Optimizing Fuzzy Neural Networks for Tuning PID Controllers Using an Orthogonal Simulated Annealing Algorithm OSA

Shinn-Jang Ho, *Member, IEEE*, Li-Sun Shu, *Student Member, IEEE*, and Shinn-Ying Ho, *Member, IEEE*

***Abstract*—In this paper, we formulate an optimization problem of establishing a fuzzy neural network model (FNNM) for efficiently tuning proportional-integral-derivative (PID) controllers of various test plants with under-damped responses using a large number P of training plants such that the mean tracking error J of the obtained P control systems is minimized. The FNNM consists of four fuzzy neural networks (FNNs) where each FNN models one of controller parameters (K, T_i, T_d, and b) of PID controllers. An existing indirect, two-stage approach used a dominant pole assignment method with $P = 198$ to find the corresponding PID controllers. Consequently, an adaptive neuro-fuzzy inference system (ANFIS) is used to independently train the four individual FNNs using input the selected 176 of the 198 PID controllers that 22 controllers with parameters having large variation are abandoned. The innovation of the proposed approach is to directly and simultaneously optimize the four FNNs by using a novel orthogonal simulated annealing algorithm (OSA). High performance of the OSA-based approach arises from that OSA can effectively optimize lots of parameters of the FNNM to minimize J. It is shown that the OSA-based FNNM with $P = 176$ can improve the ANFIS-based FNNM in averagely decreasing 13.08% error J and 88.07% tracking error of the 22 test plants by refining the solution of the ANFIS-based method. Furthermore, the OSA-based FNNMs using $P = 198$ and 396 from an extensive tuning domain have similar good performance with that using $P = 176$ in terms of J.**

***Index Terms*—Fuzzy neural network (FNN), optimal design, orthogonal experimental design (OED), proportional-integral-derivative (PID) controller, simulated annealing.**

I. Introduction

HOW TO tune proportional-integral-derivative (PID) controllers for plants with disturbance and under-damped responses is an important research from both theoretical and industrial viewpoints. Many tuning methods for PID controllers have been proposed [1], [2], [4], [10], [22], [25], [29]. Since the under-damped system with disturbance has an additional stability constraint that the damping ratio is smaller than one, it is more difficult to tune PID controllers for plants with under-damped responses than that for plants with over-damped responses [22]. Ho *et al.* [10] developed a tuning formula for self-tuning PID controller of a plant with under-damped responses based on the gain margin and phase margin specifications. The method proposed by Wang *et al.* [25] is based on the closed-loop pole allocation strategy through the use of root locus.

Recently, a two-stage approach to establishing a fuzzy neural network model (FNNM) consisting of four separate fuzzy neural networks (FNNs) for tuning PID controllers of various test plants with under-damped responses is proposed [22]. First, a dominant pole assignment method with a batch of 198 training plants is used to find the corresponding PID controllers. Second, an adaptive neuro-fuzzy inference system (ANFIS) [20], [21] is utilized to independently train the four individual FNNs for modeling the four individual parameters (K, T_i, T_d, b) of the PID controller using the input 176 of the 198 PID controllers that 22 controllers with parameters having large variation are abandoned. In [22], it is shown that the results of four test plants with various characteristics using the FNNM obtained from the ANFIS-based method have better balance performance between set-point and load-disturbance responses than those of the methods [10] and [25]. Because the FNNM is obtained without considering the global minimization of tracking errors for the entire tuning domain, the effective tuning domain is restricted and the tracking error is not yet minimized.

It is useful to establish an optimal FNNM which can efficiently tune the PID controllers for various test plants with under-damped step responses. The motivation of this study is to improve FNNM as well as the performances of the generated PID controllers by analyzing the merit and drawback of the ANFIS-based method and then propose an efficient method. In this paper, we formulate an optimization problem of establishing an FNNM for efficiently tuning PID controllers of various test plants with under-damped responses using a large number P of training plants. The objective function is to minimize the mean tracking error J of the obtained P control systems with the constraint that each of the P corresponding control systems satisfies a robustness constraint.

The major difficulties of solving the optimization problem are as follows: 1) The parameters in different FNNs have interactions and the number is as large as 258 in [22]; 2) the objective function J with these 258 parameters is highly nonlinear and its search space is multimodal; 3) it is almost impossible to derive an analytic solution and thus the optimal objective function value is not known in advance; and 4) the robustness constraints for all P control systems are very intractable.

Manuscript received December 22, 2003; revised March 19, 2005 and September 21, 2005.

S.-J. Ho is with the Department of Automation Engineering, National Formosa University, Huwei, Yunlin 632, Taiwan.

L.-S. Shu is with the Department of Information Management, Overseas Chinese Institute of Technology, Taichung 407, Taiwan.

S.-Y. Ho is with the Department of Biological Science and Technology, and Institute of Bioinformatics, National Chiao Tung University, Hsinchu 300, Taiwan (e-mail: syho@mail.nctu.edu.tw).

Digital Object Identifier 10.1109/TFUZZ.2006.876985

For coping with the aforementioned difficulties, we proposed an efficient approach to directly and simultaneously optimize the four FNNs by using a newly developed orthogonal simulated annealing algorithm (OSA) [11], [12]. High performance of the OSA-based approach arises from that OSA can simultaneously search for a solution of the 258 parameters to minimize J (for difficulties 1–3). The ANFIS-based method using a two-stage approach can obtain a feasible solution. To efficiently find a feasible solution as good as possible, the feasible solution from [22] is used as an initial solution of OSA (for difficulty 4).

OSA uses a divide-and-conquer approach to solving the large-scale problem based on orthogonal experimental design (OED) [3], [6], [18], [27]. Recently, we have successfully incorporated OED into simulated annealing [23] and genetic algorithm [8] for solving large-scale intractable engineering problems [11]–[13]. However, the newly-developed intelligent evolutionary algorithm (IEA) [13] requires an initial population of feasible solutions with high diversity (for difficulty 4) that is not an easy task. It is shown by computer simulation that the OSA-based FNNM with $P = 176$ can improve the ANFIS-based FNNM in averagely decreasing 13.08% error J and 88.07% tracking error of the 22 test plants. The OSA-based FNNM using the four specific test plants from [22] can averagely improve 15.63% tracking errors of the under-damped systems. Furthermore, the OSA-based FNNMs using $P = 198$ and 396 from an extensive tuning domain have similar good performance with that using $P = 176$ in terms of J.

This paper is organized as follows. In Section II, we formulate an optimization problem of establishing an FNNM for tuning PID controllers for plants with under-damped responses. In Section III, the analysis of the investigated optimization problem is given to illustrate the motivation of using the OSA-based approach. In Section IV, the approach OSA-based FNNM is presented in detail. Performance comparisons are given in Section V to demonstrate the efficiency of the proposed method. Finally, conclusions are given in Section VI.

II. Investigated Optimization Problem

The investigated optimization problem is to establish an FNNM for tuning PID controllers for plants with under-damped responses. Section II-A describes an under-damped system. Section II-B describes the architecture of FNNM. Section II-C gives a formulation of the investigated optimization problem.

A. Under-Damped System and PID Controller

Consider an under-damped system with disturbance. Let $G_p(s)$ be a training plant, $G_c(s)$ be a PID controller, $y_r(t)$ be a set point, $u(t)$ be a controller output, $y(t)$ be an output of the system, $e(t) = y_r(t) - y(t)$ be a tracking error, and $d(t)$ be an external disturbance. The under-damped training plant is modeled by the transfer function

$$G_P(s) = \frac{Y(s)}{U(s)} = \frac{\omega_n^2 e^{-sL}}{s^2 + 2\xi\omega_n s + \omega_n^2}, \qquad 0 < \xi < 1 \quad (1)$$

where L, ω_n, and ξ are dead time, natural frequency and damping ratio, respectively. The plant parameters (ω_n, ξ, L) in the training plants can be characterized by only two plant parameters (τ, ξ), where $\tau = \omega_n L/(1 + \omega_n L)(0 < \tau < 1)$ [10], [22]. The transfer function of the PID controller is

$$G_C(s) = \frac{U(s)}{E(s)} = K + \frac{K_i}{s} + K_d s. \quad (2)$$

For improving the set-point response of the system, a set-point weighting b is introduced into the PID controller as follows:

$$u = K\left((by_r - y) + \frac{1}{T_i}\int edt + T_d\frac{de}{dt}\right) \quad (3)$$

where K, T_i, T_d, and b are the four controller parameters. For obtaining better modeling results, T_i and T_d are normalized as T_i/L, and T_d/L, respectively [22]. Let M_s denote the maximum sensitivity and it is a robustness measure for stability, written as follows:

$$M_s = \max_{\omega}\left|\frac{1}{1 + G_p(j\omega)G_c(j\omega)}\right|. \quad (4)$$

From [2], we know that if the nominal system is stable and its M_s is less than a stability value (typical values fall in the range from 1.4 to 2.0), then the system subjected to a unit step disturbance at the process input is still stable. The design goal of FNNM is that the FNNM can generate PID controllers such that the tracking error $e(t)$ of the control system is as small as possible under the constraint that M_s is smaller than a prescribed value, such as the standard value 2.0 used by [22].

B. The Architecture of FNNM

An FNNM consists of four fuzzy neural networks (FNNs). Each FNN models one of the four controller parameters K, T_i, T_d, and b. The architecture of an FNN is shown in Fig. 1. We denote the output node i in layer l as $O_{l,i}$. Suppose plants (τ, ξ) are given and have $m \times n$ implications. The value z of one of the controller parameters K, T_i, T_d, and b is implied as follows.

Layer 1: Every node in this layer is an adaptive node with a node output defined by

$$\begin{aligned} O_{1,i} &= \mu_{\tau_i}(\tau), \qquad i = 1, \ldots, m \\ O_{1,i+m} &= \mu_{\xi_i}(\xi), \quad i = 1, \ldots, n \end{aligned} \quad (5)$$

where τ (or ξ) is the input to the node and τ_i (or ξ_i) is the fuzzy set associated with this node. In other words, outputs of this layer are the membership values of the premise part. The membership functions are characterized by the generalized bell functions

$$\begin{aligned} \mu_{\tau_i}(x) &= \frac{1}{1 + \left[(x - c_{\tau_i})^2 / a_{\tau_i}^2\right]^{b_{\tau_i}}} \\ \mu_{\xi_i}(x) &= \frac{1}{1 + \left[(x - c_{\xi_i})^2 / a_{\xi_i}^2\right]^{b_{\xi_i}}} \end{aligned} \quad (6)$$

where $\{a_{\tau_i}, b_{\tau_i}, c_{\tau_i}, a_{\xi_i}, b_{\xi_i}, c_{\xi_i}\}$ is a parameter set. Parameters in this layer are referred to as *premise parameters*.

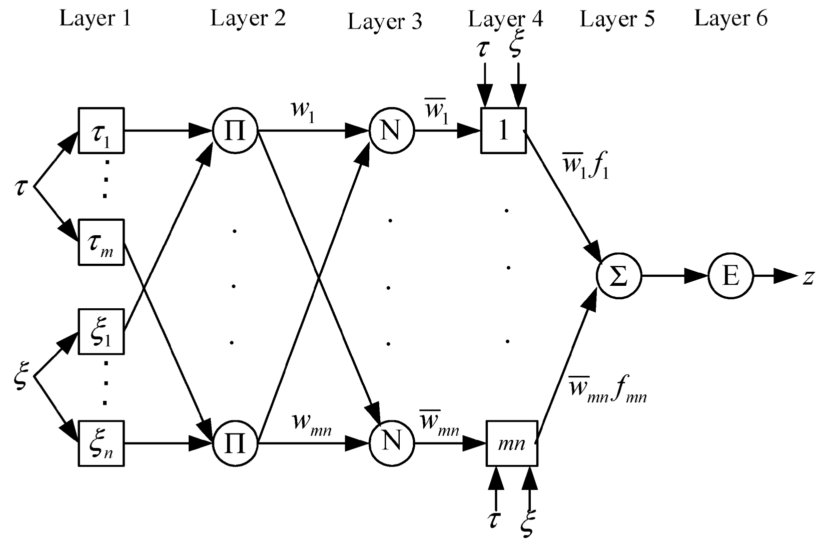

Fig. 1. Architecture of an FNN.

Layer 2: Every node in this layer is a fixed node labeled Π, which multiplies the incoming signals and outputs the product. For instance

$$\begin{aligned} O_{2,k} &= w_k \\ &= \mu_{\tau_i}(\tau) \times \mu_{\xi_j}(\xi), \\ &\qquad i = 1, \ldots, m,\ j = 1, \ldots, n \\ &\qquad k = (i-1)n + j. \end{aligned} \tag{7}$$

Each node output represents the firing strength of a rule.

Layer 3: Every node in this layer is a fixed node labeled N. The ith node calculates the ratio of the ith rule's firing strength to the sum of all rules' firing strengths

$$O_{3,i} = \overline{w}_i = \frac{w_i}{w_1 + w_2 + \ldots + w_{mn}}, \qquad i = 1, \ldots, mn. \tag{8}$$

Outputs of this layer are *called normalized firing strength*.

Layer 4: Every node i in this layer is an adaptive node with a node function

$$O_{4,i} = \overline{w}_i f_i = \overline{w}_i (p_i \tau + q_i \xi + r_i) \tag{9}$$

where $\overline{w}_i$ is the output of layer 3 and $\{p_i, q_i, r_i\}$ is a parameter set. Parameters in this layer are referred to as *consequent parameters*.

Layer 5: The single node in this layer is a fixed node labeled Σ that computes the output as the summation of all incoming signals

$$O_{5,1} = \sum_{i=1}^{mn} \overline{w}_i f_i. \tag{10}$$

Layer 6: The single node in this layer is a fixed node labeled E that computes the overall output as the exponential of incoming signal:

$$O_{6,1} = z = \exp(O_{5,1}). \tag{11}$$

For convenience, let the vector M_H, $H \in \{K, T_i, T_d, b\}$ be the parameters of an FNN for modeling H, which consists of $3(m + n + mn)$ consequent and premise parameters

$$\begin{aligned} M_H = \big[& a^H_{\tau_1}, \ldots, a^H_{\tau_m}, a^H_{\xi_1}, \ldots, a^H_{\xi_n}, p^H_1, \ldots, p^H_{mn}, \\ & b^H_{\tau_1}, \ldots, b^H_{\tau_m}, b^H_{\xi_1}, \ldots, b^H_{\xi_n}, q^H_1, \ldots, q^H_{mn} \\ & c^H_{\tau_1}, \ldots, c^H_{\tau_m}, c^H_{\xi_1}, \ldots, c^H_{\xi_n}, r^H_1, \ldots, r^H_{mn} \big]. \end{aligned} \tag{12}$$

In this study, we adopt the values of m and n used in [22]. For M_K, M_{T_i}, and M_{T_d}, $m = 4$ and $n = 3$. Due to the large variation of the parameter b, the larger values of $m = 5$ and $n = 4$ are used. The numbers of parameters of the four FNNs for K, T_i, T_d, and b are 57, 57, 57, and 87, respectively. Therefore, there are 258 parameters to be optimized for an FNNM.

C. The Optimization Problem

Designing an optimal FNNM has valuable contribution to tuning PID controllers from both theoretical and industrial viewpoints. In this paper, we obtained a useful FNNM by simultaneously formulating an optimization problem and developing an effective optimization algorithm. The optimization problem is to optimize an FNNM for tuning PID controllers for plants with under-damped responses using a large number P of training plants. For performance comparison with the FNNM in [22], the tracking error using the same Laplacian l_1-norm which treats the error uniformly at each point is adopted. The objective function is to minimize the mean tracking error J of the obtained P corresponding control systems

$$J = \sum_{j=1}^{P} \left(\int_0^{\infty} |e_j(t)|\, dt \right) / P \tag{13}$$

with the constraint that each of the P corresponding control systems satisfies the robustness constraint $M_s < 2.0$. Let X be a candidate solution consisting of the four FNN parameter vectors

$$X = [M_K,\ M_{T_i},\ M_{T_d},\ M_b] = [x_1,\ \ldots,\ x_{258}]. \tag{14}$$

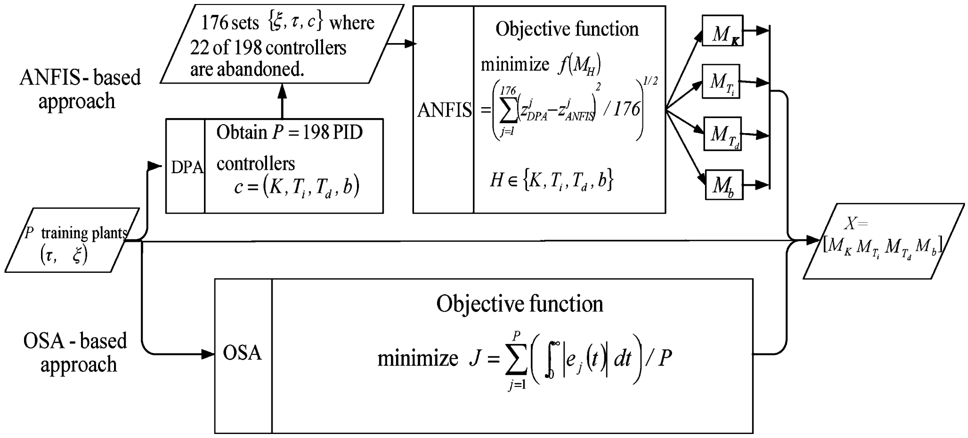

Fig. 2. Flowcharts of the ANFIS- and OSA-based approaches. The desired FNN model has a controller parameter vector $X = [M_K, M_{T_i}, M_{T_d}, M_b]$ to be optimized. The ANFIS-based approach uses the DPA method to obtain PID controllers and consequently uses ANFIS to model the four parameters K, T_i, T_d, and b of PID controllers using four independent runs to obtain X. The OSA-based approach directly and simultaneously optimize all parameters of X using the objection function of minimizing a mean tracking error J.

The quality of the optimized FNNM is evaluated in terms of J and P. An effective approach to optimizing FNNM should be able to handle an extensive tuning domain by sampling a large number P of training plants and obtain controllers with minimal tracking errors J for training and test plants.

The objective function of establishing an FNNM has the characteristics of nonlinear function, high degree of freedom, strong interactions among parameters, intractable constraints, multi-modal and huge search space. Moreover, it is hard to derive a mathematical formula for defining the objective function because of numerous highly interacted nonlinear functions in the fuzzy neural networks and under-damped system. Consequently, there is no analytic solution to the optimization problem. The essential issue is how to effectively search for a satisfactory solution X to the large-scale constrained parameter optimization problem, considering the above-mentioned characteristics.

How do we know if the objective function has a solution or not? Of course, we cannot tell with certainty whether an optimization algorithm will finally find an optimal solution. However, even though there is no certainty, there are some clues for optimizing the FNNM. Since the value $J > 0$ of the globally optimal solution cannot be known in advance because of huge search space and no analytic solution, the aim of our method using OSA is to find a potentially good approximation to the global optimum in a limited amount of computation time. Therefore, when a *feasible* solution with a small value J (as small as possible) is found which may be a local or near-local optimum, the obtained FNNM might be useful enough for real-world applications although a global optimum may be not yet found.

How to cope with the intractable constraint that each of the P corresponding control systems satisfies the robustness constraint $M_s < 2.0$ is the major concern in developing an optimization algorithm. If an algorithm can find a solution with the J value small enough, the constraint can be satisfied with a large probability. In other words, for convenient design, the algorithm can temporally ignore the constraint by focusing on pursuing the solution with a smallest J value.

To ensure high quality of the obtained PID controllers from test plants of practical applications, a large number P of training plants for increasing the sampling density in the tuning domain is helpful. However, increasing the value of P would make the search of feasible solutions more difficult if no heuristics are available. In the following section we would analyze the merit and weakness of the ANFIS-based approach [22], integrate our experience in solving large-scale optimization problems [11], [13], and develop an efficient method to establish a useful FNNM for tuning PID controllers.

III. Analysis of the Investigated Problem

In Section III-A, we briefly review and analyze the ANFIS-based approach [22]. In Section III-B, we illuminate our experience of utilizing the commonly-used population-based genetic algorithm (GA) [8], [13] and point-based simulated annealing (SA) technique [11], [12], [23] to solve the investigated problem.

A. The Merit and Weakness of the ANFIS-Based Approach

The ANFIS-based approach consists of two stages, as shown in Fig. 2. In Stage I, Shen chose a batch of $P = 198$ training plants (τ, ξ) which are combinations of ω_n, L, and $\xi = 0.1, 0.2, \ldots, 0.9$, as listed in Table I, and utilized a dominate pole assignment (DPA) method to obtain the corresponding 198 feasible PID controllers $c = (K, T_i, T_d, b)$ for systems with the robustness constraint $M_s < 2.0$. The variation of the parameters $z_{DPA} \in \{K, T_i, T_d, b\}$ obtained from DPA is large in the range $\tau > 0.5 (\omega_n L > 1.0)$ and $\xi < 0.3$. Generally, the plants in this range are those with large dead time, large natural frequency, and poor damping. These plants make it difficult to establish an FNN for identifying the relationship between (τ, ξ) and z_{DPA} using ANFIS in Stage II. Therefore, Shen abandoned 22 of the 198 PID controllers that their plants belong to this range. In Stage II, Shen used ANFIS to establish four separate FNNs to independently model the individual parameters K, T_i, T_d, and b by utilizing the 176 sets $\{\tau, \xi, c\}$.

TABLE I
198 TRAINING PLANTS (τ, ξ) WHICH ARE COMBINATIONS OF ω_n, L AND $\xi = 0.1, 0.2, \ldots, 0.9$

No.	1	2	3	4	5	6	7	8	9	10	11
ω_n	0.1	1.0	0.5	0.2	0.1	2.0	0.1	5.0	2.0	0.3	1.0
L	0.5	0.1	0.4	1.3	3.0	0.2	5.0	0.13	0.4	3.0	1.0
τ	0.048	0.091	0.167	0.206	0.231	0.286	0.333	0.394	0.444	0.474	0.500

No.	12	13	14	15	16	17	18	19	20	21	22
ω_n	0.2	3.0	0.6	6.0	2.0	8.0	1.0	3.0	2.0	4.0	3.0
L	6.0	0.5	3.0	0.4	1.5	0.5	5.0	2.0	4.0	3.0	5.0
τ	0.546	0.600	0.643	0.706	0.750	0.800	0.833	0.857	0.889	0.923	0.938

Let z^j_{DPA} and z^j_{ANFIS} be the values of a controller parameter H obtained from DPA and ANFIS for the jth training plant (τ, ξ). Shen used a root mean square error (RMSE) as an objective function for ANFIS to train an FNN, described as follows:

$$\text{minimize } f(M_H) = \left(\sum_{j=1}^{176} \left(z^j_{DPA} - z^j_{ANFIS} \right)^2 / 176 \right)^{1/2}. \tag{15}$$

The ANFIS identifies the relationship between multi-input parameters (τ, ξ) and a corresponding single-output z_{DPA}. Therefore, it takes four independent runs for ANFIS to establish the FNNM for the four controller parameters (K, T_i, T_d, b).

The merit of the two-stage ANFIS-based approach is to obtain a feasible solution to the problem of establishing an FNNM with the intractable constraints that each of the P corresponding control systems satisfies the robustness constraint $M_s < 2.0$. The weakness of the ANFIS-based approach is highlighted here.

1) The objective function (15) of optimizing the FNNM is the RMSE between z_{ANFIS} and z_{DPA}. That the four FNNs have small values of RMSE don't guarantee that the FNNM can generate PID controllers with a small value of J. It is better to optimize the FNNM by *directly* minimizing the tracking error which is a good performance measure of PID controllers.
2) The high performance of the PID controllers depends on that the four FNNs must be accurately determined *simultaneously* due to their interactions. The independent determination of each FNN would result in large tracking errors, especially when the modeling error of each FNN is very large, e.g., the errors of K, T_i, and T_d are inside $\pm 10\%$ and the error of b is inside $\pm 30\%$ [22].
3) The decrease of the number of training plants results from that the ANFIS cannot effectively model the controller parameters from the input data with large variation. If a large number P of training plants is used to extend the effective tuning domain, the variation of z_{DPA} would be very large. To avoid from decreasing the number P, it would be better to optimize the FNNM by directly determining the 258 parameters simultaneously to minimize J such that the number P can be advantageously increased.

B. Feasibility of Using GA and SA

Considering the merit and weakness of the ANFIS-based two-stage approach, an intuitive method is to directly search for an optimal solution X by minimizing the objective function J using optimization algorithms such as GA and SA, discussed here.

The majority of control applications in the literature adopted GA-based approaches [7]. Recently, researchers have become increasingly interested in the use of GA as a means to design various classes of control systems [5], [15]. Inspired from the mechanisms of natural evolution, GAs utilize a collective learning process of a population of individuals. Descendants of individuals are generated using randomized operations such as mutation and crossover. Mutation corresponds to an erroneous self-replication of individuals, while crossover exchanges information between two or more existing individuals. According to a fitness measure, the selection process favors better individuals to reproduce more often than those that are relatively worse [8]. The superiority of GA is achieved by using several search principles simultaneously such as population-based heuristics, and balance between global exploration and local exploitation. From our study using extensive simulations, we find that it is difficult to utilize GAs for obtaining a satisfactory solution to the investigated optimization problem because: 1) the population, having a large number, e.g., 50, of feasible solutions with large diversity, is hard to obtain using random generation mechanisms; 2) the children obtained from the usual search operator, crossover, are often infeasible and feasibility maintenance is difficult; and 3) the conventional GAs cannot effectively cope with the large-scale parameter optimization problem due to both large search space and expensive objective function evaluations [14].

Simulated annealing (SA) is an efficient optimization technique by iteratively refining an initial solution. Unlike the other point-based techniques such as hill-climbing, SA aims at escaping from local optima to find a globally optimal solution, and has been widely applied in various engineering problems [19], [26], [28]. The flowchart of a standard SA algorithm is shown in Fig. 3 [11]. A standard SA algorithm consists of a sequence of iterations. An initial solution is generated as a current solution. Each iteration employs a randomized perturbation on the current solution, e.g., the mutation of GA, to generate a candidate solution in the neighborhood of the current solution. The neighborhood is defined by the choice of the generation mechanism. The generation mechanism of a standard SA uses a generate-and-test method. If the candidate solution is better than the current solution, it is accepted as a new current solution. Otherwise, it is accepted according to Metropolis' criterion [17]

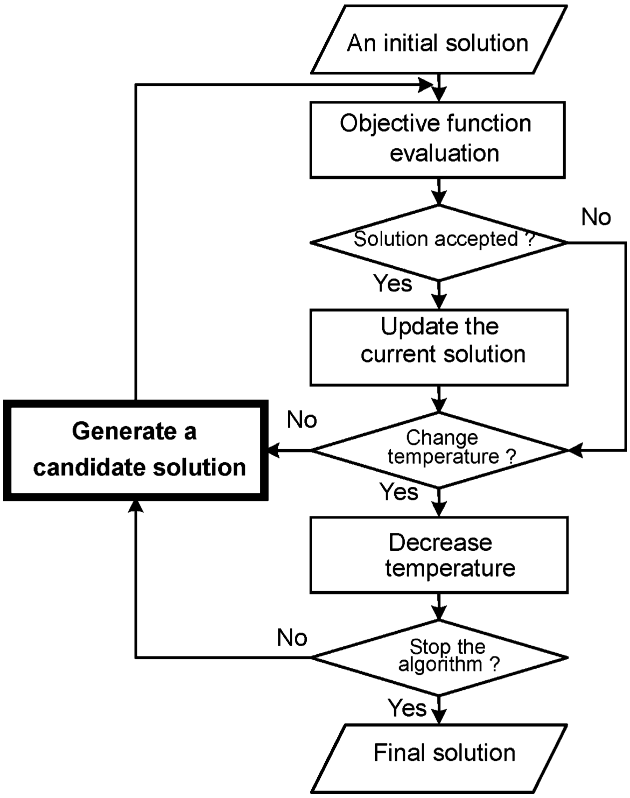

Fig. 3. Flowchart of a standard simulated annealing algorithm. The generation mechanism of OSA uses IGM, a systematic reasoning method based on orthogonal experimental design, instead of the conventional random generation mechanism [11].

based on Boltzman's probability. The generation mechanism of SA plays an important role in developing an efficient SA algorithm. The standard SA is difficult to explore an extremely large and nonlinear multimodal search space in a reasonable amount of computation time and is not acceptable for many intractable engineering applications [24]. It is also difficult to utilize a standard SA for coping with the investigated optimization problem because the random generation mechanism cannot effectively explore an extremely large search space.

IV. Proposed OSA-Based FNNM

A. Concepts of Orthogonal Experiment Design (OED)

An efficient way to study the effect of several factors simultaneously is to use OED with both orthogonal array (OA) and factor analysis. The factors are variables (parameters), which affect the response variable (objective function), and a setting (or a discriminative value) of a factor is regarded as a level of the factor. A "complete factorial" experiment would make measurements at each of all possible level combinations. However, the number of level combinations is often so large that this is impracticable, and a subset of level combinations must be judiciously selected to be used, resulting in a "fractional factorial" experiment [3], [6], [18]. OED utilizes properties of fractional factorial experiments to efficiently determine the best combination of factor levels to use in design problems.

OA is a fractional factorial array, which assures a balanced comparison of levels of any factor. OA is an array of numbers arranged in rows and columns where each row represents the levels of factors in each combination, and each column represents a specific factor that can be changed from each combination. The term "main effect" designates the effect on response variables that one can trace to a design parameter [3]. The array is called orthogonal because all columns can be evaluated independently of one another, and the main effect of one factor does not bother the estimation of the main effect of another factor [6], [18].

Factor analysis using the orthogonal array's tabulation of experimental results can allow the main effects to be rapidly estimated, without the fear of distortion of results by the effects of other factors. Factor analysis can evaluate the effects of individual factors on the evaluation function, rank the most effective factors, and determine the best level for each factor such that the evaluation function is optimized.

OED uses well-planned and controlled experiments in which certain factors are systematically set and modified, and then main effect of factors on the response can be observed. OED specifies the procedure of drawing a representative sample of experiments with the intention of reaching a sound decision [3]. Therefore, OED using OA and factor analysis is regarded as a systematic reasoning method.

An illustrative example of OED using an objective function is given as follows:

$$\text{maximize } f(x_1, x_2, x_3) = 100\,x_1 - 10\,x_2 - x_3 \qquad (16)$$

where $x_1 \in \{1, 2, 3\}$, $x_2 \in \{4, 5, 6\}$, and $x_3 \in \{7, 8, 9\}$. This maximization problem can be regarded as an experimental design problem of three factors, with three levels each. Let factors 1–3 be parameters x_1, x_2, and x_3, respectively. Let the small, medial, and large values of each parameter be the levels 1–3 of each factor, respectively. The objective function f is the responsible variable. A complete factorial experiment would evaluate $3^3 = 27$ level combinations and then the best combination $(x_1, x_2, x_3) = (3, 4, 7)$ with $f = 253$ can be obtained. Let f_h denote an objective function value of the level combination h. The factorial array and results of the complete factorial experiment are shown in Table II. A fractional factorial experiment uses a well-balanced subset of level combinations, such as the 1st, 5th, 9th, 11th, 15th, 16th, 21st, 22nd, and 26th combinations. The best one of the nine combinations is the 21st combination $(x_1, x_2, x_3) = (3, 4, 9)$ with $f = 251$. Using OED, we can reason the best combination (3, 4, 7) from analyzing the results of the nine specific combinations, described in the next section.

B. The Used OED

IGM uses one of two classes of OAs depending on applications. One is the class of two-level OAs used for optimization problems with a number of 0/1 decision variables [12]. The other is the class of three-level OAs used for optimization problems with continuous/discrete parameters [11]. All the optimization parameters are generally partitioned into N nonoverlapping groups. One group is regarded as a factor.

In this study, OSA uses OED with three-level OAs described later. Let there be N factors where each factor has three levels. The total number of level combinations is 3^N for a complete

TABLE II
RESULTS OF A COMPLETE FACTORIAL EXPERIMENT. THE UNDERLINED NUMBERS OF h FROM A WELL-BALANCED SUBSET WHICH CORRESPONDS TO AN ORTHOGONAL ARRAY $\mathrm{L}_9(3^4)$

h	**1**	2	3	4	**5**	6	7	8	**9**	10	**11**	12	13	14	**15**	**16**	17	18	19	20	**21**	**22**	23	24	25	**26**	27
x_1	1	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	2	3	3	3	3	3	3	3	3	3
x_2	4	4	4	5	5	5	6	6	6	4	4	4	5	5	5	6	6	6	4	4	4	5	5	5	6	6	6
x_3	7	8	9	7	8	9	7	8	9	7	8	9	7	8	9	7	8	9	7	8	9	7	8	9	7	8	9
f_h	53	52	51	43	42	41	33	32	31	153	152	151	143	142	141	133	132	131	253	252	251	243	242	241	233	232	231
Rank of f_h	19	20	21	22	23	24	25	26	27	10	11	12	13	14	15	16	17	18	1	2	3	4	5	6	7	8	9

factorial experiment. To use an OA of N factors, we obtain an integer $M = 3^{\lceil \log_3(2N+1) \rceil}$ where the bracket represents a ceiling operation, build an OA $\mathrm{L}_M(3^{(M-1)/2})$ with M rows and $(M-1)/2$ columns, use the first N columns, and ignore the other $(M-1)/2 - N$ columns [9], [11]. For example, if $N \in \{5, 6, \ldots, 13\}$, then $M = 27$ and $\mathrm{L}_{27}(3^{13})$ is used. The numbers 1, 2, and 3 in each column indicate the levels of the factors. Each column has an equal number of 1s, 2s, and 3s. The array is orthogonal when the nine pairs, (1,1), (1,2), (1,3), (2,1), (2,2), (2,3), (3,1), (3,2), and (3,3), appear the same number of times in any two columns. Table III illustrates an example of $\mathrm{L}_9(3^4)$.

OA can reduce the number of level combinations for factor analysis. The number of OA combinations required to analyze all individual factors is only $M = O(N)$, where $2N + 1 \leq M \leq 6N - 3$. Algorithms of constructing OAs with various levels can be found in [16]. The algorithms for constructing the two- and three-level OAs used by OSA can be found in [11]. After proper tabulation of experimental results, the summarized data are analyzed using factor analysis to determine the relative level effects of factors.

Define the main effect of factor i with level k as S_{ik} where $i = 1, \ldots, N$ and $k = 1, 2, 3$

$$S_{ik} = \sum_{h=1}^{M} f_h \cdot F_h \tag{17}$$

where $F_h = 1$ if the level of factor i of combination h is k; otherwise, $F_h = 0$. Consider the case that the objective function is to be minimized. The level k of factor i makes the best contribution to the objective function than the other two levels of factor i do when $S_{ik} = \min\{S_{i1}, S_{i2}, S_{i3}\}$. On the contrary, if the objective function is to be maximized, the level k is the best one when $S_{ik} = \max\{S_{i1}, S_{i2}, S_{i3}\}$. The main effect reveals the individual effect of a factor. The most effective factor has the largest one of main effect differences $MED_i, i = 1, \ldots, N$. Let $MED_i = \max\{S_{i1}, S_{i2}, S_{i3}\} - \min\{S_{i1}, S_{i2}, S_{i3}\}$. After the potentially best one of three levels of each factor is determined, an intelligent combination consisting of all factors with the best levels can be easily derived.

An illustrative example of OED for solving the optimization problem with (16) is described below (see Table IV). First, use an $\mathrm{L}_9(3^4)$, set levels for all factors as above mentioned, and evaluate the response variable f_h of the combination h, where $h = 1, \ldots, 9$. Second, compute the main effect S_{ik} where $i = 1, 2, 3$ and $k = 1, 2, 3$. For example, $S_{21} = f_1 + f_4 + f_7 = 456$. Third, determine the best level of each factor based on the main effect. For example, the best level of factor 1 is level 3 since $S_{13} > S_{12} > S_{11}$. Therefore, select $x_1 = 3$. Finally, the best combination $(x_1, x_2, x_3) = (3, 4, 7)$ with $f = 253$ can be obtained. The most effective factor is x_1 with $MED_1 = 600$ which is the largest one. It can be verified from (16) that x_1 has the largest coefficient 100. Note that if only OA combinations without factor analysis are used, the obtained best solution is $(x_1, x_2, x_3) = (3, 4, 9)$ with $f = 251$, not the reasoned solution (3, 4, 7).

TABLE III
ORTHOGONAL ARRAY $\mathrm{L}_9(3^4)$

Experiment no. h	Factor i 1	2	3	4	Objective function value f_h
1	1	1	1	1	f_1
2	1	2	2	2	f_2
3	1	3	3	3	f_3
4	2	1	2	3	f_4
5	2	2	3	1	f_5
6	2	3	1	2	f_6
7	3	1	3	2	f_7
8	3	2	1	3	f_8
9	3	3	2	1	f_9

TABLE IV
CONCISE EXAMPLE OF IGM

	Parameters				
h	x_1	x_2	x_3	f_h	Rank of f_h
1	1	4	7	53	19
2	1	5	8	42	23
3	1	6	9	31	27
4	2	4	8	152	11
5	2	5	9	141	15
6	2	6	7	133	16
7	3	4	9	251	3
8	3	5	7	243	4
9	3	6	8	232	8
S_{i1}	126	**456**	**429**		
S_{i2}	426	426	426		
S_{i3}	**726**	396	423		
Best level	3	1	1		
MED_i	600	60	6		
Solution x_i	3	4	7	253	1

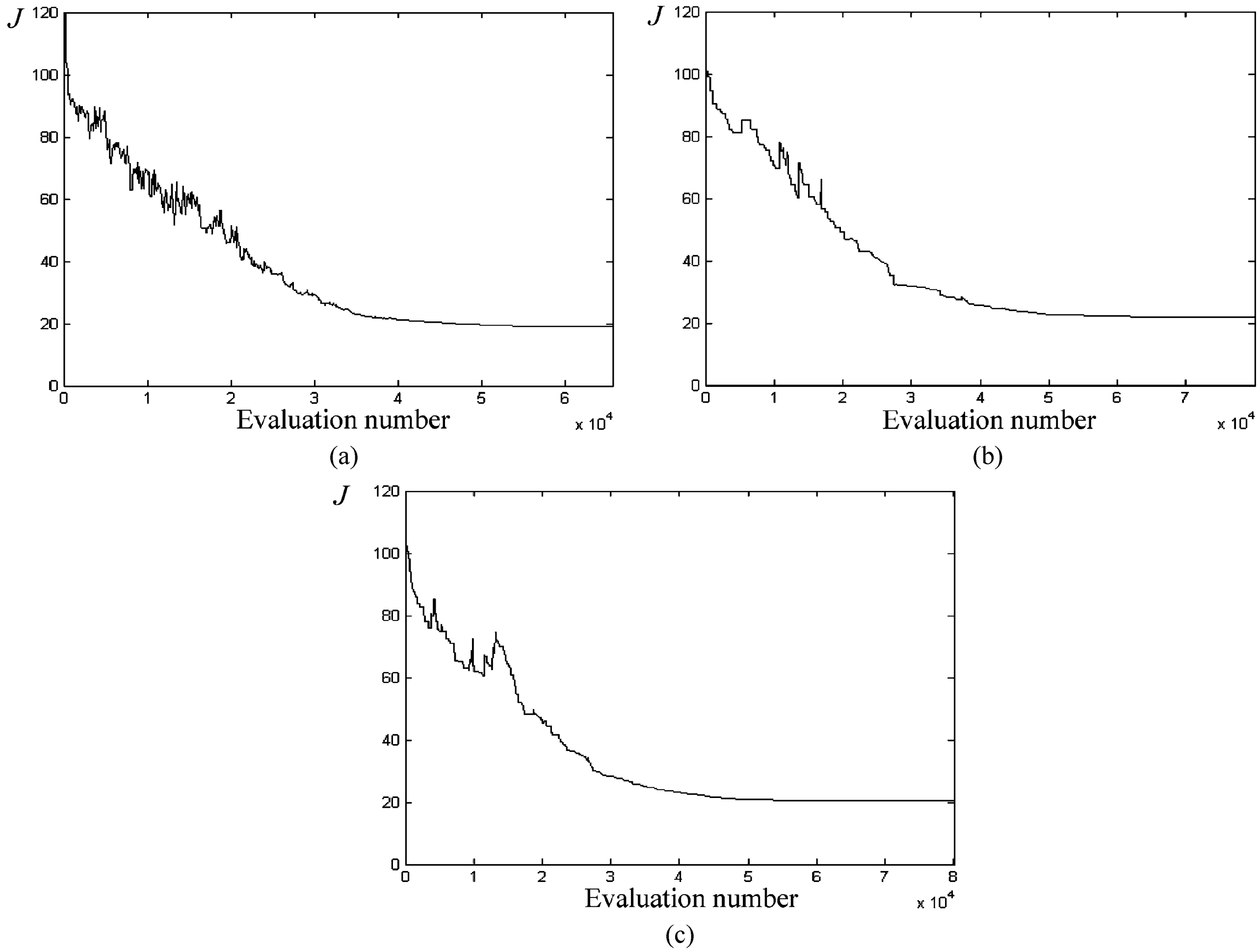

Fig. 4. Mean convergences of OSA from five independent runs. (a) T176. (b) T198. (c) T396.

C. Intelligent Generation Mechanism (IGM)

Consider a parametric optimization function with p parameters and a current solution $X = [x_1, \ldots, x_p]$. IGM generates two temporary solutions $X_1 = [x_1^1, \ldots, x_p^1]$ and $X_2 = [x_1^2, \ldots, x_p^2]$ by perturbing X, where x_i^1 and x_i^2 are generated from x_i as follows:

$$x_i^1 = x_i + \overline{x}_i \text{ and } x_i^2 = x_i - \overline{x}_i, \qquad i = 1, \ldots, p. \tag{18}$$

The values of $\overline{x}_i$ are generated by the Cauchy–Lorentz probability distribution [23]. If $x_i^1(x_i^2)$ is out of the domain range of x_i, randomly assign a feasible value to $x_i^1(x_i^2)$. IGM aims at efficiently combining good values of parameters from solutions X, X_1, and X_2 to generate a good candidate solution Q for the next move.

Divide all the p parameters into N nonoverlapping groups using the same division scheme for X, X_1, and X_2, $i = 1, \ldots, N$. The proper value of N is problem-dependent. The larger the value of N, the more efficient it is the IGM if the interactions among groups are weak. If the existing interactions among parameters are strong, the smaller the value of N, the more accurate it is the estimated main effect of individual groups. Considering the tradeoff, an efficient division criterion is to minimize the interactions among groups while maximizing the value of N. Note that the parameter N at each call of the following IGM operation can be a constant or variable value. In this study, OSA uses a constant value of N.

How to perform an IGM operation using N factors on a current solution X to an objective function f is described as follows.

- Step 1) Generate two temporary solutions X_1 and X_2 using X.
- Step 2) Divide each of X, X_1, and X_2 into N groups of parameters where each group is treated as a factor.
- Step 3) Use the first N columns of an OA $\mathrm{L}_M(3^{(M-1)/2})$, where $M = 3^{\lceil \log_3(2N+1) \rceil}$.
- Step 4) Let levels 1, 2 and 3 of factor i represent the ith groups of X, X_1, and X_2, respectively.
- Step 5) Compute f_h of the generated combination h, where $h = 2, \ldots, M$. Note that f_1 is the value of $f(X)$.
- Step 6) Compute the main effect S_{ik} where $i = 1, \ldots, N$ and $k = 1, 2, 3$.
- Step 7) Determine the best one of three levels of each factor based on the main effect.
- Step 8) The candidate solution Q is formed using the combination of the best groups.
- Step 9) Verify that Q is superior to the $M - 1$ sampling solutions derived from OA combinations and $Q \neq X$. If it is not true, let Q be the best one of these $M - 1$ sampling solutions.

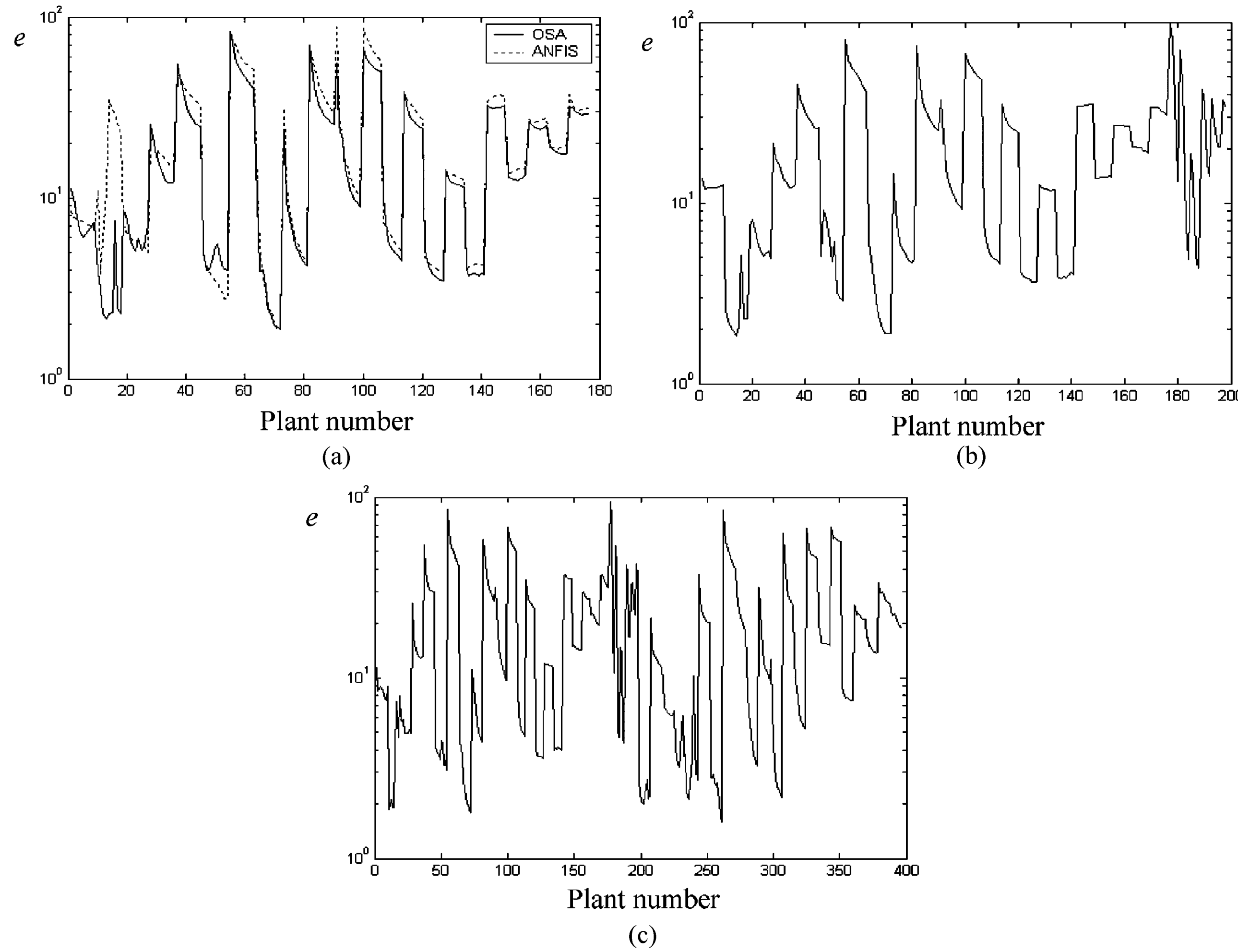

Fig. 5. Tracking error e of individual training plants. (a) T176. (b) T198. (c) T396.

The overhead of IGM in preparing OA experiments and factor analysis is relatively small, compared with the cost of function evaluations. Note that the used OAs are generated in advance. The number of objective function evaluations is M per IGM operation, which includes $M - 1$ evaluations in Step 5) and one in Step 9). Let G be the total number of iterations, which equals the number of IGM operations. The complexity of OSA is $GM + 1$ function evaluations. If interactions among groups are weak, Q is a potentially good approximation to the best one of all the 3^N combination.

D. The Proposed OSA-Based Approach

The purpose of the OSA-based approach is the same with that of the ANFIS-based approach, which is to obtain an FNNM consisting of four FNNs for tuning PID controllers. The OSA-based approach using a one-stage approach is shown in Fig. 2 for illuminating the difference from the ANFIS-based approach. We solve the investigated optimization problem using OSA to directly search for X with a minimal value of J. The only difference between OSA and a standard SA algorithm is that the generation mechanism of OSA uses an IGM-based on OED [3], [6], [9], [18], instead of the conventional random generation mechanism [11].

TABLE V
VALUES OF J FOR VARIOUS FNNMS

FNNM	Training set	Best of J	Mean of J	Standard deviation
ANFIS	T176	22.17	22.17	0
OSA	T176	18.81	19.27	0.47
OSA	T198	20.43	21.26	1.38
OSA	T396	19.86	20.39	0.65

OSA with IGM can hybridize the advantages of global exploration and local exploitation by focusing on accuracy and computation time. IGM can efficiently generate a good candidate solution for the next move by using a systematic reasoning method to efficiently exploit the neighborhood of a current solution, resulting in effectively obtaining a good solution to the large-scale parameter optimization problem [11], [12].

To effectively solve the intractable constraints of the optimization problem, we borrow a feasible solution $X_{\rm ANFIS}$ from [22] as an initial solution of OSA which is obtained using the ANFIS-based approach. Consequently, we specify effective domain ranges for each of the 258 parameters of FNNM. IGM can effectively search for a good feasible solution from the neighborhood of a current solution. The OSA-based approach is designed toward having the following objectives.

TABLE VI
TRACKING ERRORS e_{ANFIS} AND e_{OSA} FOR ANFIS AND OsSA, RESPECTIVELY, USING 22 UNSEEN TEST PLANTS T22. THE RATIO $e_{\mathrm{OSA}}/e_{\mathrm{ANFIS}}$ FOR SUM OF ALL ERRORS IS 11.93%

No.	1	2	3	4	5	6	7	8	9	10	11
τ	0.546	0.546	0.600	0.600	0.643	0.643	0.706	0.706	0.750	0.750	0.800
ξ	0.1	0.2	0.1	0.2	0.1	0.2	0.1	0.2	0.1	0.2	0.1
e_{OSA}	101.99	78.64	37.52	13.19	70.13	46.44	7.63	4.86	18.73	13.75	5.12
e_{ANFIS}	193.13	116.44	3452.99	12.87	170.81	61.60	6.09	5.62	13.34	13.11	3.75

No.	12	13	14	15	16	17	18	19	20	21	22
τ	0.800	0.833	0.833	0.857	0.857	0.889	0.889	0.923	0.923	0.938	0.938
ξ	0.2	0.1	0.2	0.1	0.2	0.1	0.2	0.1	0.2	0.1	0.2
e_{OSA}	4.36	42.59	35.44	16.51	14.05	37.45	27.63	20.54	20.47	37.40	34.12
e_{ANFIS}	3.61	33.81	33.83	18.47	14.00	1373.50	39.11	36.27	22.25	100.26	45.74

1) Accurate FNNM with a minimal value of J using a large number P of training plants can be obtained.
2) The effective tuning domain is extensive and the FNNM is accurate in obtaining PID controllers for unseen test plants.
3) FNNM can be specially designed by specifying an interesting tuning domain and sampling a proper number of training plants.

E. The Algorithm of the OSA-Based Method

There are four choices must be made in implementing an SA algorithm for solving an optimization problem: 1) solution representation, 2) objective function definition, 3) design of the generation mechanism, and 4) design of a cooling schedule. The choices 1 and 2 are problem-dependent. Designing an efficient generation mechanism plays an important role in developing SA algorithms. Generally, there are four parameters to be specified in designing the cooling schedule: 1) an initial temperature T_0, 2) a temperature update rule, 3) the number N_T of iterations to be performed at each temperature step, and 4) a stopping criterion of the SA algorithm.

The used OSA with IGM optimizes an FNNM using the objective function $J(X)$ in (13) with the robustness constraint. OSA uses a simple geometric cooling rule by updating the temperature at the $(i+1)$th temperature step using the formula: $T_{i+1} = CR \cdot T_i$, $i = 0, 1, \ldots$ where CR is the cooling rate which is a constant smaller than 1 but close to 1. The higher the temperature, the larger it is the possibility of accepting the candidate solution worse than the current solution. We borrow a feasible solution $X_{\mathrm{ANFIS}} = [a_1, \ldots, a_{258}]$ from Appendix of [22]. The effective domain range of each parameter x_i of X is defined as $a_i \pm 2a_i$, $i = 1, \ldots, 258$, derived from the experience of our computer simulations using various numbers P of raining plants.

The designs of generation mechanism and cooling schedule would affect the convergence of OSA. By gradually reducing the step size of moves and properly setting the parameters of cooling schedule, SA would finally converge to a local optimum [11], [23]. Of course, the generation mechanism would dominate the convergence performance. Considering the tradeoff of computation cost and solution quality, one can adaptively adjust the control parameters in the cooling schedule. The parameters of the following OSA for obtaining the solution X to the investigated problem are: $T_0 = 150$, $CR = 0.99$, $N_T = 1$, and the stopping condition uses $\delta = 10^{-5}$ and $N_{\mathrm{stop}} = 30$.

OED has been proven optimal for additive and quadratic models, and the selected combinations are good representations for all of the possible combinations [27]. The OED-based IGM performs well in terms of convergence performance, compared with the conventional random generation mechanism in solving large-scale optimization problem [11]. However, how to determine the size of OA and then assign the FNNM parameters to the corresponding factors is important. For advancing the performance of OSA, the FNNM parameters having stronger interactions are encoded together such as the vectors in (12) and (14). All the parameters (M_H) belonging to the same FNN are grouped together in (12). In this study, $N = 40$ and the used OA is $L_{81}(3^{40})$. Let p_i be the number of parameters of the ith FNN. One vector M_H corresponds to successive 10 factors and the p_i parameters are assigned to the 10 factors. The nine cut points are randomly specified from the $p_i - 1$ candidate cut points which separate individual parameters. The algorithm of the used OSA is described here.

Step 1) Initialize $T = T_0$. Generate an initial feasible solution X as a current solution. Let J^i be the value of J at the ith iteration and $i = 0$. Compute the value J^0.

Step 2) Perform an IGM operation using N factors on X to generate a candidate solution Q.

Step 3) Accept Q to be the new X with probability $P(Q)$:

$$P(Q) = \begin{cases} 1, & \text{if} \quad J(Q) \le J(X) \\ \exp\left(\frac{J(X)-J(Q)}{T}\right), & \text{if} \quad J(Q) > J(X) \end{cases}. \tag{19}$$

Step 4) Increase i by one. Compute J^i using the current solution X.

Step 5) Let the new value of T be $CR \cdot T$.

Step 6) Let $\Delta J^i = |J^{i-1} - J^i| / J^{i-1}$. If $\Delta J^k \le \delta$ for $k = i, i-1, \ldots, i-(N_{stop}+1)$, stop the algorithm.

Otherwise, go to Step 2).

V. PERFORMANCE EVALUATION

The performance of the OSA-based approach and the quality of the obtained FNNM as well as PID controllers are evaluated by comparing with those of the ANFIS-based approach which

TABLE VII
FOUR TEST PLANTS

Plant	Test plants	Plant parameters			
		ω_n	L	τ	ξ
P1	$G(s)=\frac{15}{(s^2+0.9s+5)(s+3)}$	2.2381	0.20	0.3092	0.1999
P2	$G(s)=\frac{18}{(s^2+s+2)(s+3)^2}$	1.2329	0.48	0.3718	0.3650
P3	$G(s)=\frac{3e^{-0.1s}}{(s^2+1.2s+1)(s+3)}$	1.0368	0.48	0.3323	0.6139
P4	$G(s)=\frac{20e^{-2s}}{(s^2+2.4s+4)(s+5)}$	2.0943	2.20	0.8217	0.6000

has the same objective. The same set of 198 training plants from [22] as listed in Table I, called T198, is divided into two sets: T176 consists of the 176 training plants used by ANFIS and T22 consists of the remainder, 22 training plants ($\tau > 0.5$ and $\xi < 0.3$) abandoned by ANFIS. To evaluate the ability of handling the problem of a large number P with an extensive tuning domain, we generate additional 198 training plants from T198 by exchanging the values of ω_n and L. The set consisting of these 198 training plants and T198 is called T396.

In the following computer simulations, the entire program is written by C where a subroutine from MATLAB is used to compute the tracking errors. The obtained FNNM can be evaluated from both training and test performances. Section V-A gives the training performance using various numbers P of training plants. Section V-B gives the test performance using unseen test plants.

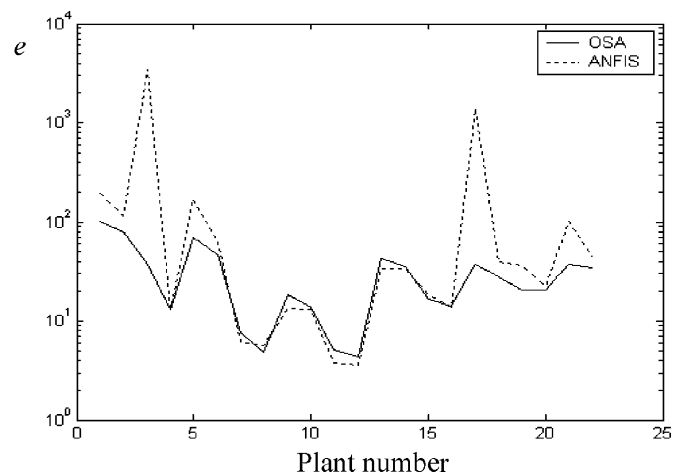

Fig. 6. Tracking error e of individual training plants using T22 as test plants.

A. Training Performance of FNNM

The training performance of FNNM is the criterion for evaluating the optimization ability of the OSA-based approach, which is evaluated in terms of J and P. A high-quality FNNM should be obtained from an extensive tuning domain by sampling a large number P of training plants and can generate PID controllers with a small mean tracking error J.

Each of the three sets T176, T198, and T396 is used as input of OSA to search for a solution X to the investigated problem. Five independent runs are conducted for each set. The mean convergences of OSA for three training sets are shown in Fig. 4, where the numbers of objective function evaluations are 65692, 79948, and 80 029 for T176, T198, and T396, respectively. The values of J for various FNNMs are given in Table V where the reported value 22.17 of J for the ANFIS-based FNNM is obtained using X_{ANFIS}. The tracking errors of individual training plants are shown in Fig. 5. The simulation results reveal the following.

1) The OSA-based FNNM using T176 can improve the ANFIS-based FNNM in averagely decreasing 13.08% error from $J = 22.17$ to 19.27. Fig. 5(a) reveals that the FNNM of OSA has smaller errors for most of training plants than that of ANFIS, where the largest tracking errors for OSA and ANFIS are close to 100.

TABLE VIII
OBTAINED CONTROLLER PARAMETERS FROM THE FNNM OF OSA USING T176

Plant	Controller parameters			
	K	T_i	T_d	b
P1	1.0517	0.5202	0.6045	0.8464
P2	1.0138	1.1402	0.8710	0.8204
P3	1.9951	1.3889	0.6646	0.6416
P4	0.3606	1.2158	0.4360	1.4027

TABLE IX
PERFORMANCE COMPARISONS OF TRACKING ERRORS e_{ANFIS} AND e_{OSA} FOR ANFIS AND OSA, RESPECTIVELY

Plant	P1	P2	P3	P4	Mean
e_{OSA}	3.9574	8.0853	4.6886	14.5149	7.81155
e_{ANFIS}	4.4275	9.6565	5.8205	17.1321	9.25915
$e_{\mathrm{OSA}}/e_{\mathrm{ANFIS}}$	89.38%	83.73%	80.55%	84.72%	84.37%

2) The OSA-based FNNM using T196 has $J = 21.26$, which is better than the ANFIS-based FNNM using T176. Fig. 5(b) shows that the tracking errors of the training plants in T176 and T22 are smaller than 100.

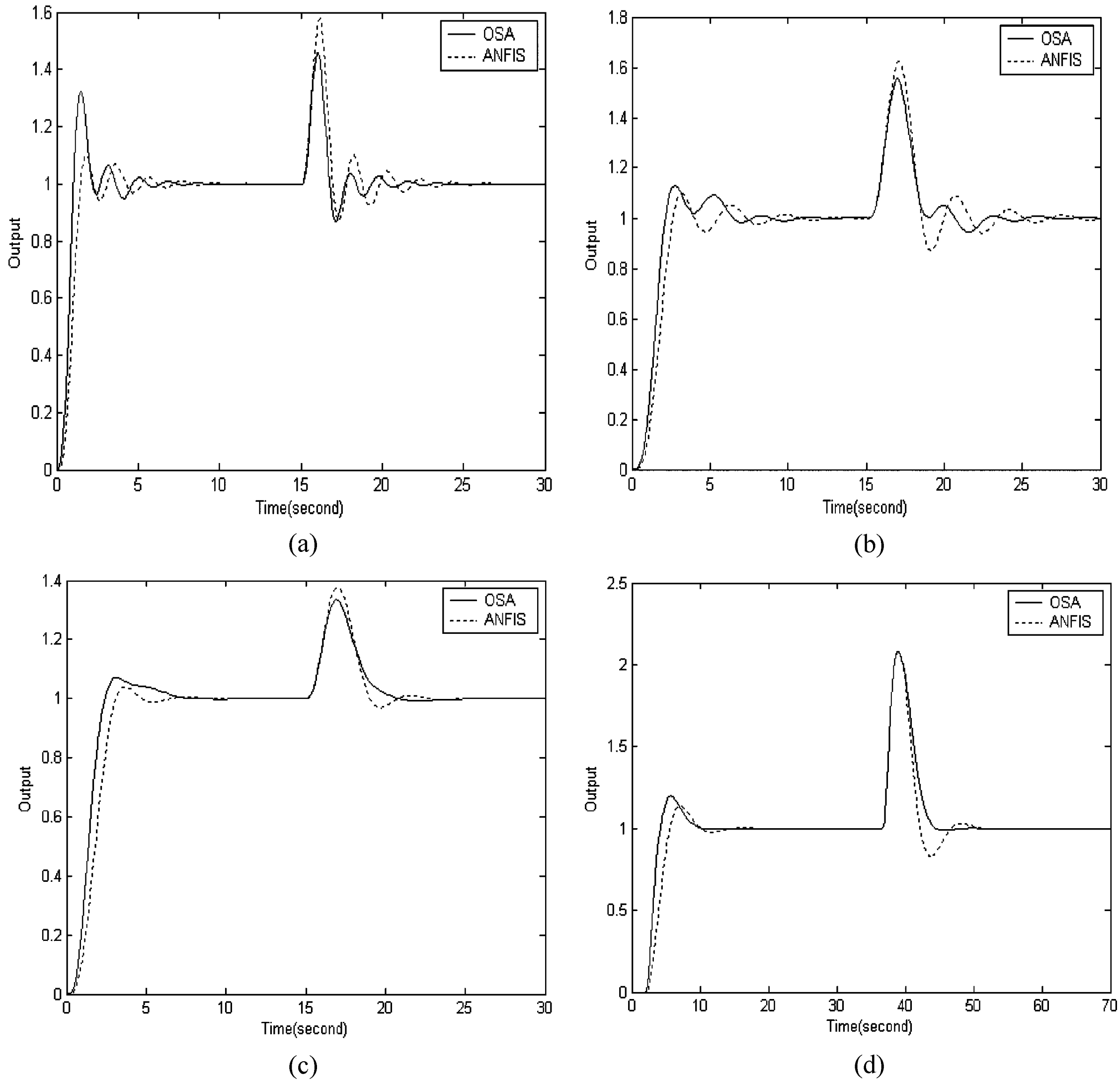

Fig. 7. Set-point and load-disturbance responses of four plants. (a) P1. (b) P2. (c) P3. (d) P4.

It means that the OSA-based approach can cope with the training plants with large dead time, large nature frequency, and poor damping.

3) The mean values of J of OSA for T176, T198, and T396 are 19.27, 21.26, and 20.39, respectively. The OSA-based FNNMs using T198 and T396 have similar good performance with that using T176 in terms of J. The tracking errors for T396 are also smaller than 100, shown in Fig. 5(c). It manifests that increase of the value of P does not result in degrading the performance of the OSA-based FNNM.
4) The standard deviation of J is not very small. To robustly obtain a more accurate solution X, we can adjust the parameters of OSA, such as increase of T_0 and CR for slow convergence. However, it would suffer from long computation time. Fortunately, the optimal design of FNNM is off-line and cost-effective for obtaining an accurate FNNM.

From the simulation results, it reveals that the OSA-based approach is efficient in optimizing fuzzy neural networks to obtain FNNMs for tuning PID controllers.

B. Test Performance of FNNM

To evaluate the test performance of FNNM, we use unseen plant models to obtain corresponding PID controllers from the established FNNM and examine the performance of their control systems. Two test problems are conducted for performance comparisons.

Test Problem 1: Since the FNNM of the ANFIS-based approach is obtained using T176, we use T22 (which are intractable for ANFIS) as the test set to obtain 22 sets of controller parameters from the FNNM obtained from OSA using T176. The individual tracking errors of the 22 plant models are shown in Fig. 6 and Table VI. The mean tracking errors of the 22 plants for ANFIS and OSA are 262.3 and 31.3, respectively. The OSA-based approach can averagely improve 88.07% tracking error. Note that all the obtained systems are stable and most systems of the test plants have tracking errors smaller than 100. The test performance is slightly worse than the training performance for the OSA-based FNNM. It means that no overtraining problem is occurred. From the test problem, the advantage of FNNM is well recognized that the FNNM can fast generate 22 PID controllers on-line from these 22 intractable test plants, relative to the conventional methods [10], [25].

Test Problem 2: For further comparing the quality of the obtained PID controllers with those of other PID controllers obtained from the conventional methods [10], [25], the four specific test plants (P1–P4) used in [22] are used to compare performance, shown in Table VII. The characteristics of the four plants are: 1) P1 is a heavily oscillatory and short apparent dead time plant; 2) P2 is a heavily oscillatory and high-order plant; 3) P3 is a high-order and moderately oscillatory plant; 4) P4 is a high-order and long apparent dead-time plant; and 5) the values of τ and ξ for the four plants do not fall in the range $\tau > 0.5$ and $\xi < 0.3$. The obtained controller parameters from the FNNM of OSA with T176 are given in Table VIII. The tracking errors of OSA and ANFIS based systems are given in Table IX. The set-point and load-disturbance responses of the four controllers are shown in Fig. 7.

The design of optimal PID controllers has two minimization objectives to be simultaneously optimized: 1) robust stability and disturbance attenuation, and 2) tracking error. Since the two competing objectives cannot be evaluated in terms of the same measurement, the optimal controller design problem is essentially a multi-objective optimization problem. In comparing two stable systems with $M_s < 2.0$ (for the first objective), the tracking error is the only quantitative measurement. The tracking error is responsible to the quality of oscillation. To further examine the oscillation performance, the set-point error and load-disturbance are analyzed.

From Table IX, the OSA-based FNNM can generate four PID controllers having a mean tracking error 7.811 55 from the four test plants while the ANFIS-based FNNM has a mean error 9.259 15. The OSA-based FNNM can averagely improve 15.63% tracking errors for the four test plants. From carefully observing the response performance in Fig. 7, the load-disturbance performances of the four OSA-based under-damped systems are all better than those of the ANFIS-based ones. However, the set-point responses of the OSA-based systems are slightly worse than those of the ANFIS-based ones. Note that the four test plants of the ANFIS-based FNNM have better balance performance between set-point and load-disturbance responses than those of the methods [10] and [25]. Therefore, the simulation results show that the proposed approach can obtain the FNNM providing high-quality PID controllers with a good balance between the set-point and load-disturbance responses.

VI. Conclusion

In this paper, we formulated an optimization problem of establishing an FNNM for efficiently tuning PID controllers of various test plants with under-damped responses using a large number P of training plants such that the mean tracking error J of the obtained P control systems is minimized. At the same time, we proposed a novel orthogonal simulation annealing algorithm OSA to solve the large-scale constrained optimization problem. High performance of the OSA-based FNNM arises from two aspects: 1) the tracking errors and stability constraints are directly embedded in the objective function for tuning PID controllers, and 2) OSA can effectively search for a near-optimal feasible solution to the large-scale parameter optimization problem. The benefits of using OSA to optimize FNNMs are also analyzed. It has been shown that the OSA-based FNNM with $P = 176$ can improve the ANFIS-based FNNM in averagely decreasing 13.08% error J and 88.07% tracking error of 22 unseen test plants. The OSA-based FNNM using four specific test plants can averagely improve 15.63% tracking errors of the under-damped systems. Furthermore, the OSA-based FNNMs using $P = 198$ and 396 from an extensive tuning domain have similar good performance with that using $P = 176$ in terms of J. The simulation results also show that the proposed approach can obtain the FNNM providing high-quality PID controllers with good oscillation performance as well as a good balance between the set-point and load-disturbance responses.

References

[1] K. J. Astrom, T. Hagglund, C.-C. Hang, and W.-K. Ho, "Automatic tuning and adaptation for PID controllers- a survey," *IFAC J. Control Eng. Practice*, vol. 1, no. 4, pp. 699–714, 1993.

[2] K. J. Astrom, *PID Controller: Theory, Design, and Tuning*. Research Triangle Park, NC: ISA, 1995.

[3] T. P. Bagchi, *Taguchi Methods Explained: Practical Steps to Robust Design*. Upper Saddle River, NJ: Prentice-Hall, 1993.

[4] T. Chai and G. Zhang, "A new self-tuning for PID regulators based on phase and amplitude margin specifications," *ACTA Automatica Sinica*, vol. 23, no. 2, pp. 167–172, 1997.

[5] B.-S. Chen and Y.-M. Cheng, "A structure-specified H_∞ optimal control design for practical applications: a genetic approach," *IEEE Trans. Control Syst. Technol.*, vol. 6, no. 6, pp. 707–718, Nov. 1998.

[6] A. Dey, *Orthogonal Fractional Factorial Designs*. New York: Wiley, 1985.

[7] P. J. Fleming and R. C. Purshouse, "Evolutionary algorithms in control systems engineering: a survey," *Control Eng. Practice*, vol. 10, pp. 1223–1241, 2002.

[8] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. Reading, MA: Addison-Wesley, 1989.

[9] A. S. Hedayat, N. J. A. Sloane, and J. Stufken, *Orthogonal Arrays: Theory and Applications*. New York: Springer-Verlag, 1999.

[10] W.-K. Ho, C.-C. Hang, and J. Zhou, "Self-tuning PID control of a plant with under-damped response with specifications on gain and phase margins," *IEEE Trans. Control Syst. Technol.*, vol. 5, no. 4, pp. 446–452, Jul. 1997.

[11] S.-J. Ho, S.-Y. Ho, and L.-S. Shu, "OSA: orthogonal simulated annealing algorithm and its application to designing mixed $\mathrm{H}_2/\mathrm{H}_\infty$ optimal controllers," *IEEE Trans. Syst., Man, Cybern. A, Syst. Humans*, vol. 34, no. 5, pp. 588–600, Sep. 2004.

[12] S.-Y. Ho, S.-J. Ho, Y.-K. Lin, and C.-C. W. Chu, "An orthogonal simulated annealing algorithm for large floorplanning problems," *IEEE Trans. Very Large Scale (VLSI) Syst.*, vol. 12, no. 8, pp. 874–876, Aug. 2004.

[13] S.-Y. Ho, L.-S. Shu, and J.-H. Chen, "Intelligent evolutionary algorithms for large parameter optimization problems," *IEEE Trans. Evol. Comput.*, vol. 8, no. 6, pp. 522–541, Dec. 2004.

[14] K. KrishnaKumar, S. Narayanaswamy, and S. Garg, "Solving large parameter optimization problems using a genetic algorithm with stochastic coding," in *Genetic Algorithms in Engineering and Computer Science*, G. Winter, J. Périaux, M. Galán, and P. Cuesta, Eds. New York: Wiley, 1995, pp. 287–303.

[15] R. A. Krohling and J. P. Rey, "Design of optimal disturbance rejection PID controllers using genetic algorithms," *IEEE Trans. Evol. Comput.*, vol. 5, no. 1, pp. 78–82, Feb. 2001.

[16] Y.-W. Leung and Y. Wang, "An orthogonal genetic algorithm with quantization for global numerical optimization," *IEEE Trans. Evol. Comput.*, vol. 5, no. 1, pp. 41–53, Feb. 2001.

[17] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *J. Chem. Phys.*, vol. 21, no. 6, pp. 1087–1092, 1953.

[18] M. S. Phadke, *Quality Engineering Using Robust Design*. Englewood Cliffs, NJ: Prentice-Hall, Inc., 1989.

[19] T. Renyuan, S. Jianzhong, and L. Yan, "Optimization of electromagnetic devices by using intelligent simulated annealing algorithm," *IEEE Trans. Magn.*, vol. 34, no. 5, pp. 2992–2995, Sep. 1998.

[20] J. S. R. Jang and C. T. Sun, "Neuro-fuzzy modeling and control," *Proc. IEEE*, vol. 83, no. 3, pp. 378–406, Mar. 1995.

[21] J. S. R. Jang, "ANFIS: adaptive-network-based fuzzy inference system," *IEEE Trans. Syst., Man, Cybern.*, vol. 23, no. 5, pp. 665–685, May–Jun. 1993.

[22] J.-C. Shen, "Fuzzy neural networks for tuning PID controller for plants with under-damped responses," *IEEE Trans. Fuzzy Syst.*, vol. 9, no. 2, pp. 333–342, Apr. 2001.

[23] H. Szu and R. Hartley, "Fast simulated annealing," *Phys. Lett.*, vol. 122, pp. 157–162, 1987.

[24] X. Tang, R. Tian, and D. F. Wong, "Fast evaluation of sequence pair in block placement by longest common subsequence computation," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 20, no. 12, pp. 1406–1413, Dec. 2001.

[25] Q. Wang, T. Lee, H. Fung, Q. Bi, and Y. Zhang, "PID tuning for improved performance," *IEEE Trans. Control Syst. Technol.*, vol. 7, no. 4, pp. 457–465, Jul. 1999.

[26] D. F. Wong, H. W. Leong, and C. L. Liu, *Simulated Annealing for VLSI Design*. Norwell, MA: Kluwer, 1988.

[27] Q. Wu, "On the optimality of orthogonal experimental design," *Acta Math. Appl. Sinica*, vol. 1, no. 4, pp. 283–299, 1978.

[28] S. Yang, J. M. Machado, G. Ni, S.-L. Ho, and P. Zhou, "A self-learning simulated annealing algorithm for global optimizations of electromagnetic devices," *IEEE Trans. Magn.*, vol. 36, no. 7, pp. 1004–1008, Jul. 2000.

[29] M. Zhaung and D. P. Atherton, "Automatic tuning of optimal PID controllers," *Proc. Inst. Elect. Eng., D*, vol. 140, pp. 216–224, 1993.

Shinn-Jang Ho (M'03) was born in Taiwan, R.O.C., in 1960. He received the B. S. degree in power mechanic engineering from National Tsing Hua University, Hsinchu, Taiwan, in 1983, and the M.S. and Ph.D. degrees in mechanical engineering from National Sun Yat-Sen University, Kaohsiung, Taiwan, in 1985 and 1992, respectively.

He is currently a Professor in the Department of Automation Engineering, National Formosa University, Huwei, Yulin, Taiwan. His research interests include optimal control, fuzzy system, evolutionary algorithms, and system optimization.

Li-Sun Shu (S'04) was born in Taiwan, R.O.C., in 1972. He received the B.A. degree in mathematics from Chung Yuan University, Chung Li, Taiwan, in 1995, and the M.S. and Ph.D. degrees in information engineering and computer science from Feng Chia University, Taichung, Taiwan, in 1997 and 2004, respectively.

He is currently an Assistant Professor at Overseas Chinese Institute of Technology. His research interests include evolutionary computation, large parameter optimization problems, fuzzy systems, and system optimization.

Shinn-Ying Ho (M'00) was born in Taiwan, R.O.C. in 1962. He received the B. S., M. S., and Ph.D. degrees in computer science and information engineering from National Chiao Tung University, Hsinchu, Taiwan, in 1984, 1986, and 1992, respectively. From 1992 to 2004, he was with the Department of Information Engineering and Computer Science at Feng Chia University, Taichung, Taiwan.

He is currently a professor in the Department of Biological Science and Technology and Institute of Bioinformatics, National Chiao Tung University, Hsin-chu, Taiwan. His research interests include evolutionary algorithms, image processing, pattern recognition, bioinformatics, data mining, virtual reality applications of computer vision, fuzzy classifier, large parameter optimization problems, and system optimization.