Proceeding of the 2004 American Control Conference
Boston, Massachusetts June 30 - July 2, 2004

Adaptive Predictive Control with Neural Prediction for a Class of Nonlinear Systems with Time-Delay

Chi-Huang Lu **Ching-Chih Tsai***
Department of Electrical Engineering National Chung-Hsing University
250, Kuo-Kuang Road, Taichung, Taiwan. *Email:cctsai@dragon.nchu.edu.tw

***Abstract*—**This paper presents an adaptive predictive control with neural prediction for a class of single-input single- output nonlinear systems with known time-delay. The well-known linear dynamic modeling approach together with the neural modeling method is employed to approximate these nonlinear systems. The predictive control law with integral action is derived based on the minimization of a generalized predictive performance criterion. A real-time adaptive control algorithm, including a recursive least-squares estimator and a proposed neural predictor, is successfully applied to achieve the control performance specifications. Stability and properties of the closed-loop control systems are investigated as well. Simulation results reveal that the proposed control gives satisfactory tracking and disturbance rejection performance for two illustrative time-delay nonlinear systems. Experimental results for a variable-frequency oil-cooling control process are performed which have shown effectiveness of the proposed method under the conditions of set-points and load changes.

***Index Terms*:** General predictive control, neural networks, nonlinear system, variable-frequency oil-cooling machine.

I. INTRODUCTION

GENERALIZED predictive control (GPC) has been extensively used for industrial applications, and the theories and design techniques using GPC have been well documented in [1-7]. In many industrial processes there usually exhibit time-delay and nonlinear dynamical phenomena, and such complicated systems may not be easily controlled by the use of linear GPC method. Recently, neural networks have been widely used as modeling tools as well as controllers for a class of nonlinear systems [8-13]. Khalid *et al.* [8] developed a feedforward multi-layer neural controller for an MIMO furnace and compared its performance with other advanced controllers. Piché *et al.* [10] established a neural-network-based technique for constructing nonlinear dynamic models from their empirical input-output data. Shi *et al.* [11] applied neural networks to build direct self-tuning controllers for induction motors. Zhu. *et al.* [12] developed a robust nonlinear predictive control with neural network compensator. Song *et al*. [13] explored a nonlinear predictive control with its application to manipulator with flexible forearm, and their nonlinear predictive controller was designed on the basis of a neural network plant model using the receding-horizon control approach. Furthermore, Tan *et al*. [14] presented neural-network-based d-step-ahead predictors for a class of nonlinear systems with time-delay.

Aside from neural modeling and control, the authors in [15] used linear dynamic modeling approach together with the neural modeling method to approximate a class of nonlinear systems and then developed an adaptive robust model predictive controller to achieve their control goals. However, the method proposed by Qin *et al.* [15] is limited to the systems with only one-step time-delay and does not have an integral action to eliminate steady-state regulation or tracking errors caused by modeling errors or constant external disturbances. To overcome such shortcomings, this paper will develop a novel adaptive predictive control with neural prediction for a class of SISO nonlinear systems with time-delay, in which the neural-network-based d-step-ahead predictors are realized by using the well-known backpropagation neural network architecture. The feasibility and effectiveness of the proposed method will be verified through its applications to two nontrivial nonlinear plants and one physical variable-frequency oil-cooling process.

The remaining parts of the paper are outlined as follows. Section II presents approximately mathematical models of a class of nonlinear systems with time-delay. The predictive control law with integral action is derived in Section III. The backpropagation neural predictor and the modified least-squares estimator are developed to estimate the unknown parameters for the system model in Section IV. A real-time adaptive neural predictive control algorithm is proposed in Section V. Section VI details the capabilities of the proposed algorithm utilizing computer simulations. Two experimental results for controlling the oil cooling process to meet the desired performance specifications are presented in Section VII. Section VIII concludes this paper.

II. APPROXIMATING NONLINEAR MODELS

The section is devoted to approximating a class of discrete-time, single-input single-output nonlinear plants discussed in [12,14]. These systems are assumed to have plant inputs as $u(\cdot): Z^+ \rightarrow \Re$, plant outputs as $y(\cdot): Z^+ \rightarrow \Re$, and the nonlinear mappings $f(\cdot): \Re^{n_y+n_u-d+1} \rightarrow \Re$, and $n_y \in Z^+$, $n_u \in Z^+$. Furthermore, $d \in Z^+$ represents the time delay of the systems. Generally speaking, such systems can be described by the following nonlinear autoregressive moving averaging (NARMA) models with given time-delay

$$y(k)=f(y(k-1),\ y(k-2),\ \cdots,\ y(k-n_y),\ u(k-d),\ u(k-d-1),\ \cdots,\ u(k-n_u)). \quad (1)$$

To design the proposed predictive controller, the well-known linear dynamic modeling approach together with the neural modeling method is employed to approximate these nonlinear systems. Thus, we have

$$y(k)=a(z^{-1})y(k-1)+b(z^{-1})u(k-d)+n(k-1) \quad (2)$$

where

$$a(z^{-1})=a_1+a_2z^{-1}+a_3z^{-2}+\cdots+a_{n_a}z^{-(n_a-1)}$$

$$b(z^{-1})=b_0+b_1z^{-1}+b_2z^{-2}+\cdots+b_{n_b}z^{-n_b}$$

$$n(k-1)=\varphi(y(k-1),\ \cdots,\ y(k-n_y),\ u(k-d),\ \cdots,\ u(k-n_u)).$$

$a(z^{-1})$ and $b(z^{-1})$ are two polynomials in the back shifting operator

z^{-1} with $n_a \le n_y$ and $n_b \le (n_u - d)$, and $\varphi(\cdot): \Re^{n_y+n_u-d+1} \to \Re$. Note that the two polynomials $a(z^{-1})$ and $b(z^{-1})$, and the nonlinear term $n(k-1)$ can be identified using the recursive least-squares estimator combined with the proposed neural predictor, respectively.

III. PREDICTIVE CONTROL LAW WITH INTEGRAL ACTION

This section is devoted to developing a novel predictive control for improving tracking performance and disturbance rejection abilities of this type of nonlinear control system. With the shifting operator z^{-1} and the identified system parameters, the system model (2) can be rewritten by

$$\tilde{a}(z^{-1})y(k) = b(z^{-1})z^{-d}u(k) + z^{-1}n(k) \tag{3}$$

where $\tilde{a}(z^{-1}) = 1 - a_1 z^{-1} - a_2 z^{-2} - \cdots - a_{n_a} z^{-n_a}$.

The predictive control law is derived so as to minimize the following cost function

$$J = \sum_{j=d}^{N_2} \left(y(k+j) - r(k+j) + h_j(z^{-1})\Delta n(k+j-1) \right)^2 + \sum_{j=d}^{d+N_u-1} \left(q_j(z^{-1})\Delta u(k+j-d) \right)^2 \tag{4}$$

where $r(k)$ is an input reference signal, $\Delta = 1 - z^{-1}$, and the main function of $h_j(z^{-1})$ is used to remove the effect of $n(k-1)$ on the closed-loop control system. $q_j(z^{-1})$ is a selected weighting polynomial and $q_j(z^{-1}) = q_{j,0} + q_{j,1}z^{-1} + \cdots + q_{j,d}z^{-d}$. N_2 and N_u denote the maximum output horizon and the control horizon, respectively.

To derive the predictive control law, the following two equalities are used to solve for $e_j(z^{-1})$, $f_j(z^{-1})$, and $g_j(z^{-1})$, in order to find a j step-ahead predictor of $y(k)$

$$1 = \Delta e_j(z^{-1})\tilde{a}(z^{-1}) + z^{-j} f_j(z^{-1}) \tag{5}$$

$$g_j(z^{-1}) = e_j(z^{-1}) b(z^{-1}) \tag{6}$$

where $j = 1, 2, \cdots$, and

$$e_j(z^{-1}) = 1 + e_{j,1}z^{-1} + e_{j,2}z^{-2} + \cdots + e_{j,j-1}z^{-(j-1)}$$

$$f_j(z^{-1}) = f_{j,0} + f_{j,1}z^{-1} + f_{j,2}z^{-2} + \cdots + f_{j,n_a}z^{-n_a}$$

$$g_j(z^{-1}) = g_{j,0} + g_{j,1}z^{-1} + g_{j,2}z^{-2} + \cdots + g_{j,j+n_b-1}z^{-(j+n_b-1)}.$$

Thus, the j step-ahead output prediction of $y(k)$ is calculated by

$$y(k+j) = f_j(z^{-1})y(k) + g_j(z^{-1})\Delta u(k+j-d). \tag{7}$$

Using (7), the cost function J in (4) can be equivalently expressed in the subsequent quadratic form

$$J = (Fy(k) + \overline{G}U + L + MN - R)^T (Fy(k) + \overline{G}U + L + MN - R) + U^T Q^T Q U \tag{8}$$

where

$$F = \left[f_d(z^{-1}) \quad f_{d+1}(z^{-1}) \quad \cdots \quad f_{N_2}(z^{-1}) \right]^T$$

$$U = \left[\Delta u(k) \quad \Delta u(k+1) \quad \cdots \quad \Delta u(k+N_u-1) \right]^T$$

$$M = diag\left[e_d(z^{-1}) + h_d(z^{-1}) \quad e_{d+1}(z^{-1}) + h_{d+1}(z^{-1}) \quad \cdots \quad e_{N_2}(z^{-1}) + h_{N_2}(z^{-1}) \right]$$

$$N = \left[n(k+d-1) \quad n(k+d) \quad \cdots \quad n(k+N_2-1) \right]^T$$

$$\overline{G} = \begin{bmatrix} g_{d,0} & 0 & \cdots & 0 \\ g_{d+1,1} & g_{d+1,0} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_{N_2,N_u-1} & g_{N_2,N_u-2} & \cdots & g_{N_2,0} \end{bmatrix}$$

$$L = \begin{bmatrix} \sum_{i=1}^{d} g_d u(k-i) \\ \sum_{i=2}^{d+1} g_{d+1} u(k-i+1) \\ \vdots \\ \sum_{i=N_u}^{N_2} g_{N_2} u(k-i+N_u-1) \end{bmatrix}$$

$$R = \left[r(k+d) \quad r(k+d+1) \quad \cdots \quad r(k+N_2) \right]^T$$

$$Q = \left[q_d(z^{-1}) \quad q_{d+1}(z^{-1}) \quad \cdots \quad q_{d+N_u-1}(z^{-1}) \right].$$

Because the cost function J is quadratic in U, a minimum solution for U is easily obtained from

$$\left. \frac{\partial J}{\partial U} \right|_{U=U^*} = 0. \tag{9}$$

which, by letting N_u =1, leads to the following present increment control output

$$\Delta u(k) = (G^T G + q_0)^{-1} G^T \left(R - Fy(k) \right) - m(z^{-1})\Delta \hat{n}(k+d-1) \tag{10}$$

where $G = \left[g_d \quad g_{d+1} \quad \cdots \quad g_{N_2} \right]^T \equiv \left[g_{d,0} \quad g_{d+1,0} \quad \cdots \quad g_{N_2,0} \right]^T$, $q_0 > 0$, and $m(z^{-1})$ is a polynomial and can be obtained from $b(z^{-1})m(z^{-1}) = 1$.

IV. PARAMETER ESTIMATION AND NEURAL PREDICTION

A. System Parameters Estimation

This subsection modifies the recursive least-squares estimation (RLSE) method to identify unknown system parameters. The system model (2) can be rewritten as

$$y(k) = \zeta(k)^T \theta(k) + n(k-1) \tag{11}$$

where

$$\zeta(k) = \left[y(k-1) \quad \cdots \quad y(k-n_a) \quad u(k-d) \quad \cdots \quad u(k-n_b) \right]^T$$

$$\theta(k) = \left[a_1(k) \quad \cdots \quad a_{n_a}(k) \quad b_0(k) \quad \cdots \quad b_{n_b}(k) \right]^T.$$

Define the error covariance matrix $P(k) = \left(\Theta^T(k)\Theta(k) \right)^{-1}$ where $\Theta(k) = \sum_k \varsigma^T(k)\varsigma(k)$, and assume that the matrix $\Theta(k)$ has full rank for $k \ge 0$. Given $\hat{\theta}(0)$ and $P(0)$, the least-squares estimate $\hat{\theta}(k)$ then satisfies the following recursive equations

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \frac{P(k-1)\zeta^T(k)\left(y(k) - \zeta(k)\hat{\theta}(k-1) - \hat{n}(k-1) \right)}{1 + \zeta(k)P(k-1)\zeta^T(k)} \tag{12}$$

$$P(k) = P(k-1) - \frac{P(k-1)\zeta^T(k)\zeta(k)P(k-1)}{1 + \zeta(k)P(k-1)\zeta^T(k)} \tag{13}$$

where $P(k)$ denotes a $(n_a + n_b + 1) \times (n_a + n_b + 1)$ symmetric matrix and $P(0) = \delta I$, δ is a positive number and I is $(n_a + n_b + 1)$ -order unity matrix, and $\hat{n}(k-1)$ is the output of the neural predictor. Note that these recursive least-squares estimation equations can be easily derived using the least-squares loss function $V(k) = \frac{1}{2}\left(y(k) - \zeta^T(k)\hat{\theta}(k) - \hat{n}(k-1) \right)^2$ in [16].

Furthermore, for the system model (11), if the modified RLSE equations (12) and (13) are used to identify the model parameter vector $\theta(k)$ utilizing appropriate inputs with desired persistent excitation conditions, then the estimated parameter $\hat{\theta}(k)$ is bounded if $\hat{n}(k-1)$ is bounded.

B. Neural Predictor

This subsection aims to develop a neural predictor using a multiplayer feedforward neural network architecture described in [17]. The neural predictor can be trained to learn $\hat{n}(k-1)$ by using the input vector

$$x=\left[y(k-1) \ \cdots \ y(k-n_y) \ \ u(k-d-1) \ \cdots \ u(k-n_u) \ \ 1\right]^T. \quad (14)$$

This type of neural predictor has a two-layer perceptron network with n_i inputs, n_j hidden units, and only one output variable. The predictor's output is denoted by $\hat{n}(k-1)$, and W_j stands for the weighting between the hidden and output layers. Mathematically, we have

$$\hat{n}(k-1)=\sum_{j=1}^{n_j} W_j \Gamma\left(\sum_{i=1}^{n_i} w_{ij} x_i\right) \quad (15)$$

where $\Gamma(\chi)=1/\left(1+e^{-\chi}\right)$, $n_i=n_y+n_u-d+1$, and x_i represents the *i*th entry of the input vector for the neural predictor. Let w_{ij} s be the weightings between the input and hidden layers. To update the weightings w_{ij} of the neural predictor, we define the following performance criterion ψ

$$\psi=\frac{1}{2}\left(n(k-1)-\hat{n}(k-1)\right)^2 \quad (16)$$

where $n(k-1)=y(k)-\hat{a}(z^{-1})y(k-1)-\hat{b}(z^{-1})u(k-d)$.

Therefore, the weights can be recursively adjusted in order to reduce the cost function ψ to its minimum value by the gradient descent method, and the weights are updated by

$$\boldsymbol{W}(k)=\boldsymbol{W}(k-1)-\eta\frac{\partial\psi}{\partial \boldsymbol{W}(k-1)} \quad (17)$$

where η is so-called learning rate, and $\partial\psi/\partial\boldsymbol{W}(k-1)$ can be calculated as follows;

$$\frac{\partial\Psi}{\partial W_j(k-1)}=-\left(n(k-1)-\hat{n}(k-1)\right)\Gamma(\beta),\ \beta=\sum_{i=1}^{n_i}\left(w_{ij}(k-1)x_i\right) \quad (18)$$

$$\frac{\partial\Psi}{\partial w_{ij}(k-1)}=-\left(\left(n(k-1)-\hat{n}(k-1)\right)W_j(k)\frac{e^{-\beta}}{\left(1+e^{-\beta}\right)^2}x_i\right). \quad (19)$$

Assume that the weight values $w_{ij}(k)$, $W_j(k)$ for the neural predictor (15) are updated using (17), (18) and (19), then the predictor (15) will locally converge at an exponential rate, provided that the learning rate η satisfies the following condition:

$$0<\eta<\frac{2}{\left\|\frac{\partial\hat{n}(k-1)}{\partial\boldsymbol{W}(k-1)}\right\|^2}. \quad (20)$$

Notice that above statement can be shown by choosing a Lyapunov function candidate $V(k)=\frac{1}{2}\left(n(k-1)-\hat{n}(k-1)\right)^2$ and the proof procedure can be referred to [18].

Using the neural predictor, the j-step-ahead prediction of $\hat{n}(k-1)$ can be recursively obtained from employing the following input vector

$$x=\left[\hat{y}(k+j-1) \ \cdots \ \hat{y}(k+j-n_y) \ \ u(k+j-d-1) \ \cdots \ u(k+j-n_u) \ \ 1\right]^T \quad (21)$$

where $\hat{y}(k+m)=\hat{a}(z^{-1})\hat{y}(k+m-1)+\hat{b}(z^{-1})u(k+m-d)+\hat{n}(k+m-1)$ and $m<d$. Consequently, the term $\Delta\hat{n}(k+d-1)$ in the control increment $\Delta u(k)$ can be definitely computed utilizing the neural predictor.

V. REAL-TIME ADAPTIVE CONTROL ALGORITHM

To make the controller exhibit adaptive characteristics, we include the recursive least-square estimation method in the control loop, and propose the sequel real-time adaptive neural control algorithm.

Step 1) Select set-points $r(k)$.

Step 2) Set d, N_2, q_0, n_a, n_b, n_y, n_u, n_j, η, and off-line learn $\hat{n}(k-1)$.

Step 3) Measure the current process output $y(k)$.

Step 4) Estimate the system parameters $\hat{a}(z^{-1})$, $\hat{b}(z^{-1})$ using (12) and (13).

Step 5) Learn $\hat{n}(k-1)$ by the neural network.

Step 6) Compute the increment control output $\Delta u(k)$ based on (10).

Step 7) Output the control signal $u(k)$ to the control process.

Step 8) Repeat steps 3-7.

The stability of the adaptive predictive control method relies upon convergence of the estimates $\hat{a}_i$, $\hat{b}_i$, $\hat{n}(k-1)$ of system (2) by the RLSE method and the neural predictor, respectively. The following theorem will state that the aforementioned algorithm has zero steady-state tracking errors and is asymptotically stable.

Theorem 1: Assume that the upper bounds for n_a, n_b, n_y, n_u be known, and all set-points $r(k)$ are set to be constant, i.e. $r(k)=r$. Then the closed-loop system has the following property:

$$\lim_{k\to\infty}\{y(k)-r\}=0. \quad (22)$$

Proof: We substitute the control law (10) into (2) and obtain the resultant closed-loop system

$$\left(\Delta\hat{a}(z^{-1})+\hat{b}(z^{-1})z^{-d}(G^TG+q_0)^{-1}G^TF\right)y(k)$$
$$=\hat{b}(z^{-1})z^{-d}(G^TG+q_0)^{-1}G^TR+\left(1-\hat{b}(z^{-1})m(z^{-1})\right)\Delta n(k-1). \quad (23)$$

With the properties $\Delta\big|_{z=1}=0$, $F\big|_{z=1}=[\underbrace{1 \ 1 \ \cdots \ 1}_{N_2-N_1+1}]^T$, and $R\big|_{z=1}=r(k)[\underbrace{1 \ 1 \ \cdots \ 1}_{N_2-N_1+1}]^T$, it follows that

$$\lim_{k\to\infty}\{y(k)-r\}=\left\{\left(\Delta\hat{a}(z^{-1})+\hat{b}(z^{-1})z^{-d}(G^TG+q_0)^{-1}G^TF\right)^{-1}\right.$$
$$\left.\left(\hat{b}(z^{-1})z^{-d}(G^TG+q_0)^{-1}G^TR+(1-\hat{b}(z^{-1})m(z^{-1}))\Delta n(k-1)\right)-r\right\}\Big|_{z=1}$$
$$=\left\{\left(\hat{b}(z^{-1})z^{-d}(G^TG+q_0)^{-1}G^TF\right)^{-1}\left(\hat{b}(z^{-1})z^{-d}(G^TG+q_0)^{-1}G^TR\right)-r\right\}\Big|_{z=1}$$
$$=0$$

Theorem 2: Let the error terminal constrain be zero, as can be referred to [6] and Theorem 1, and the upper bounds for n_a, n_b, n_y, n_u be known. Then the closed-loop system is asymptotically stable.

Proof: By defining a Lyapunov function candidate

$$V(k) = \sum_{j=d}^{N_2} \left(\hat{y}(k+j) - r(k+j)\right)^2 . \quad (24)$$

we have

$$\begin{aligned}\Delta V(k+1) &\equiv V(k+1) - V(k) \\ &= \left(\hat{y}(k+N_2+1) - r(k+N_2+1)\right)^2 - \left(\hat{y}(k+d) - r(k+d)\right)^2 \\ &= -\left(\hat{y}(k+d) - r(k+d)\right)^2 \\ &= -\left(f_d(z^{-1})y(k) + g_d \Delta u(k) - r(k+d)\right)^2 \\ &< 0\end{aligned}$$

which establishes Lyapunov stability of the closed-loop system..

VI. COMPUTER SIMULATIONS

The objective of this simulation is to study the feasibility of the adaptive predictive control with neural prediction for the underlying two illustrative nonlinear systems. The simulation study also includes an investigation of the effect of load disturbances on the control system employing the proposed controller.

A. Example 1

The following nonlinear system is modified from a model in [9]. The simulation was performed for two sets of reference inputs $r(k)$, the time-delay d, and the parameters given by

$$y(k) = 0.9831y(k-1) - 0.0853u(k-6) - 0.0088y(k-1)u(k-6) + 0.0036y(k-2)u^2(k-7)$$

$$r(k) = \begin{cases} 1, & 0 < k \le 400 \\ -1, & 400 < k \le 800 \end{cases},$$

$$d = 6, \quad N_2 = 50, \quad q_0 = 4500, \quad n_j = 10, \quad \eta = 0.1,$$

$$n_a = 1, \quad n_b = 0, \text{ i.e., } \tilde{a}(z^{-1}) = 1 - a_1 z^{-1}, \quad b(z^{-1}) = b_0 .$$

Fig. 1 shows the response of the adaptive neural predictive control under set-point changes. We observe in Fig. 1 that the proposed controller is capable of giving an excellent set-point tracking performance.

In order to investigate disturbance rejection ability of load disturbances on the performance of the proposed controller, the mathematical model was perturbed to

$$y(k) = 0.9831y(k-1) - 0.0853u(k-6) - 0.0088y(k-1)u(k-6) + 0.0036y(k-2)u^2(k-7) + v(k)$$

where

$$v(k) = \begin{cases} 0.2 & \text{at } k = 200 \\ -0.2 & \text{at } k = 600 \\ 0.0 & \text{otherwise.} \end{cases}$$

Fig. 2 depicts the simulation result for the proposed controller with load disturbances. Consequently, the adaptive predictive control with neural networks demonstrated a good disturbance rejection capability.

B. Example 2

The following modified nonlinear system is taken from [19]. The simulation was conducted for two sets of reference inputs $r(k)$, and the time delay d, and the following parameters

$$y(k) = \frac{y(k-1)}{1 + y^2(k-1)} + 5u(k-6) + u(k-7), \quad r(k) = \begin{cases} 1, & 0 < k \le 200 \\ -1, & 200 < k \le 400 \\ 1, & 400 < k \le 600 \\ -1, & 600 < k \le 800 \end{cases}$$

$$d = 6, \quad N_2 = 50, \quad q_0 = 75000, \quad n_a = 1, \quad n_b = 0, \quad n_j = 10, \quad \eta = 0.1.$$

Fig. 3 depicts the response of the adaptive neural predictive controller under set-point changes. We see in Fig. 3 that the proposed controller is capable of giving an excellent set-point tracking performance.

In order to explore the effect of load disturbances on the performance of the proposed controller, the model was added with an external disturbance, i.e., $y(k) = y(k-1)/(1 + y^2(k-1)) + 5u(k-6) + u(k-7) + v(k)$ where $v(k) = -0.2$ at $300 \le k < 500$ and $v(k) = 0.1$ at $k \ge 500$. Fig. 4 shows the simulation result for the proposed controller with the load disturbances. The result reveals that the proposed controller exhibits a good disturbance rejection capability.

In comparison with the adaptive predictive controller without neural predictor, Figs. 5 and 6 show the responses of the proposed controller without neural predictor and $q_0 = 15000$ for two cases: no disturbance $v(k) = 0$ and perturbed $v(k)$, respectively. As can be seen in Fig. 6, the adaptive predictive without neural predictor exhibits a poor performance for the system with the disturbance.

VII. APPLICATION TO A VARIABLE- FREQUENCY OIL-COOLING PROCESS

A. Brief Description of a Variable-Frequency Oil-Cooling Machine

The temperature control system for the variable-frequency oil-cooling machine has been extensively used in the manufacturing industry. The oil cooling process developed in [20-21] is composed of a compressor driven by a variable-frequency induction motor, a condenser, an expander, an evaporator, a pumps and a heat exchanger. The dynamics of such a system is highly nonlinear and very difficult to obtain its exact mathematical model. The controller consists of two platinum temperature sensing modules with an accuracy of $\pm 0.1^o C$ of the resistance-to-voltage (R-V) transducers and the AD/DA board inserted in the slot of the personal computer (PC). There are 32-channel 12-bit analog-to-digital converters and 2-channel 12-bit digital-to-analog converters ranging from 0 to 5V. The 586 compatible PC is responsible for editing, compiling, and running the C program codes for the controller. The sampling period in this temperature control system is 3 seconds. Fig. 7 shows a schematic diagram of the PC-based temperature control system.

B. Experimental Results and Discussion

The aims of the temperature control system are to reach the set-points during startup as rapidly as possible while avoiding large overshoot, and to track the set-points with least errors in the presence of set-point changes and load disturbances. Hence, the controller is required to meet the following performance specifications: the overshoot must be less than $0.5\ ^oC$ and the steady-state error in the temperature zone must remain within $\pm 0.2\ ^oC$ for any arbitrary constant step commands and disturbances. The following experiments were performed to observe whether these goals were achieved. In all these experiments, the real-time adaptive control algorithm presented in Section V was implemented using the PC-based controller.

The first experiment was adopted to test the set-point tracking capability of the proposed method. The appropriate values of the maximum output horizon N_2 and the weighing value q_0 were carefully chosen so as to enable the overall system to meet the required performance specifications. This controlled experiment was conducted with the following parameter settings

$$r(k) = \begin{cases} 20^o C, & 0 < k \le 600 \\ 18^o C, & 600 < k \le 1200 \end{cases}$$

$d = 12$, $N_2 = 100$, $q_0 = 3500$, $n_a = 1$, $n_b = 0$,

$n_y = 2$, $n_u = 13$, $n_j = 10$, $\eta = 0.1$.

Consequently, the control signal $u(k)$ for the process is given by

$$u(k) = u(k-1) + \left[\sum_{j=d}^{N_2} g_j r(k+j) - \sum_{j=d}^{N_2} g_j f_{j,0} y(k) - \sum_{j=d}^{N_2} g_j f_{j,1} y(k-1) \right] \Big/ \left(\sum_{j=d}^{N_2} g_j^2 + q_0 \right) - \Delta n(k+d-1) \Big] / \hat{b}_0 \quad (25)$$

where

$$g_d = \hat{b}_0, \quad g_{d+1} = (1+\hat{a}_1)\hat{b}_0, \quad g_{N_2} = (\sum_{m=0}^{N_2-d} \hat{a}_1^m)\hat{b}_0$$

$$f_{j,0} = 1 + \hat{a}_1 + \hat{a}_1^2 + \cdots + \hat{a}_1^j, \quad f_{j,1} = -(\hat{a}_1 + \hat{a}_1^2 + \cdots + \hat{a}_1^j).$$

Figs. 8 and 9 display the set-point tracking response and the control signal of the PC-based temperature control system. The resultant maximum overshoot was less than $0.5^o C$ and the steady-state errors remained within $\pm 0.2\ ^o C$.

The second experiment was performed to examine tracking and disturbance rejection capabilities of the designed controller in the presence of a heat load change of 1000 Watts for the time duration $410 \le k \le 820$. Figs. 10 and 11 show the temperature response and control signal of the proposed controller with the setpoint $r(k) = 20^o C$ for the temperature control system. We observe in Fig. 10 that the steady-state temperature errors remained within $\pm 0.2\ ^o C$, and the proposed adaptive neural predictive control would exhibit acceptable disturbance rejection ability.

VIII. CONCLUSIONS

This paper has presented a systematic design methodology to develop an adaptive neural predictive control with integral action for a class of nonlinear SISO systems with time-delay. Such systems are assumed to be well approximated by combining linear system models with nonlinear time-varying terms. The set-point tracking and load disturbance rejection capabilities of the proposed method can be improved by slightly modifying the performance criterion function. The stability and properties of the overall closed-loop system are investigated as well. The proposed real-time control algorithm, consisting of the recursive least-squares estimator, the neural predictor, and the modified generalized predictive control law, has been successfully applied to achieve the performance specifications for two illustrative highly nonlinear systems and one physical variable-frequency oil-cooling process. Through computer simulations and experimental results, the proposed method has been proven useful and effective under the conditions of set-points and constant load changes.

ACKNOWLEDMENT

This study was supported by the National Science Council of the Republic of China under Grant NSC91-2622-E-005-008-CC3.

REFERENCES

[1]D. W. Clarke and C. Mohtadi, "Properties of generalized predictive control," *Automatica*, vol. 25, no. 6, pp. 859-875, 1989.

[2]D. W. Clarke, "Application of generalized predictive control to industrial process," *IEEE Contr. Syst. Mag*., vol. 8, pp. 49-55, Apr. 1988.

[3]B. Kouvaritakis, J. A. Rossiter, and J. R. Gossner, "Improved Algorithm for Multivariable Stable Generalised Predictive Control," *IEE Proc. - Control Theory Appl*., vol. 144, no. 4, pp. 309-312, 1997.

[4]F. Allgower, and A. Zheng, Ed., *Nonlinear Model Predictive Control*, Birkhauser Verlag, 2000.

[5]E. F. Camcho, and C. Bordons, *Model Predictive Control*, Springer-Verlag London limited, 2000.

[6]J. M. Maciejowski, *Predictive Control with Constrain*. New Jersey: Prentice-Hall, 2002.

[7]S. Huang, K. K. Tan, and T. H. Lee, *Applied Predictive Control*, London: Springer-Verlag, 2002.

[8]M. Khalid, S. Omatu, and R. Yusof, "MIMO Furnace Control with Neural Networks," *IEEE Trans. Contr. Syst. Technol*., vol. 1, pp. 238-245, 1993.

[9]M. A. Henson, and D. E. Seborg, *Nonlinear Process Control*, Prentice Hall PTR, 1997.

[10]S. Piché, B. Sayyar-Rodsari, D. Johnson, and M. Gerules, "Nonlinear Model Predictive Control Using Neural Networks," *IEEE Contr. Syst. Mag*., pp. 53-61, June 2000.

[11]K. L. Shi, T. F. Chan, and S. L. Ho, "Direct Self Control of Induction Motor Based on Neural Networks," *IEEE Trans. Ind. Applicat*., vol. 37, no. 5, pp. 1290-1298, 2001.

[12]K. Y. Zhu and X. F. Qin, "Robust nonlinear predictive control with neural network compensator," in *Proc. 35th Decision and Control Conf*., Kobe, Japan, pp. 3816-3817, 1996.

[13]B. J. Song and A. J. Koivo, "Nonlinear Predictive Control with Application to Manipulator with Flexible Forearm," *IEEE Trans. Ind. Electron*., vol. 46, no. 5, pp. 923-932, 1999.

[14]Y. Tan, and A. V. Cauwenberghe, "Neural-network-based d-step-ahead predictors for nonlinear systems with time delay," *Engineering Applications of Artificial Intelligence*, vol. 12, pp. 21-35, 1999.

[15]X. F. Qin, and K. Y. Zhu , "Robust Adaptive Decoupling Design for Generalized Predictive Control with Neural Network," in *Proc. 35th Decision and Control Conf*., Kobe, Japan, pp. 2426-2431, 1996.

[16]K. J. Åström and B. Wittenmark, *Adaptive Control*. New Jersey: Addison-Wesley, 1995.

[17]C. T. Lin and C. S. George Lee, *Neural Fuzzy Systems*. New Jersey: Prentice-Hall, 1999.

[18]C. C. Ku and K. Y. Lee, "Diagonal Recurrent Neural Networks for Dynamical System Control," *IEEE Trans. Neural Networks*, vol. 6, No. 1, pp. 144-156, 1995.

[19]X. Li, Z. Chen, and Z. Yuan, "Simple Recurrent Neural Network-Based Adaptive Predictive Control for Nonlinear Systems," *Asian Journal of Control*, vol. 4, no. 2, pp. 231-239, 2002.

[20]C. H. Huang, and C. C. Tsai, "Model reference adaptive predictive control of a variable-frequency oil-cooling machine," Proceedings of the 2002American Control Conference, vol.6, pp. 4928-4933, May 2002.

[21]C. C. Tsai, and F. J. Teng, and S. H. Lin, "Direct Self-Tuning Model Following Predictive Control of a variable-frequency oil-cooling machine," Proceedings of the 2003 American Control Conference, pp. 603-608, June 2003.

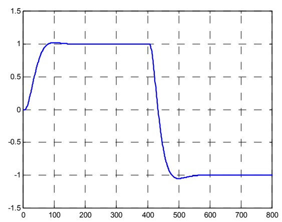

Fig. 1. Set-point tracking simulation result for the adaptive predictive controller with neural prediction

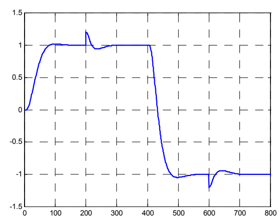

Fig. 2. Simulation result for the controller in the presence of load disturbances.

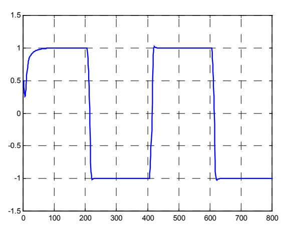

Fig. 3. Set-point tracking simulation result for the adaptive predictive controller with neural prediction.

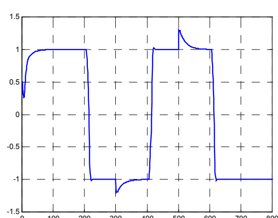

Fig.4. Simulation tracking result for the proposed controller in the presence of the load disturbances

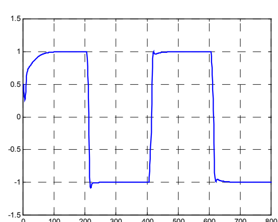

Fig. 5. Set-point tracking simulation result for the adaptive predictive controller without neural prediction

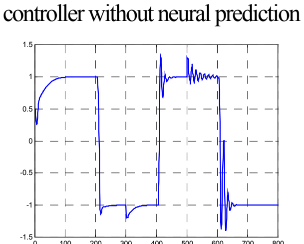

Fig. 6. Poor tracking result for the proposed controller without neural prediction for the load disturbances

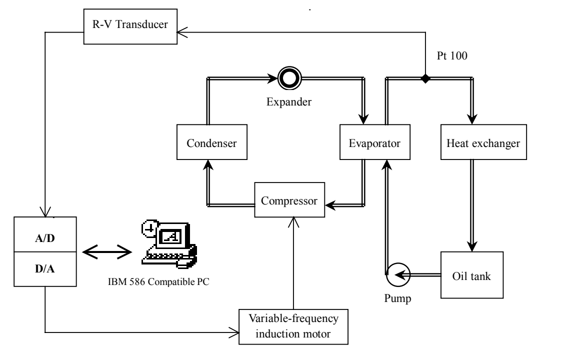

Fig. 7. A schematic diagram of the PC-based temperature control system

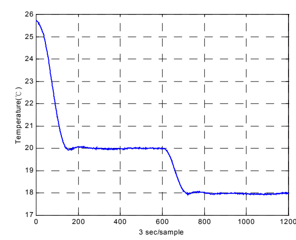

Fig. 8. Set-point tracking response of the adaptive predictive controller with neural prediction

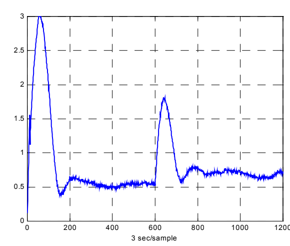

Fig. 9. Control signal of the controller

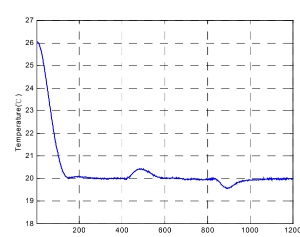

Fig. 10. Temperature response of the controller with the load changes

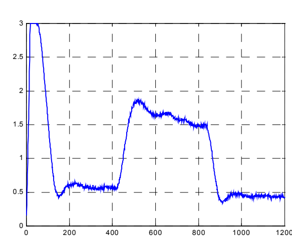

Fig. 11. Control signal of the controller with the load disturbances