks	AR(3)	ARI(3,1)	Mean	Range	Variance
Ave.	PPC(M=6)	-0.09319	-0.05229	-0.05587	-0.02900	0.00580	-0.04491	0.09899	0.00133
	EWMA	0.01695	0.00944	-0.01050	0.01948	0.00343	0.00776	0.02997	0.00014
	PCC	1.14726	1.11042	1.02599	1.12796	1.32834	1.14799	0.30235	0.01231
Std.	PPC(M=6)	1.75047	1.75420	1.64453	1.77320	1.88890	1.76226	0.24437	0.00756
	EWMA	1.76539	1.83319	2.30123	2.32403	4.70119	2.58501	2.93580	1.46597
	PCC	1.82140	1.68404	2.81913	2.12181	6.89650	3.06858	5.21246	4.77099
MSE	PPC(M=6)	3.09290	3.06855	2.71983	3.13443	3.58604	3.12035	0.86621	0.09523
	EWMA	3.15610	3.36222	5.42435	5.41372	23.40176	8.15163	20.24566	73.84859
	PCC	4.82460	4.07326	10.12159	5.79437	58.12683	16.58813	54.05357	544.6968

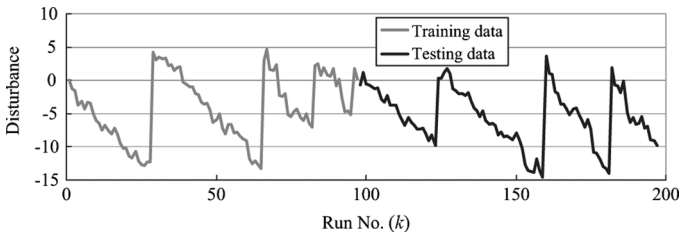
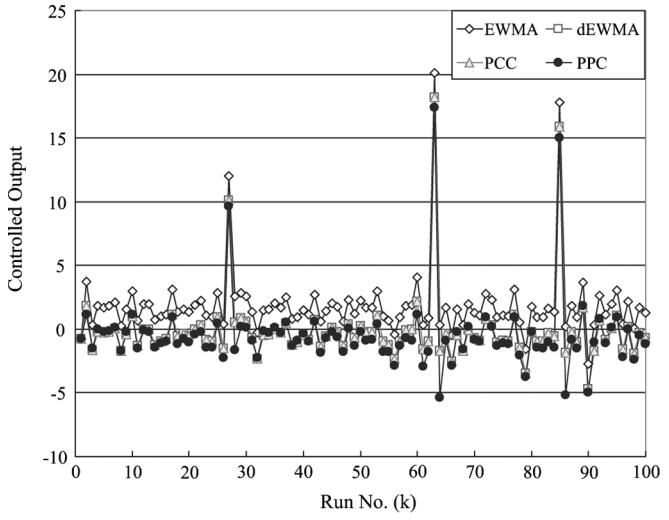
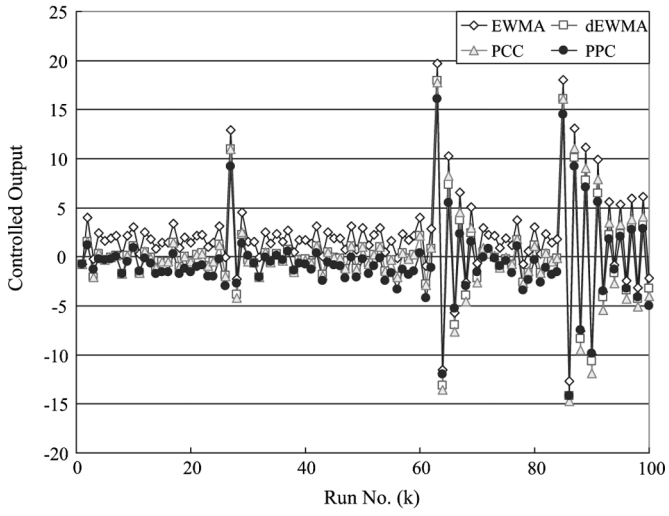


Fig. 11. Testing pattern composed of ramp disturbance, step disturbance, and noise.

drifted 0.01β per run from the first run, which amounts to a model mismatch ξ of approximately 1–2 for the training data and approximately 2–3 for the testing data. Thus, the simulation of the controllers with the fixed optimal controller parameters had eight scenarios with the permutation of hybrid disturbance, two types of system variation, and four control algorithms. The simulation of the self-tuning controllers included six scenarios, with permutations of the hybrid disturbance, two types of system variation, and three self-tuning controllers.



(a) System is fixed with a bias.



(b) System drifts 0.01 β per run from k =1.

Fig. 12. Output of the controllers with fixed optimum controller parameters.

Fig. 12(a) and 1(b) shows the output of the controllers with the fixed optimal controller parameters. Table VI summarizes the simulation results of Figs. 12(a) and (b). From Table VI, PCC and double EWMA performs slightly better than PPC when system is fixed with a bias. When the system gain drifts, the PPC has a good output Std. and MSE. Fig. 13(a) and (b) shows the controlled output and the tuned parameters of the self-tuning controllers. The results reveal that the PPC offset observed with the fixed propagation parameters disappeared, while the self-tuning EWMA [9] still had an offset near the target and converged slowly. In addition, the transient time of the self-tuning PCC [17] was slower than that of the self-tuning PPC, as shown in Fig. 13(a), and oscillated [Fig. 13(b)]. Table VII summarizes the simulation results for the self-tuning controllers and shows that the self-tuning PPC was superior to the other controllers in output Ave., Std., and MSE. In addition, the performance of off-line searching for the best parameters was better than that of self-tuning controllers when the process had a fixed bias. When the system drifted, the self-tuning PPC had the best performance of all in our simulations. Furthermore, the results are consistent whether the process had a fixed bias or drift.

VI. Conclusion

We used swarm intelligence to develop a new process controller, the PPC. PPC links swarm intelligence and APC by treating the intercepts of the linear regression model at different runs as a social insect colony and modeling the interactions among the intercepts in terms of propagation. We proposed a workable PPC scheme with the strategy of tuning the propagation parameter adaptively. The method can be easily extended to the MIMO case. Dealing with high order disturbances is the advantage of PPC; in particular, no training time as for neural work and no special rules as for fuzzy logic are the advantages of the proposed self-tuning PPC. Smaller stability region is the disadvantage. For the five anthropogenic disturbances, fixed PPC is more consistent than EWMA, PPC, and double EWMA, in sense of output Std. and MSE; self-tuning PPC improved Ave. index over fixed PPC and is the most consistent in Std. and MSE among candidate controllers. In short, PPC has advantage in higher order disturbance, such as AR(3) and ARI(3,1), and is as good as the competitors in lower order disturbance. For different disturbance model other than those in the paper, one can speculate PPC will perform well since it is capable of deal with complex disturbance with a scalable Mth order controller. In addition, for the semiconductor fabrication data, the double EWMA controller seems to perform slightly better than the PPC when system is fixed with a bias and the proposed self-tuning PPC was superior to the other candidate self-tuning controllers in our simulations and had the best performance when the process drifted.

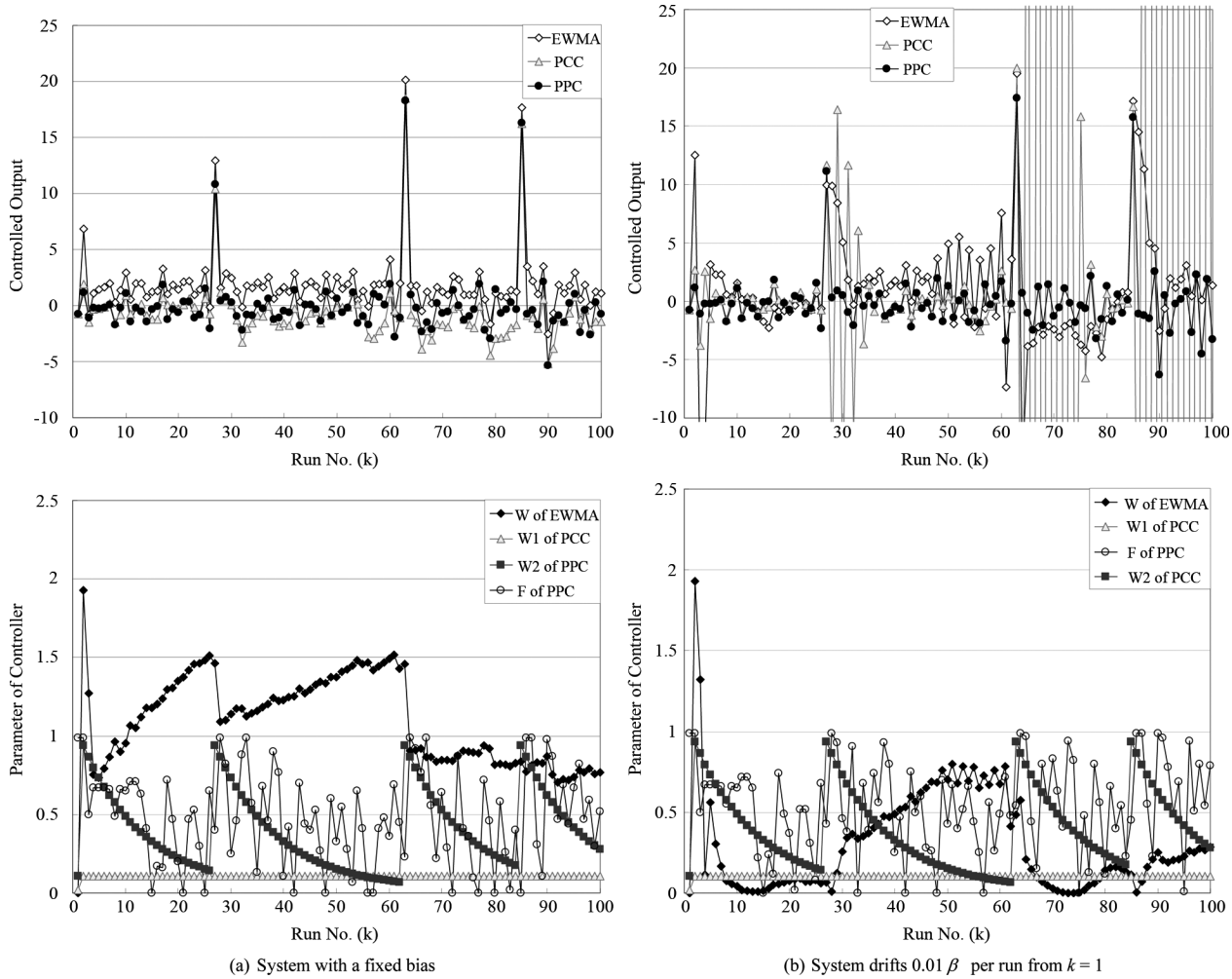
Because PPC is a meta-heuristic method and not designed for any specific type of disturbance, the performance can only be assessed by different cases. Outstanding issues requiring further study include effect of the magnitude of the model mismatch on the output MSE, a two-dimensional pheromone basket, the role of the evaporation parameter in PPC, and extension of the digital pheromone infrastructure to other fields such as parameter estimation or the disturbance observer.

Appendix A

We look at (2) and (4), and assume that the system is nondissipative over the short term. If the pattern of the external input to a pheromone basket $\mathbf{R}(k,0)$ is a series of ones, the final transition result of aggregated pheromone $\mathbf{S}(k,\infty)$ should also be a series of ones, regardless of the value of the propagation parameter F. Therefore, the transition function should obey the energy balance law, which means that the summation of the aggregated pheromone before transition is equal to that of the final transition results. In (2), when $i \to \infty$ and $F \in [0,1)$, $\sum_m [r(k,b_m,i)+q(k,b_m,i)] = F^i \sum_m r(k,b_m,0) = 0$. Thus, substituting (2) into (4) yields

$$\sum_m s(k,b_m,i+1) = \sum_m (E \times s(k,b_m,i)) = E \times \sum_m s(k,b_m,i)\ \forall m = 1,2,\ldots,\mathrm{M} \quad \text{(A1)}$$

and E should be 1.



(a) System with a fixed bias

(b) System drifts 0.01 β per run from $k = 1$

Fig. 13. Controlled output and the tuned parameters of the self-tuning controllers.

TABLE VI
SIMULATION RESULTS OF CONTROLLERS WITH FIXED PARAMETERS OBTAINED FROM MINIMUM MSE

System variation	Method	Controller parameters	Ave.	Std.	MSE
System is fixed with a bias.	PPC	$\overline{F} = 0.67$	–0.4371	2.7726	8.2264
	EWMA	$W = 0.99$	1.8758	2.8765	11.7100
	PCC	$W_1 = 0.99$ $W_2 = 0.01$	–0.0601	2.8798	8.2142
	d-EWMA	$W_1 = 0.99$ $W_2 = 0.01$	–0.0599	2.8800	8.2149
System drifts 0.01 β per run from $k = 1$	PPC	$\overline{F} = 0.90$	–1.6276	2.4672	11.9115
	EWMA	$W = 0.65$	1.9128	4.5622	23.0363
	PCC	$W_1 = 0.64$ $W_2 = 0.01$	–0.0463	4.5428	19.3256
	d-EWMA	$W_1 = 0.63$ $W_2 = 0.01$	–0.0343	4.2623	17.3057

TABLE VII
SIMULATION RESULTS OF SELF-TUNING CONTROLLERS

System variation	Method	Ave.	Std.	MSE
System is fixed with a bias.	PPC	0.0061	2.9847	8.8195
	EWMA	1.9226	2.9656	12.4037
	PCC	–0.5580	3.0796	9.7001
System drifts 0.01 β per run from $k = 1$	PPC	0.0189	3.0312	8.6145
	EWMA	0.8997	5.0904	26.2336
	PCC	–0.2061	185.9166	20736.8782

APPENDIX B

To overcome the end effect as shown in Fig. 2, we modify the propagation-out ratio in the two extremities from $F/|N(b_{m'})|$ to Γ, which can be obtained by final value theorem in the Appendix C. First, (2) and (5) are rewritten as (A2) and (A3), found at the bottom of the next page.

By (A2) and (A3), (A8) in Appendix C becomes

$$\begin{bmatrix} \mathbf{Q}(k,i+1) \\ \mathbf{S}(k,i+1) \end{bmatrix} = \begin{bmatrix} \mathbf{T_{11}} & \mathbf{T_{12}} \\ \mathbf{T_{21}} & \mathbf{T_{22}} \end{bmatrix} \begin{bmatrix} \mathbf{Q}(k,i) \\ \mathbf{S}(k,i) \end{bmatrix} + \begin{bmatrix} \mathbf{U_{11}} & \mathbf{U_{12}} \\ \mathbf{U_{21}} & \mathbf{U_{22}} \end{bmatrix} \begin{bmatrix} \mathbf{R}(k,i) \\ \mathbf{R}(k,i) \end{bmatrix} \quad \text{(A4)}$$

where

$$\mathbf{Q}(k,i)=\begin{bmatrix} q(k,b_1,i)\\ q(k,b_2,i)\\ \vdots\\ q(k,b_m,i)\\ \vdots\\ q(k,b_{\mathrm{M}-1},i)\\ q(k,b_{\mathrm{M}},i)\end{bmatrix}_{\mathrm{M}\times 1}$$

$$\mathbf{S}(k,i)=\begin{bmatrix} s(k,b_1,i)\\ s(k,b_2,i)\\ \vdots\\ s(k,b_m,i)\\ \vdots\\ s(k,b_{\mathrm{M}-1},i)\\ s(k,b_{\mathrm{M}},i)\end{bmatrix}_{\mathrm{M}\times 1}$$

$$\mathbf{R}(k,i)=\begin{bmatrix} r(k,b_1,i)\\ r(k,b_2,i)\\ \vdots\\ r(k,b_m,i)\\ \vdots\\ r(k,b_{\mathrm{M}-1},i)\\ r(k,b_{\mathrm{M}},i)\end{bmatrix}_{\mathrm{M}\times 1}$$

$$\mathbf{T_{11}}=\begin{bmatrix} 0 & \frac{F}{2} & 0 & \cdots & 0 & 0\\ \Gamma & 0 & \frac{F}{2} & \cdots & 0 & 0\\ & \vdots & \ddots & & & \vdots\\ 0 & \cdots & \frac{F}{2} & 0 & \frac{F}{2} & \cdots & 0\\ \vdots & & & & \ddots & \vdots & \\ 0 & & \cdots & & \frac{F}{2} & 0 & \Gamma\\ 0 & & \cdots & & 0 & \frac{F}{2} & 0\end{bmatrix}_{\mathrm{M}\times\mathrm{M}}$$

$$\mathbf{T_{12}}=\mathbf{0}_{\mathrm{M}\times\mathrm{M}}$$

$$\mathbf{T_{21}}=\begin{bmatrix} 1-\Gamma & 0 & & \cdots & 0\\ 0 & (1-F) & & \cdots & 0\\ \vdots & & \ddots & & \vdots\\ 0 & \cdots & & (1-F) & 0\\ 0 & \cdots & & 0 & 1-\Gamma\end{bmatrix}_{\mathrm{M}\times\mathrm{M}}$$

$$\mathbf{T_{22}}=\mathbf{I}_{\mathrm{M}\times\mathrm{M}}$$
$$\mathbf{U_{11}}=\mathbf{T_{11}}$$
$$\mathbf{U_{12}}=\mathbf{0}_{\mathrm{M}\times\mathrm{M}}$$
$$\mathbf{U_{21}}=\mathbf{0}_{\mathrm{M}\times\mathrm{M}}$$
$$\mathbf{U_{22}}=\mathbf{T_{21}}.$$

Without loss of generality, take "M equal to 5 and $\mathbf{R}(k,0)$ is the 5×1 matrix of ones" as an example. The final propagation result of $\mathbf{V}(k)$ obtained from (A9) must be the 5×1 matrix of ones.

$$\begin{aligned}\mathbf{V}(k)&=\begin{bmatrix} v_1(k)\\ v_2(k)\\ v_3(k)\\ v_4(k)\\ v_5(k)\end{bmatrix}_{5\times 1}\\ &=\frac{1}{F^2+\Gamma F-2}\begin{bmatrix} -\frac{1}{2}(F^2-2F-4)(\Gamma-1)\\ (F-1)(F+2\Gamma+2)\\ (F-1)(2F+\Gamma F+2)\\ (F-1)(F+2\Gamma+2)\\ -\frac{1}{2}(F^2-2F-4)(\Gamma-1)\end{bmatrix}_{5\times 1}\\ &=\begin{bmatrix}1\\1\\1\\1\\1\end{bmatrix}_{5\times 1}\end{aligned}\tag{A5}$$

Solving (A5) yields

$$\Gamma=\frac{F}{2-F}.\tag{A6}$$

Relatively, when $\mathbf{R}(k,0)$ is the $M\times 1$ matrix of ones, (A9) in Appendix C becomes

$$\begin{aligned}&\lim_{\mathrm{z}\to 1}(\mathrm{z}-1)\mathrm{Z}\{\mathbf{H}(k,i)\}\\ &=\lim_{\mathrm{z}\to 1}(\mathrm{z}-1)(\mathrm{z}\mathbf{I}-\mathbf{T})^{-1}\mathbf{U}\mathbf{I}_{2\mathrm{M}\times 1}=\begin{bmatrix}\mathbf{0}_{\mathrm{M}\times 1}\\ \mathbf{1}_{\mathrm{M}\times 1}\end{bmatrix}_{2\mathrm{M}\times 1}.\end{aligned}\tag{A7}$$

Equation (A7) shows that the final propagation result $\mathbf{S}(k,\infty)$ is also an $M\times 1$ matrix of ones. Thus, the modified transition functions not only obey the energy balance law but also avoid the end effect.

$$\begin{aligned}&q(k,b_m,i+1)\\ &=\begin{cases}\frac{F}{2}(r(k,b_{m+1},i)+q(k,b_{m+1},i)), & \text{if } m=1\\ \Gamma(r(k,b_{m-1},i)+q(k,b_{m-1},i))+\frac{F}{2}(r(k,b_{m+1},i)+q(k,b_{m+1},i)), & \text{if } m=2\\ \frac{F}{2}(r(k,b_{m-1},i)+q(k,b_{m-1},i))+\frac{F}{2}(r(k,b_{m+1},i)+q(k,b_{m+1},i)), & \text{if } 2<m<\mathrm{M}-1\\ \frac{F}{2}(r(k,b_{m-1},i)+q(k,b_{m-1},i))+\Gamma(r(k,b_{m+1},i)+q(k,b_{m+1},i)), & \text{if } m=\mathrm{M}-1\\ \frac{F}{2}(r(k,b_{m-1},i)+q(k,b_{m-1},i)), & \text{if } m=\mathrm{M}\end{cases}\end{aligned}\tag{A2}$$

$$\begin{aligned}&s(k,b_m,i+1)\\ &=\begin{cases}s(k,b_m,i)+(1-\Gamma)(r(k,b_m,i)+q(k,b_m,i)), & \text{if } m=1,\mathrm{M}\\ s(k,b_m,i)+(1-F)(r(k,b_m,i)+q(k,b_m,i)), & \text{if } 1<m<\mathrm{M}\end{cases}\end{aligned}\tag{A3}$$

$$G_{\mathrm{MA6}}(z)=\frac{(z-1)(6z^5+5z^4+4z^3+3z^2+2z+1)}{6z^6+(\xi-1)z^5+(\xi-1)z^4+(\xi-1)z^3+(\xi-1)z^2+(\xi-1)z+(\xi-1)} \tag{A11}$$

APPENDIX C

Because states $\mathbf{Q}(k,i)$ and $\mathbf{S}(k,i)$ are updated simultaneously, (6) and (7) can be rewritten in matrix form:

$$\begin{bmatrix}\mathbf{Q}(k,i+1)\\ \mathbf{S}(k,i+1)\end{bmatrix}=\begin{bmatrix}\mathbf{T_{11}} & \mathbf{T_{12}}\\ \mathbf{T_{21}} & \mathbf{T_{22}}\end{bmatrix}\begin{bmatrix}\mathbf{Q}(k,i)\\ \mathbf{S}(k,i)\end{bmatrix}+\begin{bmatrix}\mathbf{U_{11}} & \mathbf{U_{12}}\\ \mathbf{U_{21}} & \mathbf{U_{22}}\end{bmatrix}\begin{bmatrix}\mathbf{R}(k,i)\\ \mathbf{R}(k,i)\end{bmatrix} \tag{A8}$$

where

$$\mathbf{Q}(k,i)=\begin{bmatrix}q(k,b_1,i)\\ q(k,b_2,i)\\ \vdots\\ q(k,b_m,i)\\ \vdots\\ q(k,b_{\mathrm{M}-1},i)\\ q(k,b_{\mathrm{M}},i)\end{bmatrix}_{\mathrm{M}\times 1}$$

$$\mathbf{S}(k,i)=\begin{bmatrix}s(k,b_1,i)\\ s(k,b_2,i)\\ \vdots\\ s(k,b_m,i)\\ \vdots\\ s(k,b_{\mathrm{M}-1},i)\\ s(k,b_{\mathrm{M}},i)\end{bmatrix}_{\mathrm{M}\times 1}$$

$$\mathbf{R}(k,i)=\begin{bmatrix}r(k,b_1,i)\\ r(k,b_2,i)\\ \vdots\\ r(k,b_m,i)\\ \vdots\\ r(k,b_{\mathrm{M}-1},i)\\ r(k,b_{\mathrm{M}},i)\end{bmatrix}_{\mathrm{M}\times 1}$$

$$\mathbf{T_{11}}=\begin{bmatrix}0 & \frac{F}{2} & 0 & \cdots & 0 & 0\\ \frac{F}{2} & 0 & \frac{F}{2} & \cdots & 0 & 0\\ \vdots & \ddots & & & & \vdots\\ 0 & \cdots & \frac{F}{2} & 0 & \frac{F}{2} & \cdots & 0\\ \vdots & & & \ddots & \vdots & & \\ 0 & \cdots & & \frac{F}{2} & 0 & \frac{F}{2} & \\ 0 & \cdots & & 0 & \frac{F}{2} & 0 & \end{bmatrix}_{\mathrm{M}\times\mathrm{M}}$$

$$\mathbf{T_{12}}=\mathbf{0}_{\mathrm{M}\times\mathrm{M}}$$

$$\mathbf{T_{21}}=\begin{bmatrix}1-\left(\frac{F}{2-F}\right) & 0 & \cdots & 0\\ 0 & (1-F) & \cdots & 0\\ \vdots & & \ddots & \vdots\\ 0 & \cdots & (1-F) & 0\\ 0 & \cdots & 0 & 1-\left(\frac{F}{2-F}\right)\end{bmatrix}_{\mathrm{M}\times\mathrm{M}}$$

$$\mathbf{T_{22}}=\mathbf{I}_{\mathrm{M}\times\mathrm{M}}$$

$$\mathbf{U_{11}}=\mathbf{T_{11}}$$

$$\mathbf{U_{12}}=\mathbf{0}_{\mathrm{M}\times\mathrm{M}}$$

$$\mathbf{U_{21}}=\mathbf{0}_{\mathrm{M}\times\mathrm{M}}\text{ and}$$

$$\mathbf{U_{22}}=\mathbf{T_{21}}.$$

In (A8), $\mathbf{Q}(k,\infty)$ and $\mathbf{S}(k,\infty)$ can be obtained with the z-transform and the final value theorem

$$\begin{aligned}\mathbf{H}(k,\infty)&=\lim_{\mathrm{z}\to 1}(\mathrm{z}-1)\mathrm{Z}\left\{\mathbf{H}(k,i)\right\}\\ &=\lim_{\mathrm{z}\to 1}(\mathrm{z}-1)(\mathrm{z}\mathbf{I}-\mathbf{T})^{-1}\mathbf{U}\begin{bmatrix}\mathbf{R}(k,i)\\ \mathbf{R}(k,i)\end{bmatrix}_{2\mathrm{M}\times 1}\\ &=\begin{bmatrix}\mathbf{0}_{\mathrm{M}\times 1}\\ \mathbf{V}(k)\end{bmatrix}_{2\mathrm{M}\times 1}.\end{aligned} \tag{A9}$$

where $\mathbf{H}(k,i)=\begin{bmatrix}\mathbf{Q}(k,i)\\ \mathbf{S}(k,i)\end{bmatrix}_{2\mathrm{M}\times 1}$, $\mathbf{T}=\begin{bmatrix}\mathbf{T_{11}} & \mathbf{T_{12}}\\ \mathbf{T_{21}} & \mathbf{T_{22}}\end{bmatrix}_{2\mathrm{M}\times 2\mathrm{M}}$, $\mathbf{U}=\begin{bmatrix}\mathbf{U_{11}} & \mathbf{U_{12}}\\ \mathbf{U_{21}} & \mathbf{U_{22}}\end{bmatrix}_{2\mathrm{M}\times 2\mathrm{M}}$ and $\mathbf{V}(k)=\begin{bmatrix}v_1(k)\\ \vdots\\ v_{\mathrm{M}}(k)\end{bmatrix}_{\mathrm{M}\times 1}$. Note that if the above system is stable, $\mathbf{Q}(k,\infty)$ in (A9) will converge to the matrix of 0 and $\mathbf{S}(k,\infty)$ will converge to $\mathbf{V}(k)$. In addition, the z-transform of the external input $\mathrm{Z}\{\mathbf{R}(k,i)\}$ is equal to $\mathbf{R}(k,0)$ since $\mathbf{R}(k,i)$ is an impulse at $i=0$ by definition in Section II-A. Thus, $\mathbf{V}(k)$ is a function of F and $\mathbf{R}(k,0)$ for a specific M. The final propagation results can be obtained analytically using (A9).

APPENDIX D

When F approaches 1, the PPC is equal to disturbance with a moving average (MA) filter; this result can also be observed from Fig. 5(c). In the following, we analyze the stability of MA(4) and MA(6) to show a higher order MA controller has a larger stability region than the lower order one.

1) The transfer function from disturbance to output of a MA(4) controller is

$$G_{\mathrm{MA4}}(z)=\frac{(z-1)(4z^3+3z^2+2z+1)}{4z^4+(\xi-1)z^3+(\xi-1)z^2+(\xi-1)z+(\xi-1)} \tag{A10}$$

where the stability region is 5.

2) The transfer function from disturbance to output of a MA(6) controller is (A11), found at the top of the page. where the stability region is 7.

REFERENCES

[1] E. Sachs, A. Hu, and A. Ingolfsson, "Run by run control: Combining SPC and feedback control," *IEEE Trans. Semicond. Manuf.*, vol. 8, no. 1, pp. 26–43, Feb. 1995.

[2] S. J. Qin, G. Cheery, R. Good, J. Wang, and C. A. Harrison, "Semiconductor manufacturing process control and monitoring: A fab-wide framework," *J. Proc. Contr.*, vol. 16, pp. 179–191, 2006.
[3] J. S. Hunter, "The exponentially weighted moving average," *J. Qual. Technol.*, vol. 18, pp. 203–209, 1986.
[4] A. Ingolfsson and E. Sachs, "Stability and sensitivity of an EWMA controller," *J. Qual. Technol.*, vol. 25, pp. 271–287, Oct. 1993.
[5] S.-T. Tseng, R.-J. Chou, and S.-P. Lee, "A study on a multivariate EWMA controller," *IIE Trans.*, vol. 34, pp. 541–549, Jun. 2002.
[6] R. P. Good and S. J. Qin, "On the stability of MIMO EWMA run-to-run controllers with metrology delay," *IEEE Trans. Semicond. Manuf.*, vol. 19, no. 1, pp. 78–86, Feb. 2006.
[7] T. Smith and D. Boning, "A self-tuning EWMA controller utilizing artificial neural network function approximation techniques," *IEEE Trans. Compon., Packag., Manuf. Technol. C*, vol. 20, no. 2, pp. 121–132, 1997.
[8] C. C. Hsu and C. T. Su, "A neural network-based adaptive algorithm on the single EWMA controller," *Int. J. Manufact. Technol.*, vol. 23, pp. 586–593, Apr. 2004.
[9] N. S. Patel and S. T. Jenkins, "Adaptive optimization of run-to-run controllers: The EWMA example," *IEEE Trans. Semicond. Manuf.*, vol. 13, no. 1, pp. 97–107, Feb. 2000.
[10] S. Adivikolanu and E. Zafiriou, "Extensions and performance/robustness tradeoffs of the EWMA run-to-run controller by using the internal model control structure," *IEEE Trans. Electron. Packag. Manufact.*, vol. 23, no. 1, pp. 56–68, Jan. 2000.
[11] S. T. Tseng, A. B. Yeh, F. Tsung, and Y. Y. Chan, "A study of the variable EWMA controller," *IEEE Trans. Semicond. Manuf.*, vol. 16, no. 4, pp. 633–642, Nov. 2003.
[12] S. Bulter and J. Stefani, "Supervisory run-to-run control of polysilicon gate etch using in situ ellipsometry," *IEEE Trans. Semicond. Manuf.*, vol. 7, no. 2, pp. 193–201, May 1994.
[13] A. Chen and R. S. Guo, "Age-based double EWMA controller and its application to CMP processes," *IEEE Trans. Semicond. Manuf.*, vol. 14, no. 1, pp. 11–19, Feb. 2001.
[14] S.-T. Tseng and N.-J. Hsu, "Sample-size determination for achieving asymptotic stability of a double EWMA control scheme," *IEEE Trans. Semicond. Manuf.*, vol. 18, no. 1, pp. 104–111, Feb. 2005.
[15] E. D. Castillo and R. Rajagopal, "A multivariate double EWMA process adjustment scheme for drifting process," *IIE Trans.*, vol. 34, pp. 1055–1068, Dec. 2002.
[16] E. D. Castillo, "Long run and transient analysis of a double EWMA feedback controller," *IIE Trans.*, vol. 31, no. 12, pp. 1157–1169, 1999.
[17] C. T. Su and C. C. Hsu, "A time-varying weights tuning method of the double EWMA controller," *Omega*, vol. 32, pp. 473–480, 2004.
[18] S. T. Tseng, W. Song, and Y. C. Chang, "An initial intercept iteratively adjusted (IIIA) controller: An enhanced double EWMA feedback control scheme," *IEEE Trans. Semicond. Manuf.*, vol. 18, no. 3, pp. 448–457, Aug. 2005.
[19] X. A. Wang and R. L. Mahajan, "Artificial neural network model-based run-to-run process controller," *IEEE Trans. Compon., Packag., Manuf. Technol. C*, vol. 19, no. 1, pp. 19–26, Jan. 1996.
[20] C. T. Su, J. T. Wong, and S. C. Tsou, "A process parameters determination model by integrating artificial neural network and ant colony optimization," *J. Chinese Inst. Ind. Eng.*, vol. 22, no. 4, pp. 346–354, 2005.
[21] J. Wang, Q. P. He, S. J. Qin, C. A. Bode, and M. A. Purdy, "Recursive least square estimation for run-to-run control with metrology delay and its application to STI etch process," *IEEE Trans. Semicond. Manuf.*, vol. 18, no. 2, pp. 309–319, May 2005.
[22] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems, Santa Fe Institute Studies in the Sciences of Complexity*. New York: Oxford Univ. Press, 1999.
[23] M. Dorigo, V. Maniezzo, and A. Colorni, "The ant system: Optimization by a colony of cooperating agents," *IEEE Trans. Syst. Man Cybern. B*, vol. 26, no. 1, pp. 29–41, Feb. 1996.
[24] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Networks*, Piscataway, NJ, 1995, vol. 4, pp. 1942–1948.
[25] J. M. Bishop, "Stochastic searching networks," in *Proc. 1st IEE Conf. Artificial Neural Networks*, London, U.K., Oct. 1989, pp. 329–331.
[26] H. Jiang, L. Li, F. Qiao, and Q. Wu, "The new method of dynamic scheduling in semiconductor fabrication line," *8th ICARCV*, vol. 3, pp. 1874–1878, Dec. 2004.
[27] S. Brueckner, "Return from the Ant: Synthetic Ecosystems for Manufacturing Control," Ph.D. dissertation, Humboldt Univ. of Berlin, Berlin, Germany, 2000.
[28] H. V. D. Parunak, M. Purcell, and R. O'Connell, "Digital pheromones for autonomous coordination of swarming UAV's," in *Proc. 1st AIAA Unmanned Aerospace Vehicles, Systems, Technologies, and Operations Conf.*, May 2002.
[29] J. A. Sauter, R. S. Matthews, H. V. D. Parunak, and S. Brueckner, "Performance of digital pheromones for swarming vehicle control," *AAMAS*, pp. 903–910, Jul. 2005.
[30] B. Walter, A. Sannier, D. Reiners, and J. Oliver, "UAV swarm control: Calculating digital pheromone fields with the GPU," in *I/ITSEC*, Dec. 2005.
[31] H. V. D. Parunak and S. Brueckner, "Swarming coordination of multiple UAV's for collaborative sensing," in *Proc. 2nd AIAA Unmanned Unlimited Systems, Technologies, and Operations Conf.*, San Diego, CA, Sep. 2003.
[32] J. H. Chen, T. W. Kuo, and A. C. Lee, "Run-by-run process control of metal sputter deposition: Combining time series and extended kalman filter," *IEEE Trans. Semicond. Manuf.*, vol. 20, no. 3, pp. 278–285, Aug. 2007.
[33] K. J. Astrom and B. Wittenmark, *Computer-Controlled Systems: Theory and Design*, 3rd ed. Englewood Cliffs, NJ: Prentice-Hall, 1997, pp. 453–462.

Der-Shui Lee received the B.S. degree in mechanical engineering from the Nation Chung-Hsing University, Tai-Chung, Taiwan, in 1997, and the M.S. degree in mechanical engineering from the National Chiao-Tung University, Hsin-Chu, Taiwan, in 1999. Currently, he is an on-the-job Ph.D. student in mechanical engineering at the National Chiao-Tung University.

After graduation, since 2000, he served as an Assistant Researcher in Chung-shan institute of Science and Technology (CSIST). His research interests include run-to-run (R2R) process control, swarm intelligence, and decision support system.

An-Chen Lee (M'04) received the B.S. and M.S. degrees in Power Mechanical Engineering from National Tsing-Hua University, Hsinchu, Taiwan, and the Ph.D. degree in 1986 from University of Wisconsin-Madison in mechanical engineering.

He is designated as Chair Professor of National Chiao Tung University and currently a Professor in the Department of Mechanical Engineering. His current research interests are CNC machine tool control technology, Magnetic bearing technology, Rotor dynamic and control, and Semiconductor manufacturing process control.

Prof. Lee served as an Editorial Board member of International Journal of Precision Engineering and Manufacturing, Chinese society of Mechanical Engineers, and International Journal of Applied Mechanics and Engineering. He is the recipient of National Science Committee (NSC) Excellent Research Award (1991–1992), NSC Distinguished Research Award (1993–1994, 1995–1996, 1997–1998), NSC research fellow (1999–2001, 2002–2004), NSC research fellow Award (2005), Chinese Society of Mechanical Engineers Distinguished Engineering Professor Award (1995), and Chinese Society of Mechanical Engineers Distinguished Engineering Professor Award (2001).