Neuro-Fuzzy Generalized Predictive Control of Boiler Steam Temperature

X.-J. Liu and C. W. Chan

Abstract—**Reliable control of superheated steam temperature is necessary to ensure high efficiency and high load-following capability in the operation of modern power plant. This is often difficult to achieve using conventional PI controllers, as power plants are nonlinear and contain many uncertainties. A nonlinear generalized predictive controller based on neuro-fuzzy network (NFGPC) is proposed in this paper, which consists of local GPCs designed using the local linear models of the neuro-fuzzy network that models the plant. The proposed nonlinear controller is applied to control the superheated steam temperature of a 200-MW power plant. From the experiments on the plant and the simulation of the plant, much better performance than the traditional cascade PI controller or the linear GPC is obtained.**

Index Terms—**Generalized predictive control, neuro-fuzzy networks, superheated steam temperature.**

I. Introduction

CONTINUOUS processes in power plant and power station are complex systems characterized by nonlinearity, uncertainty, and load disturbance. The superheater is an important part of the steam generation process in the boiler-turbine system, where steam is superheated before entering the turbine that drives the generator. Not only the steam generation process is highly nonlinear, the temperature and the pressure in the superheater are extremely high. Therefore, controlling superheated steam temperature is not only technically challenging, but also economically important [1].

In [2], a simplified nonlinear model of a drum boiler-turbine unit is obtained from a series of experiments. A fourth order nonlinear model of the boiler with time delays, measurement noise, and load disturbances was presented in [3]. This model provides a useful basis for developing control strategies for the boiler. More recently, artificial intelligence has been applied to model steam-boiler systems, e.g., the identification of boiler models discussed in [4], and the modeling of a 200-MW boiler system based on neural networks [5].

The proportional-integral (PI) controllers are still the most popular controller for controlling steam-boiler systems, and are commonly available in the distributed control system (DCS) installed in the control room of power stations. The PI controllers are usually tuned by experts heuristically by trials and errors. Not only it is time consuming, the performance of the controllers is not always satisfactory, especially when there are large changes in the load or the operating point. Cascade PI controllers tuned for a fixed load are common for controlling steam temperature in a power plant. However, when the load changes are large, the human operation is often required to replace the PI controllers for a better performance.

The multivariable multistep adaptive regulator has been applied to control the superheated steam temperature in a 150 T/h boiler [1], and generalized predictive control was proposed to control the steam temperature [6]. A nonlinear long-range predictive controller based on neural networks is developed in [7] to control the main steam temperature and pressure, and the reheated steam temperature at several operating levels. Plant nonlinearity was accounted for without resorting to on-line parameter-estimation as in self-tuning control. The control of the main steam pressure and temperature based on a nonlinear model that consists of nonlinear static constants and linear dynamics is presented in [8].

A well-designed nonlinear controller based on a nonlinear model of the process may perform well, tremendous computing time, however, is required to update the parameters of the nonlinear model and the controller, making it unsuitable for on-line adaptation. Further, the lack of accurate and detailed technical data also makes it difficult to produce a sufficiently accurate nonlinear model of the industrial process to be used in designing the nonlinear controller.

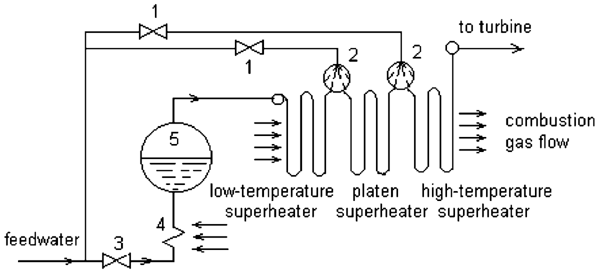
In practice, the complex power plant is often controlled manually by experienced operators based on their knowledge of the plant, when the range of the load change is large. As fuzzy logic is capable of incorporating human experiences via the fuzzy rules, on-line self-organizing fuzzy logic controllers have been proposed for controlling boiler-turbine systems [9]. In addition to direct control applications, fuzzy logics have also been used in designing specifications. The design of fuzzy logic controllers is somehow time consuming, as the fuzzy rules are often obtained by trials and errors. In contrast, neural networks not only have the ability to approximate nonlinear functions with arbitrary accuracy, they can also be trained from experimental data. However, the training of neural networks is computing intensive, restricting their online applications. The neuro-fuzzy networks (NFNs) developed recently have the advantages of model transparency of fuzzy logic, and learning capability of neural networks [10]. Since NFNs are linear-in-weights networks, they can be trained using linear least squares method, and can be further simplified to reduce significantly the computing time for on-line applications [11]. The NFNs have been used to develop self-tuning control [12], [13], and is therefore

Manuscript received February 11, 2003; revised November 5, 2004. This work was supported by in part National Natural Science Foundation of China under Grant 69804003 and in part by Beijing Natural Science Foundation under Grant 4062030, Paper no. TEC-00033-2003.

X.-J. Liu is with Department of Automation, North China Electric Power University, Beijing 102206, China (e-mail: liuxj@ncepu.edu.cn).

C. W. Chan is with Department of Mechanical Engineering, The University of Hong Kong, Hong Kong (e-mail: mechan@hkucc.hku.hk).

Digital Object Identifier 10.1109/TEC.2005.853758



1--- superheater spray valve 2---superheater spray 3---feedwater valve 4---economiser 5---drum

Fig. 1. Boiler and superheater steam generation process.

a useful tool for developing nonlinear predictive control [14]. To use predictive control, the plant must be reasonably linear near some local specified operating points. Since NFNs can be considered as a network that consists of several local regions, each of which contains a local linear model, nonlinear predictive control based on NFNs can be devised with the network incorporating all the local generalized predictive controllers (GPC) designed using the respective local linear models. Following this approach, the nonlinear generalized predictive controllers based on the NFNs, or simply, the neuro-fuzzy generalized predictive controllers (NFGPCs) are derived here. The proposed controller is then applied to control the superheated steam temperature of the 200-MW power unit. Experimental data obtained from the plant are used to train the NFN model, and from which local GPCs that form part of the NFGPC is then designed. The proposed controller is tested first on the simulation of the process, before applying it to control the power plant.

II. Plant Description

The power plant considered in this paper is a pulverized coal-firing steam-boiler generation unit rated at 200 MW. The maximum steam consumption of the power plant is 670 T/h at a superheated steam pressure and temperature of 16.7 MPa and 540 °C, respectively. A two stage water sprayers is used to control the superheated temperature. The objective is to control the superheated steam temperature by controlling the flow of spray water using the spray water valves.

From Fig. 1, the steam generated from the boiler drum passes through the low-temperature superheater before it enters the radiant-type platen superheater. Water is sprayed onto the steam to control the superheated steam temperature in both the low and high temperature superheaters. Proper control of the superheated steam temperature is extremely important to ensure the overall efficiency and safety of the power plant, as the temperature in the high temperature superheater is the highest in the plant. It is undesirable that the steam temperature is too high, as it can damage the superheater and the high pressure turbine, or too low, as it will lower the efficiency of the power plant. Therefore, the superheated steam temperature is to be controlled by adjusting the flow of spray water to within ±10 °C during the transient states, and ±5 °C at the steady state. It is also important to reduce the temperature fluctuations inside the superheater, as it helps to minimize mechanical stress that causes micro-cracks in the unit, in order to prolong the life of the unit and to reduce maintenance costs. As the GPC is derived by minimizing these fluctuations, it is amongst the controllers that are most suitable for achieving this goal.

There are three main factors that can affect the superheated steam temperature: load, gas flow, and inlet steam temperature of the superheater. Other factors include steam flow, feedwater temperature, enthalpy of steam entering the superheater and combustion gas temperature. Unpredictable disturbances, such as dust deposition on the furnace walls and sedimentation in the steam pipe can also affect the heat transfer coefficients, and consequently, the operation of the superheater. This type of variations is slow and can be catered for by updating periodically the model of the superheater.

As the steam generation process is complex and highly nonlinear, the performance of the conventional PID-based controllers is often unsatisfactory. The proposed NFGPC trained by experimental data, and incorporating also the experiences of the operator is used to control the superheated steam temperature of this 200-MW power generation unit.

III. NFN Modelling

A. Structure of the Nonlinear Predictive Model by NFN

Consider the following general single-input single-output nonlinear dynamic system:

$$\begin{aligned} y(t) = f[&y(t-1), \ldots, y(t-n'_y), u(t-d), \ldots, \\ &u(t-d-n'_u+1), e(t-1), \ldots, e(t-n'_e)] + e(t)/\Delta \end{aligned} \tag{1}$$

where $f[\cdot]$ is a smooth nonlinear function such that a Taylor series expansion exists, $e(t)$ is a zero mean white noise and Δ is the differencing operator, n'_y, n'_u, n'_e, and d are, respectively, the known orders and time delay of the system. Let the local linear model of the nonlinear system (1) at the operating point $\mathrm{O}(t)$ be given by the following controlled auto-regressive integrated moving average (CARIMA) model:

$$A(z^{-1})y(t) = z^{-d}B(z^{-1})u(t) + C(z^{-1})e(t)/\Delta$$

or

$$\bar{A}(z^{-1})y(t) = z^{-d}B(z^{-1})\Delta u(t) + C(z^{-1})e(t) \tag{2}$$

where $\bar{A}(z^{-1}) = \Delta A(z^{-1}), B(z^{-1})$, and $C(z^{-1})$ are polynomials in z^{-1}, the backward shift operator. Note that the coefficients of these polynomials are a function of the operating point $\mathrm{O}(t)$. The nonlinear system (1) is partitioned into several operating regions, such that each region can be approximated by a local linear model. Since NFNs is a class of associative memory networks with knowledge stored locally [10], they can be applied to model this class of nonlinear systems. A schematic diagram of the NFN is shown in Fig. 2, where the membership functions are given by B-spline basis functions. The input of the network is the antecedent variable $[x_1, x_2 \ldots x_n]$, and the output, $\hat{y}(t)$, is a weighted sum of the output of the local linear models $\hat{y}_i(t)$. A property of the NFNs is that the network stores information and learns locally, and that *a priori*

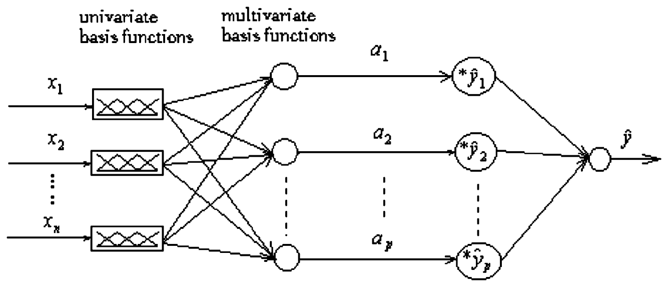


Fig. 2. Neuro-fuzzy network.

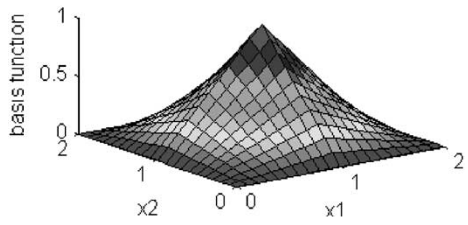


Fig. 3. Multivariate basis function.

knowledge of the process can be incorporated into the design of the network. B-spline functions are used as the membership functions in the NFNs for the following reasons [10]. First, B-spline functions can be readily specified by the order of the basis function and the number of inner knots. Second, they are defined on a bounded support, and the output of the basis function is always positive, i.e., $\mu_k^j(x) = 0, x \notin [\lambda_{j-k}, \lambda_j]$ and $\mu_k^j(x) > 0, x \in (\lambda_{j-k}, \lambda_j)$. Third, the basis functions form a partition of unity, i.e., $\sum_j \mu_k^j(x) \equiv 1, x \in [x_{\min}, x_{\max}]$. And fourth, the output of the basis functions can be obtained by a recurrence equation.

It is shown in [10] that for membership functions of fuzzy variables given by univariate B-spline basis functions, the membership functions of the fuzzy variables derived from the fuzzy rules can be obtained by the tensor product of the univariate basis functions. As an example, consider the NFN shown in Fig. 2, which consists of the following fuzzy rules:

IF operating condition i (x_1 is positive small, …, and x_n is negative large).

THEN the output is given by the local CARIMA model i

$$\begin{aligned}\hat{y}_i(t) &= \bar{a}_{i1}\hat{y}_i(t-1) + \cdots \bar{a}_{in_{\bar{a}}}\hat{y}_i(t-n_{\bar{a}})\\ &\quad + b_{i0}\Delta u_i(t-d) + \cdots + b_{in_b}\Delta u_i(t-d-n_b)\\ &\quad + e_i(t) + \cdots + c_{in_c}e_i(t-n_c)\end{aligned}$$

or

$$\bar{A}_i(z^{-1})\hat{y}_i(t) = z^{-d}\Delta B_i(z^{-1})u_i(t) + C_i(z^{-1})e_i(t) \quad (3)$$

where $\bar{A}_i(z^{-1}), B_i(z^{-1})$, and $C_i(z^{-1})$ are polynomials in the backward shift operator z^{-1}, and d is the dead time of the plant, $u_i(t)$ is the control, and $e_i(t)$ is a zero mean independent random variable with a variance of σ^2. The multivariate basis function $a_i(x_k)$, or the transformed input vector, is obtained by the tensor products of the univariate basis functions $\mu_{A_k^i}(x_k)$, as follows:

$$a_i = \prod_{k=1}^{n} \mu_{A_k^i}(x_k), \qquad \text{for } i = 1, 2, \ldots, p \quad (4)$$

where n is the dimension of the input vector x, and p, the total number of weights in the NFN, is given by

$$p = \prod_{i=1}^{n}(R_i + k_i) \quad (5)$$

where k_i and R_i are the order of the basis function and the number of inner knots respectively. The properties of the univariate B-spline basis functions described previously also apply to the multivariate basis functions, which is defined on the hyper-rectangles, $(k_1 \times k_2 \times \cdots \times k_n)$. Similarly, the output of the multivariate basis functions is positive inside this domain and zero elsewhere. An example of the multivariate basis function formed from two second-order univariate basis functions is shown in Fig. 3.

Assuming the center of gravity defuzzification method is used, the output of the NFN with p fuzzy rules is

$$\hat{y} = \frac{\sum_{i=1}^{p}\hat{y}_i a_i}{\sum_{i=1}^{p} a_i} = \sum_{i=1}^{p}\hat{y}_i\alpha_i \quad (6)$$

B. Learning of the NFN

Consider a quadratic cost function

$$I = \frac{1}{2}e^2 = \frac{1}{2}(\hat{y} - y)^2 \quad (7)$$

where $\hat{y}$ is the network output, and y is the plant output. The parameters of $\bar{A}_i(z^{-1})$ and $B_i(z^{-1})$, the local linear model are updated proportional to the negative gradient of I, as

$$\dot{\bar{a}}_{il} = -g\frac{\partial I}{\partial \bar{a}_{ij}} = -ge\frac{\partial \hat{y}}{\partial \hat{y}_i}\frac{\partial \hat{y}_i}{\partial \bar{a}_{ij}} = -ge\alpha_i(x)\hat{y}_i(t-l) \quad (8)$$

$$\dot{b}_{im} = -g\frac{\partial I}{\partial b_{il}} = -ge\frac{\partial \hat{y}}{\partial \hat{y}_i}\frac{\partial \hat{y}_i}{\partial b_{il}} = -ge\alpha_i(x)u_i(t-d-m) \quad (9)$$

where $g > 0$ is the learning rate, $1 \leq l \leq n_{\bar{a}}$, and $0 \leq m \leq n_b$. In the on-line learning law (8) and (9), the initial value of the weights should be chosen to ensure a fast convergence of the estimated parameters to some constant values. To achieve this, *apriori* information and experiences about the plant should be used as far as possible.

Define the performance index

$$L(t) = \frac{1}{W}\sum_{t=1}^{W}\left|\frac{\hat{y}(t) - y(t)}{y(t)}\right| \quad (10)$$

where W is the number of data. This index will be used later to evaluate the performance of the models.

IV. Nonlinear Neuro-Fuzzy Network Generalized Predictive Control

In this section, the NFGPC is derived based on the NFN presented in Section III.

A. Derivation of NFGPC

The GPC is obtained by minimizing the following cost function [15]:

$$J = E\left\{\sum_{j=d}^{N} q_j[\hat{y}(t+j) - y_r(t+j)]^2\right\} + \sum_{j=1}^{M} \lambda_j[\Delta u(t+j-1)]^2 \tag{11}$$

where q_j and λ_j are, respectively, the weighting factors for the prediction error and the control, $y_r(t+j)$ is the jth step ahead reference trajectory, d is the minimum costing horizon, N and M are, respectively, the maximum costing horizon for the prediction error and the control. The control computed from the NFGPC is the weighted sum of the control obtained from p local GPC controllers

$$\Delta u(t) = \sum_{i=1}^{p} a_i \Delta u_i(t) \tag{12}$$

where $\Delta u_i(t)$ is the control in the ith region, $a_i(x)$ is defined previously in (4). Note that the weights in the NFGPC are identical to that in the NFN that models the process. Although the control computed by (12) is similar to the Takagi-Sugeno fuzzy model [16], it is different in that the "consequence" does not necessarily contain the "premise" component. Since switching between local GPC controllers in the NFGPC involves fuzzy logics, it can be interpreted not only as a fuzzy controller, but also as a fuzzy supervisor. Since switching between local controllers is achieved through fuzzy logics, the control can be smooth if the weights $a_i(x)$ are suitably selected. From the NFN (6) and the control (12), J given by (11) can be rewritten as

$$J = E\left\{\sum_{j=d}^{N} q_j\left[\sum_{i=1}^{p} \alpha_i(\hat{y}_i(t+j) - y_r(t+j))\right]^2\right\} + \sum_{j=1}^{M} \lambda_j\left[\sum_{i=1}^{p} \alpha_i \Delta u_i(t+j-1)\right]^2. \tag{13}$$

Because of the interactions between the sub-systems, minimizing this cost function requires tremendous computation. To reduce the amount of computation, the cost function is simplified first using the Cauchy inequality [17]. Since

$$\left[\sum_{i=1}^{p} \alpha_i(\hat{y}_i(t+j) - y_r(t+j))\right]^2 \leq p\sum_{i=1}^{p}[\alpha_i(\hat{y}_i(t+j) - y_r(t+j))]^2$$

hence

$$\left[\sum_{i=1}^{p} \alpha_i \Delta u_i(t+j-1)\right]^2 \leq p\sum_{i=1}^{p}[\alpha_i \Delta u_i(t+j-1)]^2. \tag{14}$$

Equation (14) implies that the sum of the weighted squared errors can be an upper bound of the cost function J. Rewriting (13) gives

$$\begin{aligned} &E\left\{\sum_{j=d}^{N}\sum_{i=1}^{p} q_j[\alpha_i(\hat{y}(t+j) - y_r(t+j))]^2\right\} + \sum_{j=1}^{M}\sum_{i=1}^{p} \lambda_j[\alpha_i \Delta u_i(t+j-1)]^2 \\ &= E\left\{\sum_{i=1}^{p}(\alpha_i)^2 \sum_{j=d}^{N} q_j[\hat{y}_i(t+j) - y_r(t+j)]^2\right\} + \sum_{i=1}^{p}(\alpha_i)^2 \sum_{j=1}^{M} \lambda_j[\Delta u_i(t+j-1)]^2 \\ &= \sum_{i=1}^{p}(\alpha_i)^2 J_i \end{aligned} \tag{15}$$

where

$$J_i = E\left\{\sum_{j=d}^{N} q_j[\hat{y}_i(t+j) - y_r(t+j)]^2\right\} + \sum_{j=1}^{M} \lambda_j[\Delta u_i(t+j-1)]^2. \tag{16}$$

Equation (15) shows that minimizing J_i is essentially the same as that of minimizing J. From (16), a set of local generalized predictive controllers is obtained, which forms part of the NFGPC.

B. Local GPC

The local GPC [15] is given by

$$\Delta U_i(t) = \left(G_i^T Q G_i + \lambda\right)^{-1} G_i^T Q[Y_r(t+1) - F_i \Delta U_i(t-1) - S_i(z^{-1}) y_i(t)] \tag{17}$$

where

$$\begin{aligned} Y_i(t+1) &= [\hat{y}_i(t+1), \hat{y}_i(t+2), \ldots, \hat{y}_i(t+N)]^T \\ Y_r(t+1) &= [\hat{y}_r(t+1), \hat{y}_r(t+2), \ldots, \hat{y}_r(t+N)]^T \\ \Delta U_i(t) &= [\Delta u_i(t), \Delta u_i(t+1), \ldots, \Delta u_i(t+M-1)]^T \\ \Delta U_i(t-1) &= [\Delta u_i(t-n_b), \Delta u_i(t-n_b+1), \ldots, \\ &\qquad \Delta u_i(t-1)]^T \\ \varepsilon_i(t+1) &= [R_{i1}(z^{-1}) e_i(t+1), \ldots R_{iP}(z^{-1}) e_i(t+N)]^T \\ S_i(z^{-1}) &= [S_{i1}(z^{-1}), S_{i2}(z^{-1}), \ldots, S_{iP}(z^{-1})]^T \end{aligned}$$

$S_i(z^{-1})$ and $R_i(z^{-1})$ satisfy the Diophantine equation

$$1 = \bar{A}_i(z^{-1}) R_{ij}(z^{-1}) + z^{-j} S_{ij}(z^{-1}) \tag{18}$$

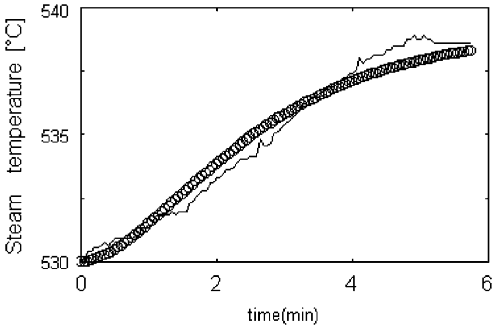


Fig. 4. Step response of superheated steam temperature (solid line) for a 18% step change in spray water flow to the high temperature superheater and the modeling result (circled line).

and

$$G_i(z^{-1}) = B_i(z^{-1})R_i(z^{-1}) = g_{j,0} + g_{j,1}z^{-1} + \cdots + g_{j,n_b+j-1}z^{-(n_b+j-1)} \tag{19a}$$

$$g_{j+1,i} = g_{j,i} + s_{j,0}b_{i-j} \tag{19b}$$

$$Q_i = \mathrm{diag}(q_{i1}, q_{i2}, \ldots, q_{iN})$$

$$T_i = \mathrm{diag}(\lambda_{i1}, \lambda_{i2}, \ldots, \lambda_{iM}) \tag{19c}$$

$$G_i = \begin{bmatrix} g^i_{1,0} & & \cdots & 0 \\ g^i_{2,1} & g^i_{1,0} & & \vdots \\ \vdots & \vdots & & \\ g^i_{N,N-1} & g^i_{N-1,N-2} & \cdots & g^i_{N-M+1,N-M} \end{bmatrix}_{N\times M} \tag{19d}$$

$$F_i = \begin{bmatrix} g^i_{1,n_b} & g^i_{1,n_b-1} & \cdots & g^i_{1,2} & g^i_{1,1} \\ g^i_{2,n_b+1} & g^i_{2,n_b} & \cdots & g^i_{2,3} & g^i_{2,2} \\ \vdots & \vdots & & \vdots & \vdots \\ g^i_{N,n_b+N-1} & g^i_{N,n_b+N-2} & \cdots & g^i_{N,N+1} & g^i_{N,N} \end{bmatrix}_{N\times n_b}. \tag{19e}$$

The optimized M steps ahead control is computed, and only the first step ahead control is implemented, using a receding horizon principle [15], giving

$$\Delta u_i(t) = d^T_{i1}[Y_r(t+1) - F_i\Delta U_i(t-1) - S_i(z^{-1})y_i(t)] \tag{20}$$

where $d^T_{i1} = (1, 0, \ldots, 0)(G^T_i QG_i + \lambda)^{-1}G^T_i Q$ is the first row of the matrix $(G^T_i QG_i + \lambda)^{-1}G^T_i Q$.

V. Neuro-Fuzzy Modeling and Predictive Control of Superheated Steam Temperature

A. *Neuro-Fuzzy Modeling of Superheated Steam Temperature*

Let θ be the superheated steam temperature, and μ_θ, the flow of spray water to the high temperature superheater. The response of θ to a step change of 18% in μ_θ under 180 MW load condition is shown in Fig. 4. Clearly, θ can be approximated by a second order model [18]

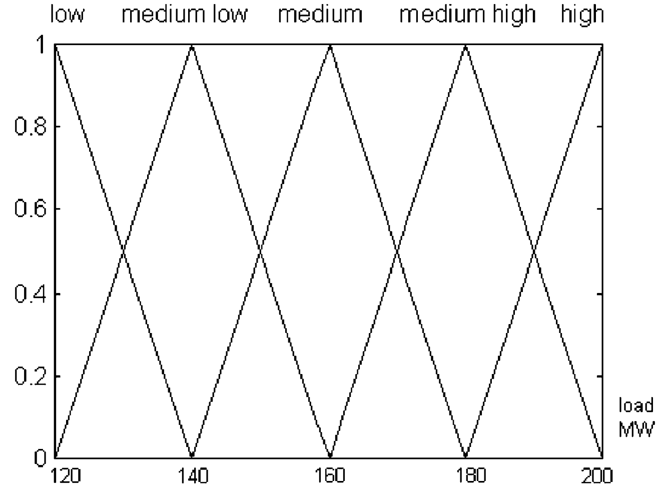


Fig. 5. Membership functions for local models.

$$G(s) = \frac{\theta}{\mu_\theta(s)} = \frac{K_p}{(T_1s+1)(T_2s+1)}e^{-\tau s} \tag{21}$$

where $T_1 = 185$ s, $T_2 = 175$ s under this load condition. Equation (21) can be written in discrete form as shown in (2), where $\bar{A}(z^{-1}) = \Delta A(z^{-1})$ is a third-order polynomial.

The linear model (21) is, however, only a local model for the selected operating point. As discussed previously, a number of these linear models are used in the neuro-fuzzy model given by (3).

Since load is the unique antecedent variable, it is used to select the division between the local regions in the NFN. Based on this approach, the load is divided into five regions as shown in Fig. 5, using also the experience of the operators, who regard a load of 200 MW as high, 180 MW as medium high, 160 MW as medium, 140 MW as medium low, and 120 MW as low. The power plant is operated usually near the low end of the load at 140 MW during the night and the high end at 200 MW during the day. Since a load less than 120 MW occurs only during the start-up and the run-down of the power plant, it is therefore not included in the modeling. This type of partitioning ensures that any change in the real-valued input signal will be reflected by a change in the degree of membership. Notice that uniform division was chosen for ease of implementation, as nonuniform division would involve a lot more trials and errors, which is not feasible for a real plant.

The data for training the neuro-fuzzy model should ideally contain as much information as possible on the dynamics of the process, as the generalization ability of the NFN depends on the quality of the training data. For this reason, it is important that the experiments should be designed to cover sufficiently the full operation region of the power plant. It may be argued that data for estimating the parameters of the model should be generated with the plant under persistent excitation. However, it may not be desirable or acceptable to put the plant under too much stress. Further, as the dynamics of the plant is well understood by the

TABLE I
LOCAL CARIMA MODELS IN NEURO-FUZZY MODEL

Load	$\overline{A}_i$	B_i
High	$1-2.8824z^{-1}+2.7682z^{-2}-0.8858z^{-3}$	0.0035
Medium high	$1-2.8940z^{-1}+2.7915z^{-2}-0.8975z^{-3}$	0.0028
Medium	$1-2.9000z^{-1}+2.8100z^{-2}-0.9100z^{-3}$	0.0023
Medium low	$1-2.9130z^{-1}+2.8279z^{-2}-0.9149z^{-3}$	0.0019
Low	$1-2.9200z^{-1}+2.8416z^{-2}-0.9216z^{-3}$	0.0016

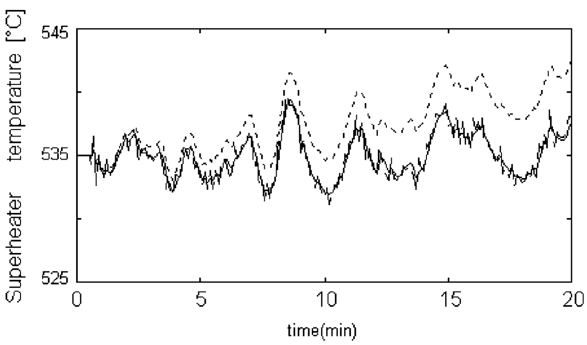


Fig. 6. Actual superheated steam temperature (solid line) versus linear model (dotted line) and NFN model (dashed line).

operators, sufficiently reliable estimate of the parameters of the plant model can be obtained from step changes in the input. In order that the neurofuzzy model of the plant can approximate the plant, the experiment is repeated over the full operating range of the plant. The NFN that models the power plant is then trained by the gradient method (8) from the training data.

Since the length of the superheater tube is measured in tens of meters, a time delay therefore exists between the spray water flow and the superheated steam temperature at the outlet of the superheater. This time delay is extremely difficult to determine experimentally, due to the measurement noise and the dynamic of the process. A reasonable estimate of this time delay τ is 30 s, and is used as the minimum control horizon. Assume $e_i(t)$ is a zero mean uncorrelated random sequence, then $C_i = 1$. For a sampling interval of 30 s, the estimated linear local models $\bar{A}(z^{-1})$ used in the NFN are shown in Table I.

From Fig. 6, good generalization result of the superheated steam temperature over the full operating range is obtained from the NFN. In contrast, the linear model estimated using the same data can only give good approximation over certain, but not the full operating range, illustrating that the plant is highly nonlinear. The performance index (10) is computed for both models. The one for the NFN is 0.1551, while that for the linear model is 0.3469, showing clearly the superiority of the NFN over the linear model.

TABLE II
NEURO-FUZZY GPC CONTROL LAW PARAMETERS $(F_i = 0)$

Load	$S_i(z^{-1})$	d_{i1}
High	$\begin{pmatrix} 2.88-2.77z^{-1}+0.88z^{-2} \\ 5.54-7.09z^{-1}+2.55z^{-2} \\ 8.88-12.8z^{-1}+4.91z^{-2} \\ 12.8-19.7z^{-1}+7.86z^{-2} \\ 17.2-27.6z^{-1}+11.3z^{-2} \\ 22.1-36.4z^{-1}+15.3z^{-2} \\ 27.3-45.9z^{-1}+19.6z^{-2} \\ 32.9-56.1z^{-1}+24.2z^{-2} \\ 38.7-66.8z^{-1}+29.1z^{-2} \\ 44.7-78.0z^{-1}+34.3z^{-2} \end{pmatrix}$	$\begin{pmatrix} 0.027 \\ 0.071 \\ 0.125 \\ 0.183 \\ 0.244 \\ 0.303 \\ 0.362 \\ 0.419 \\ 0.475 \\ 0.531 \end{pmatrix}$
Medium high	$\begin{pmatrix} 2.89-2.79z^{-1}+0.90z^{-2} \\ 5.58-7.18z^{-1}+2.60z^{-2} \\ 8.98-13.0z^{-1}+5.01z^{-2} \\ 13.0-20.1z^{-1}+8.06z^{-2} \\ 17.6-28.2z^{-1}+11.7z^{-2} \\ 22.6-37.3z^{-1}+15.8z^{-2} \\ 28.0-47.2z^{-1}+20.3z^{-2} \\ 33.8-57.9z^{-1}+25.1z^{-2} \\ 39.9-69.2z^{-1}+30.3z^{-2} \\ 46.2-81.0z^{-1}+35.8z^{-2} \end{pmatrix}$	$\begin{pmatrix} 0.023 \\ 0.061 \\ 0.110 \\ 0.167 \\ 0.227 \\ 0.290 \\ 0.354 \\ 0.421 \\ 0.488 \\ 0.556 \end{pmatrix}$
Medium	$\begin{pmatrix} 2.90-2.81z^{-1}+0.91z^{-2} \\ 5.60-7.24z^{-1}+2.64z^{-2} \\ 9.00-13.1z^{-1}+5.10z^{-2} \\ 13.0-20.2z^{-1}+8.19z^{-2} \\ 17.5-28.4z^{-1}+11.8z^{-2} \\ 22.5-37.4z^{-1}+15.9z^{-2} \\ 27.7-47.2z^{-1}+20.4z^{-2} \\ 33.2-57.5z^{-1}+25.2z^{-2} \\ 38.9-68.1z^{-1}+30.2z^{-2} \\ 44.7-79.1z^{-1}+35.4z^{-2} \end{pmatrix}$	$\begin{pmatrix} 0.019 \\ 0.054 \\ 0.100 \\ 0.152 \\ 0.211 \\ 0.273 \\ 0.338 \\ 0.405 \\ 0.472 \\ 0.540 \end{pmatrix}$
Medium low	$\begin{pmatrix} 2.91-2.83z^{-1}+0.92z^{-2} \\ 5.66-7.32z^{-1}+2.66z^{-2} \\ 9.16-13.3z^{-1}+5.18z^{-2} \\ 13.3-20.7z^{-1}+8.38z^{-2} \\ 18.1-29.4z^{-1}+12.2z^{-2} \\ 22.5-39.1z^{-1}+16.6z^{-2} \\ 29.4-49.9z^{-1}+21.5z^{-2} \\ 35.6-61.5z^{-1}+26.9z^{-2} \\ 42.3-73.9z^{-1}+32.6z^{-2} \\ 49.8-87.7z^{-1}+38.9z^{-2} \end{pmatrix}$	$\begin{pmatrix} 0.017 \\ 0.046 \\ 0.087 \\ 0.135 \\ 0.191 \\ 0.252 \\ 0.318 \\ 0.388 \\ 0.462 \\ 0.540 \end{pmatrix}$
Low	$\begin{pmatrix} 2.92-2.84z^{-1}+0.92z^{-2} \\ 5.68-7.38z^{-1}+2.69z^{-2} \\ 9.22-13.5z^{-1}+5.24z^{-2} \\ 13.5-21.0z^{-1}+8.50z^{-2} \\ 18.4-29.8z^{-1}+12.4z^{-2} \\ 23.8-39.8z^{-1}+16.9z^{-2} \\ 29.9-50.8z^{-1}+22.0z^{-2} \\ 36.5-63.0z^{-1}+27.6z^{-2} \\ 43.4-76.0z^{-1}+33.6z^{-2} \\ 50.8-89.8z^{-1}+40.0z^{-2} \end{pmatrix}$	$\begin{pmatrix} 0.014 \\ 0.041 \\ 0.077 \\ 0.121 \\ 0.173 \\ 0.231 \\ 0.294 \\ 0.362 \\ 0.434 \\ 0.511 \end{pmatrix}$

B. Neuro-Fuzzy Predictive Control of Steam Temperature

From the trained NFN given in Table I, the NFGPC is determined and is shown in Table II. It is tested first to control the simulated power plant. In the simulation, the sampling interval is selected to be 30 s. For the NFGPC, the time delay d, which is also the minimum cost horizon, is set to 30 s, $Q = I$, and $\lambda = 0.1 \times I$. To investigate the effect of the control horizon M on the performance of NFGPC, several values of M are chosen with the prediction horizon N set to a relatively large

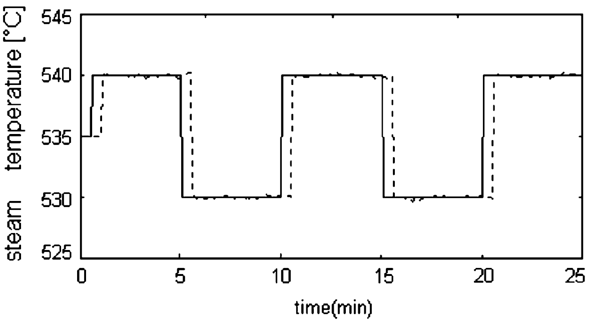


Fig. 7. Closed-loop response of superheated steam temperature using NFGPC (solid line—setpoint, dotted line—superheated steam temperature).

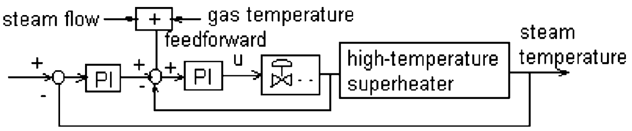


Fig. 8. Cascade control of superheated steam temperature.

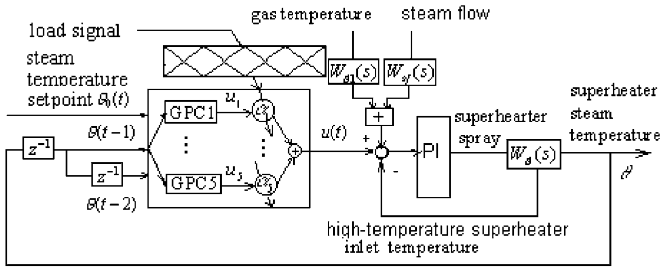


Fig. 9. Cascade NFGPC control of superheated steam temperature with feedforward control.

value of 10. For small M, the closed-loop response is sluggish. Reasonably good performance is obtained for $M = 6$. There is, however, little improvement when M is increased further. The control of the superheated steam temperature using the NFGPC for $M = 6$ is shown in Fig. 7. The performance of the NFGPC is clearly satisfactory. The same NFGPC is used later to control the actual power plant.

Cascade control scheme, as shown in Fig. 8, is widely used to control the superheated steam temperature. There are basically two nested loops. The inner one controls the superheated steam temperature by manipulating the spray water valve, and the outer one controls the superheated steam temperature by controlling the set-point of the inner loop. Feedfoward control, with the steam flow and the gas temperature as inputs, can be applied to provide a faster response to large variations in these two variables. In practice, the feedforward paths are activated only when there are significant changes in these variables. The control scheme also prevents the faster dynamics of the plant, i.e., the spray water valve and the water/steam mixing, from affecting the slower dynamics of the plant, i.e., the high temperature superheater [1]. The conventional control scheme shown in Fig. 8 involves two PI controllers for controlling the inner and outer loops. It is proposed here to replace the PI controller in the outer loop by the NFGPC, as shown in Fig. 9.

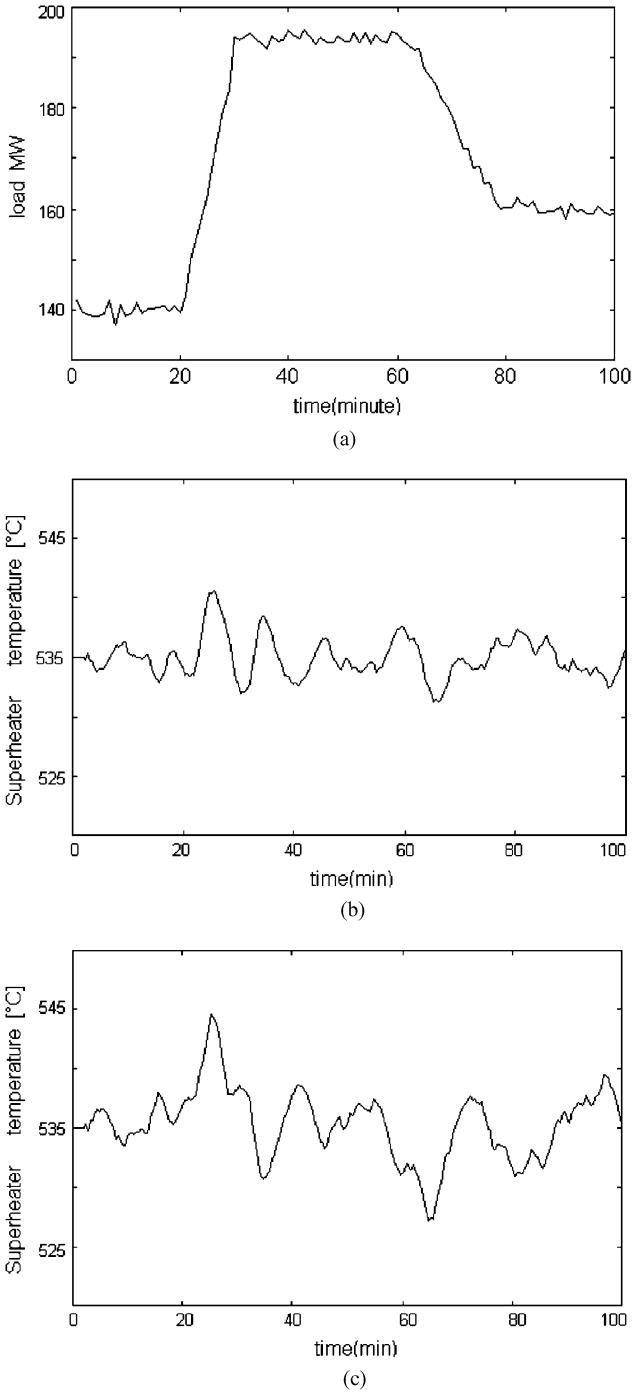


Fig. 10. (a) Load changes. (b) Superheated steam temperature controlled by the NFGPC. (c) Superheated steam temperature controlled by the cascade PI controller.

Denote the set-point and the measurement of the superheated steam temperature respectively by $\theta_0(t)$ and $\theta(t)$. As already discussed, the NFGPC consists of five local GPC controllers, each with $\theta_0(t), \theta(t-1), \theta(t-2)$ as its inputs. The local property of NFN implies only two local controllers are activated each time, determined by the load signal through five triangular membership functions. Consider the load changes shown in Fig. 10(a), where the load increases from 140 to 195 MW at

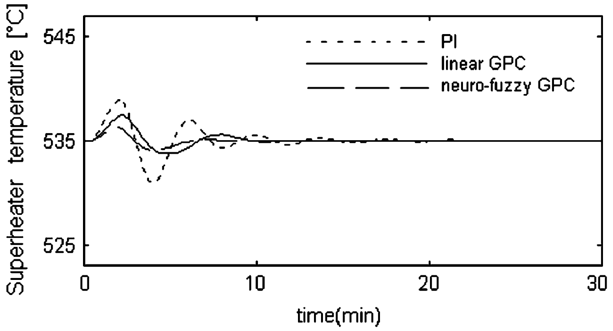


Fig. 11. Comparison of the NFGPC, conventional linear GPC, and cascade PI controller.

a rate of between 1%/min and 2.5%/min around 20 min, and decreases gradually to 160 MW from 60 min. When the load increases, more fuel is consumed by the furnace, leading to an increase in the gas temperature and thus an increase in the superheated steam temperature. To control the superheated steam temperature, the flow of the water spray to the superheater is adjusted by the control system shown in Fig. 9. From Fig. 10(b), good control of the superheated steam temperature is achieved by the NFGPC, as the temperature fluctuations for both the upward and downward load changes are within ±7 °C. This result is comparable to that presented in [6], which is tested on a 380 MW unit, under coal mill stop disturbance. In contrast, the fluctuations in the superheated steam temperature is much larger using the conventional cascade PI controller, as shown in Fig. 10(c).

As a further illustration, the power plant is simulated using the NFN model given in Table I, and is controlled, respectively, by the NFGPC, the conventional linear GPC controller, and the cascaded PI controller while the load changes from 160 MW to 200 MW. The conventional linear GPC controller is the local controller designed for the “medium” operating region. The results are shown in Fig. 11, showing that, as expected, the best performance is obtained from the NFGPC as it is designed based on a more accurate process model. This is followed by the conventional linear GPC controller, as the plant is nonlinear. The performance of the conventional cascade PI controller is the worse, indicating that it is unable to control satisfactory the superheated steam temperature under large load changes. This may be the reason for controlling the power plant manually when there are large load changes.

In practice, the control $u(t)$ of the GPC is usually computed by (12) and (20). However, if $u(t)$ exceeds the physical limits of the actuator, then actuator saturation occurs. In this case, the performance of the NFGPC will be affected. The constraint optimization by quadratic programming algorithm [19] can be incorporated to optimize the cost functions of the local GPCs subject to the actuator limits. Let the change in the rate of the spray water valve be limited to

$$-0.17 < \Delta u < 0.17. \qquad (22)$$

The performance of the NFGPC taking into account the rate limits of the actuator given by (22) for a load disturbance of 40 MW is shown in Fig. 12, where the control of spray water valve is normalized to [0, 1]. The control signal for the constrained NFGPC seems to be better in anticipating the effect of the actuator limits. When Δu exceeds the constraints, a new set of control signals is obtained, anticipating the control is going to exceed the limits. For a total comparison, Table III gives the summary of integral absolute errors (IAE) of these methods.

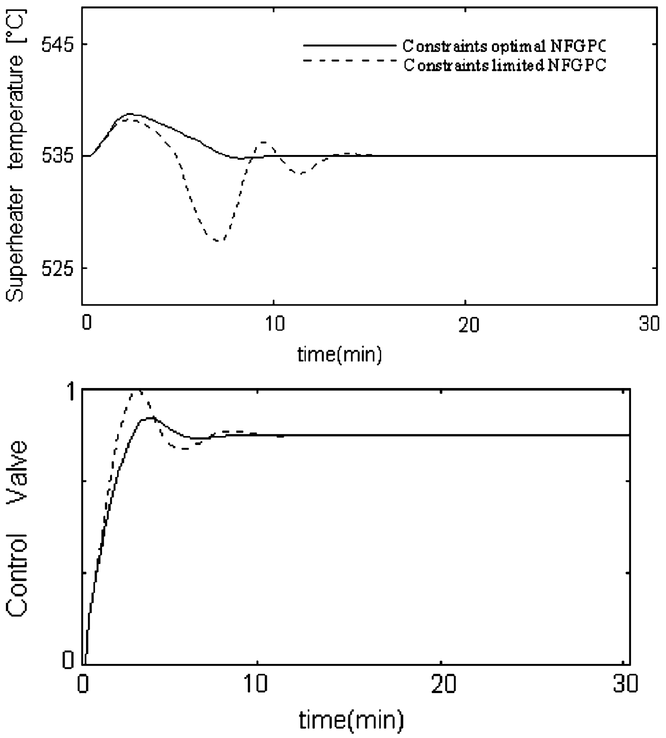


Fig. 12. Comparison of the constraints optimal NFGPC and the constraints limited NFGPC performance.

TABLE III
COMPARISON OF IAE

PI	Linear GPC	NFGPC	Constraints optimal NFGPC	Constraints limited NFGPC
17.02	7.974	3.921	9.303	19.01

Since GPC can be extended to minimize the cost function J taking into account of inequality constraint by solving of quadratic programming, it can accommodate the actuator constraints, which are neglected in the majority of other control algorithms. Although better simulation results are obtained, the computation load is still quite heavy. Further works will be conducted to devise methods to reduce the computation load when the NFGPC is subject to actuator constraints. The modified NFGPC will then be applied to control the power plant. It should be noted that constraint GPC is an active area of research that is receiving immense interest [20]–[22]. However, the computing efficiency, the stability and the feasibility question are still the widely discussed problems. In existing approaches, the actuator saturation compensators are often designed for a specific plant and controller to ensure the compensated closed-loop system to be stable.

VI. Conclusion

The modeling and control of a 200-MW power plant using the neuro-fuzzy approach is presented in this paper. The NFN consists of five local CARIMA models, and the output of the network is the interpolation of the local models using memberships given by the B-spline basis functions. The proposed NFGPC is similarly constructed, which consists of five GPCs, each of which is designed from the CARIMA models in the NFN. The NFGPC is most suitable for processes with smooth nonlinearity, such that its full operating range can be partitioned into several local linear operating regions. The proposed approach has been successfully applied to model the 200-MW power plant, from which the NFGPC is designed. The NFGPC is applied first to the simulated power plant before applying it to the actual power plant. Much better performance is obtained from NFGPC than from the traditional cascade PI controller in controlling the actual power plant under large load changes. Comparison with the cascade PI controller and the linear GPC is also made on the simulated power plant. As expected, NFGPC outperforms the other two control schemes. The proposed NFGPC therefore provides a useful alternative for controlling this class of nonlinear power plants, which are formerly difficult to be controlled using traditional methods.

References

[1] R. N. Silva, P. O. Shirley, J. M. Lemos, and A. C. Goncalves, "Adaptive regulation of super-heated steam temperature: A case study in an industrial boiler," *Contr. Eng. Pract.*, vol. 8, no. 8, pp. 1405–1415, 2000.

[2] K. J. Astrom and K. Ecklund, "A simplified nonlinear model of a drum boiler-turbine unit," *Int. J. Contr.*, vol. 16, no. 1, pp. 145–169, 1972.

[3] G. Pellegrinetti and J. Bentsman, "Nonlinear control oriented boiler modeling—A benchmark problem for controller design," *IEEE Trans. Contr. Syst. Technol.*, vol. 4, no. 1, pp. 57–64, Jan. 1996.

[4] P. K. Chawdhry and B. W. Hogg, "Identification of boiler models," *Proc. Inst. Electr. Eng. Contr. Theory Appl.*, vol. 136, no. 5, pp. 261–271, Sep. 1989.

[5] G. Irwin, D. Brown, B. W. Hogg, and E. Swidenbank, "Neural network modelling of a 200-MW boiler system," *Proc. Inst. Electr. Eng., Control Theory Appl.*, vol. 142, no. 6, pp. 529–536, Nov. 1995.

[6] T. Moelbak, "Advanced control of superheater steam temperatures—An evaluation based on practical applications," *Contr. Eng. Pract.*, vol. 7, no. 7, pp. 1–10, 1999.

[7] G. Prasad, E. Swidenbank, and B. W. Hogg, "A neural net model-based multivariable long-range predictive control strategy applied thermal power plant control," *IEEE Trans. Energy Conv.*, vol. 13, no. 2, pp. 176–182, Jun. 1998.

[8] S. Goto, M. Nakamura, and S. Matsumura, "Automatic realization of human experience for controlling variable pressure boilers," *Contr. Eng. Pract.*, vol. 10, no. 10, pp. 15–22, 2000.

[9] U. C. Moon and K. Y. Lee, "A boiler-turbine system control using a fuzzy auto-regressive moving average (FARMA) model," *IEEE Trans. Energy Conv.*, vol. 18, no. 1, pp. 142–148, Mar. 2003.

[10] M. Brown and C. J. Harris, *Neurofuzzy Adaptive Modelling and Control*. Englewood Cliffs, NJ: Prentice-Hall, 1994.

[11] C. W. Chan, K. C. Cheung, and W. K. Yeung, "A computation efficient on-line training algorithm for neurofuzzy networks," *Int. J. Syst. Sci.*, vol. 31, no. 3, pp. 297–306, 2000.

[12] C. W. Chan, X. J. Liu, and W. K. Yeung, "Neurofuzzy network based self-tuning control with offset elimination," *Int. J. Syst. Sci.*, vol. 34, no. 2, pp. 111–122, 2003.

[13] X. J. Liu, F. Lara-Rosano, and C. W. Chan, "Model-reference adaptive control based on neurofuzzy networks," *IEEE Trans. Syst., Man, Cybern. C*, vol. 34, no. 3, pp. 302–309, Aug. 2004.

[14] J. Zhang and A. J. Morris, "Long range predictive control of nonlinear processes based on recurrent neuro-fuzzy network models," *Neural Comput. Appl.*, pp. 50–59, 2000.

[15] D. W. Clarke, C. Mohtadi, and P. S. Tuffs, "Generalized predictive control," *Automatica*, vol. 23, no. 2, pp. 137–160, 1987.

[16] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modelling and control," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-15, no. 1, pp. 116–132, Jan./Feb. 1985.

[17] Y. L. Huang, H. H. Lou, J. P. Gong, and T. F. Edgar, "Fuzzy model predictive control," *IEEE Trans. Fuzzy Syst.*, vol. 8, no. 6, pp. 665–678, Dec. 2000.

[18] X. J. Liu and X. X. Zhou, "Identification of boiler models and its fuzzy logic strategy," in *Proc. 14th IFAC World Congr.*, Beijing, China, Jun. 1999, pp. 149–154.

[19] C. L. Lawson and R. J. Hanson, *Solving Least Squares Problems*. Englewood Cliffs, NJ: Prentice-Hall, 1974.

[20] T. T. C. Tsang and D. W. Clarke, "Generalised predictive control with input constraints," *Proc. Inst. Electr. Eng., Control Theory Appl.*, vol. 135, no. 6, pp. 451–460, Nov. 1988.

[21] J. A. Rossiter and B. Kouvaritakis, "Constrained stable generalised predictive control," *Proc. Inst. Electr. Eng., Contr. Theory Appl.*, vol. 140, no. 4, pp. 243–254, Jul. 1993.

[22] P. O. M. Scokaert and D. W. Clarke, "Stabilising properties of constrained predictive control," *IEE Proc.-Contr. Theory Appl.*, vol. 141, no. 5, pp. 295–304, Sep. 1994.

Xiang-Jie Liu received the Ph.D. degree in electrical and electronic engineering from the Research Center of Automation, Northeastern University, Shenyang, China, in 1997.

He subsequently held a postdoctoral position with the China Electric Power Research Institute (CEPRI), Beijing, China, until 1999. He has been an Associate Professor in CEPRI since 1999. He was a Research Associate with the University of Hong Kong. He joined the National University of Mexico in 2001, where he was a Professor with the Research Center of Applied Science and Technology Development. He is now a Professor with the Department of Automation, North China Electric Power University, Beijing, China. His current research areas include fuzzy control, neural networks, filtering, intelligent control theory and its application in industrial process, development and application of DCS.

Dr. Liu was a candidate for the "Application Price" at the 14th IFAC World Congress.

C. W. Chan received the Higher Diploma in mechanical engineering from Hong Kong Polytechnic, Hong Kong, and the M.Sc. and Ph.D. degrees from the Control Systems Centre, University of Manchester Institute of Science and Technology, Manchester, U.K.

He was with Weir Pumps, PLC, National Engineering Laboratory, and the Unilever Research Port Sunlight Laboratory before joining the University of Hong Kong, Hong Kong. His current research interests include system identification, compensation for nonlinear actuators, fuzzy logic control, and neural networks based modeling and control. He is Regional Editor (Asia Pacific) of the *International Journal of Systems Science*.

Dr. Chan is a member of the Hong Kong Institution of Engineers, the Institution of Mechanical Engineers, the Institution of Electrical Engineers, and an Associate Fellow of the Institute of Mathematics and its Applications.