Model-Reference Neural Predictive Controller Design

Chi-Huang Lu, Ching-Chih Tsai

Abstract

The paper presents a model-reference neural predictive controller design for a class of single-input single-output nonlinear systems with known time-delay. The control law is developed based on the minimization of a cost function with the model reference scheme and the neural-network-based predictive performance criterion. A real-time adaptive control algorithm, including a neural predictor and a model-reference neural predictive controller, is proposed. Simulation results reveal that the proposed controller gives satisfactory tracking and disturbance rejection performance for some illustrative discrete-time nonlinear time-delay systems.

Key words：Model-reference control, neural networks, nonlinear system, prediction control, time-delay.

Chi-Huang Lu：Assistant Professor, Department of Electrical Engineering, HIT
Ching-Chih Tsai：Professor, Department of Electrical Engineering, National Chung-Hsing University

模型參考類神經預估控制器設計

呂奇璜、蔡清池

摘要

本論文提出單輸入單輸出時間延遲非線性系統其模型參考類神經預估控制器設計。此控制法則是以模型參考方法與基於類神經網路預估準則結合之性能指標求其最小化發展出，並且內含類神經模型預估器與模型參考類神經控制器之即時適應控制演繹法則亦被陳述。離散非線性延遲系統例子中，電腦模擬顯示所提之控制器能夠滿足設定點追蹤與干擾排斥之良好性能。

關鍵字：模型參考控制、類神經網路、非線性系統、預估控制、時間延遲。

呂奇璜：修平技術學院電機系助理教授
蔡清池：國立中興大學電機系教授

1. 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 [1-5]. In many industrial processes they usually exhibit various 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 [6-10]. Khalid et al. [6] developed a feedforward multilayer neural controller for an MIMO furnace and compared its performance with other advanced controllers. Piche et al. [7] established a neural-network-based technique for constructing nonlinear dynamic models from their empirical input-output data. Zhu. et al. [8] developed a robust nonlinear predictive control with neural network compensator. Song et al. [9] explored a nonlinear predictive control with its application to a 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. [10] presented neural-network-based d-step-ahead predictors for a class of nonlinear systems with time-delay. Ren et al. [11] proposed generalized certainty equivalence adaptive model reference control for a comprehensive theory of stochastic adaptive filtering, control and identification. A simple recurrent neural network-based adaptive predictive control for nonlinear systems was proposed by Li et al. [12].

This paper will develop a novel model-reference neural predictive control for a class of SISO nonlinear time-delay systems, in which the neural-network-based predictors and controllers are constructed by using the well-known multilayer feedforward networks. The feasibility and effectiveness of the proposed method will be verified through its applications to two nontrivial nonlinear plants. The remainder of the paper is outlined as follows. Section 2 describes how to construct a multi-step neural predictor for a class of nonlinear time-delay systems. A model reference neural predictive control law is derived based on the previous neural predictor in Section 3. A real-time adaptive control algorithm is proposed in Section 4. Section 5 details the capabilities of the proposed control method utilizing computer simulations. Section 6 concludes this paper.

2. NEURAL MODEL

The section is devoted to developing the neural predictor for a class of discrete-time, single-input single-output nonlinear plants discussed in [13]. These systems are assumed to have plant inputs as , plant outputs as , and the nonlinear mappings , and , . Furthermore, represents the time delay of the systems. Generally speaking, such systems can be described by the following nonlinear autoregressive moving averaging (NARMA) models with time-delays

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

The well-known multilayer feedforward neural-networks architecture has been adopted to approximate the nonlinear systems described in [14,15]. In the sequel, this type of neural network model (NNM) will be trained to learn the nonlinear mapping $f(\cdot)$ by using the input vector

$$\boldsymbol{x}(k)=\begin{bmatrix}u(k-d) & \cdots & u(k-n_u) & y(k-1) & \cdots & y(k-n_y) & 1\end{bmatrix}^T \quad (2)$$

where the NNM has a two-layer perceptron network with n_i inputs, n_j hidden units, and an output variable. $\hat{y}(k)$ denotes the output of the NNM, and $w_{ij}(k)$ and $w_j(k)$ stand for the weighting values between the input layer and hidden layer, and the hidden layer and output layer, respectively. The NNM has the following input/output mapping relationship

$$\hat{y}(k)=\sum_{j=1}^{n_j}W_j(k)\Gamma\left(\sum_{i=1}^{n_i}w_{ij}(k)x_i(k)\right) \quad (3)$$

where $\Gamma(\chi)=1/(1+e^{-\chi})$, $n_i=n_u+n_y-d+2$, and $x_i(k)$ represents the ith entry of the input vector for the NNM. To update the weights $w_{ij}(k)$ and $w_j(k)$ of the multilayer feedforward networks, we define the following performance criterion

$$J(k)=\frac{1}{2}\big(y(k)-\hat{y}(k)\big)^2 \quad (4)$$

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

$$\boldsymbol{W}(k+1)=\boldsymbol{W}(k)-\eta\frac{\partial J(k)}{\partial \boldsymbol{W}(k)} \quad (5)$$

where η is a positive learning rate, and $\partial J(k)/\partial \boldsymbol{W}(k)$ can be calculated as follows;

$$\frac{\partial J(k)}{\partial W_j(k)}=\big(\hat{y}(k)-y(k)\big)\Gamma\big(h_j(k)\big),\quad h_j(k)=\sum_{i=1}^{n_i}\big(w_{ij}(k)x_i(k)\big) \quad (6)$$

$$\frac{\partial J(k)}{\partial w_{ij}(k)}=\big(\hat{y}(k)-y(k)\big)W_j(k)\frac{e^{-h_j(k)}}{\left(1+e^{-h_j(k)}\right)^2}x_i(k) \quad (7)$$

3.MODEL-REFERENCE NEURAL PREDICTIVE CONTROL

This section is devoted to developing a model-reference neural predictive controller (NPC) for improving tracking performance and disturbance rejection abilities of the SISO nonlinear control systems (1). The control law is derived so as to minimize the following cost function

$$J_c(k)=\frac{1}{2}\sum_{p=d}^{N}\Big(A_m(z^{-1})\big(y^*(k+p)-\hat{y}(k+p)\big)\Big)^2 \quad (8)$$

where $y^*(k)$ is an output signal of the selected reference model, and $N \geq d$. The reference model denotes

$$G_m(z^{-1}) \equiv \boldsymbol{Z}\{y^*(k)/r(k)\} = z^{-d}B_m(z^{-1})/A_m(z^{-1})$$

where $r(k)$ is an input reference signal, $A_m(z^{-1}) = 1 - a_1 z^{-1} - \cdots - a_{n_A} z^{-n_A}$, and $B_m(z^{-1}) = b_0 + b_1 z^{-1} + \cdots + b_{n_B} z^{-n_B}$. Note that $G_m(z^{-1})$ is assumed an asymptotically stable system and $G_m(z^{-1})\big|_{z=1} = 1$.

In order to design the proposed method, we construct the model-reference neural predictive controller using the multilayer feedforward neural-networks architecture, and use a different input vector as

$$\boldsymbol{X}(k) = \left[u(k-1) \quad y(k) \quad \cdots \quad y(k-n_y+1) \quad 1\right]^T . \tag{9}$$

The controller is mathematically expressed by

$$u(k) = \sum_{j=1}^{m_j} V_j(k)\Gamma\left(\sum_{i=1}^{m_i} v_{ij}(k)X_i(k)\right) \tag{10}$$

where $\nu_{ij}(k)$ and $V_j(k)$ are the weighting values for the NPC between the input layer and the hidden layer, and the hidden layer and the output layer, respectively. Furthermore, $m_i=n_y+2$ and $X_i(k)$ represents the ith entry of the input vector for the NPC. The controller's weights with positive learning rates η_c are updated by the dynamic backpropagation algorithm as below

$$\boldsymbol{V}(k+1) = \boldsymbol{V}(k) - \eta_c \frac{\partial J_c(k)}{\partial \boldsymbol{V}(k)} \tag{11}$$

With Eqs. (8) and (11), we define $e_c(k) \equiv \sum_{p=d}^{N}\left(A_m(z^{-1})\left((r(k+p) - \hat{y}(k+p)\right)\right)$, and obtain

$$\frac{\partial J_c(k)}{\partial \boldsymbol{V}(k)} = e_c(k)\frac{\partial e_c(k)}{\partial \boldsymbol{V}(k)} \tag{12}$$

Let $u(k) = u(k+1) = \cdots = u(k+N)$. Then $\partial J_c(k)/\partial V(k)$ can be obtained from

$$\frac{\partial J_c(k)}{\partial V_j(k)} = \left(\sum_{p=d}^{N}\left(A_m(z^{-1})\hat{y}(k+p)\right)y_u(k+p) - \sum_{p=0}^{N-d}\left(B_m(z^{-1})r(k+p)\right)y_u(k+p)\right)\Gamma\left(g_j(k)\right) \tag{13}$$

$$\frac{\partial J_c(k)}{\partial v_{ij}(k)} = \left(\sum_{p=d}^{N}\left(A_m(z^{-1})\hat{y}(k+p)\right)y_u(k+p) - \sum_{p=0}^{N-d}\left(B_m(z^{-1})r(k+p)\right)y_u(k+p)\right)V_j(k)\widetilde{G}(k)X_i(k) \tag{14}$$

where

$$y_u(k+p) = \sum_{j=1}^{n_j} W_j(k+p)\frac{e^{-h_j(k+p)}}{\left(1+e^{-h_j(k+p)}\right)^2}w_{1j}(k+p)$$

$$h_j(k+p) = \sum_{i=1}^{n_i}\left(w_{ij}(k+p)x_i(k+p)\right)$$

$$\widetilde{G}(k) = \frac{e^{-g_j(k)}}{\left(1+e^{-g_j(k)}\right)^2}, \quad g_j(k) = \sum_{i=1}^{m_i}\left(v_{ij}(k)X_i(k)\right).$$

4. REAL-TIME ADAPTIVE CONTROL ALGORITHM

To make the controller exhibit adaptive characteristics, we include the neural predictor and controller in the control loop, and propose the following real-time model-reference adaptive neural predictive control algorithm.

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

Step 2) Set d, n_y, n_u, N, n_j, m_y, η , η_c , $A_m(z^{-1})$, $B_m(z^{-1})$.

Step 3) Off-line train neural predictor and NPC.

Step 4) Measure the system output .

Step 5) On-line learn neural predictor and NPC.

Step 6) Compute the control signal $u(k)$.

Step 7) Output $u(k)$ to the controlled plant.

Step 8) Repeat steps 4-7.

5. COMPUTER SIMULATIONS

The objectives of the following simulations are to explore the feasibility and effectiveness of the neural predictive control for the underlying two nonlinear systems. The simulation studies also include investigation of the effects of constant load disturbances on the performance of the proposed controller.

Example 1: The following modified nonlinear system is taken from a model in [16], and given by

$$y(k) = 0.0125 + 0.9831y(k-1) + 0.0853u(k-2)$$

$$0.0288y(k-1)u(k-2) + 0.0176y(k-1)^2u(k-3)\ .$$

The simulation was performed for a time-varying reference input $r(k)$, some parameters given by

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

$$N = 10\ ,\quad \eta = 0.1\ ,\quad \eta_c = 1\ ,\quad n_j = m_j = 10$$

$$A_m(z^{-1}) = 1 - 0.9z^{-1}\ ,\quad B_m(z^{-1}) = 0.1\ .$$

Figs. 1 and 2 show the response and control signal of the model-reference 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 the proposed controller with load disturbances, we let the mathematical model be perturbed to

$$(k) = 0.0125 + 0.9831y(k-1) + 0.0853u(k-2)$$

$$0.0288y(k-1)u(k-2) + 0.0176y(k-1)^2u(k-3) + \xi(k)$$

where $\xi(k) = \begin{cases} 0.02, & 200 \le k \le 600 \\ 0, & \text{otherwise.} \end{cases}$

Fig. 3 depicts the simulation result for the proposed controller with the load disturbances. Consequently, the model-reference adaptive neural predictive controller demonstrated a good disturbance rejection capability.

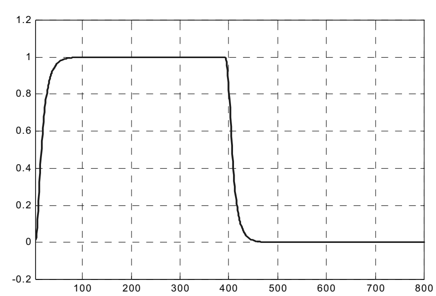

Fig. 1. Set-point tracking simulation result for the model-reference neural predictive controller

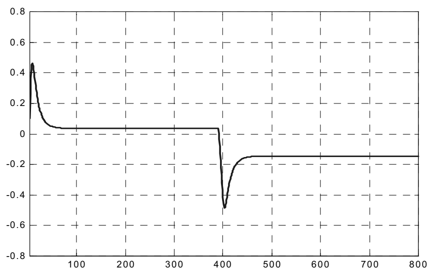

Fig. 2. Simulated control signal of the controller

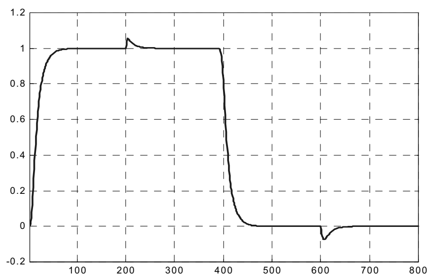

Fig. 3. Simulation result for the controller in the presence of load disturbances

Example 2: The following modified nonlinear system is modified from [12], and described by

$$y(k)=\frac{y(k-1)}{1+y^2(k-1)}+5u(k-6)+u(k-7)$$

The simulation was conducted for a time-varying reference input $r(k)$, and the parameters given by

$$r(k)=\begin{cases}1, & 0<k\le 200\\ -1, & 200<k\le 400\\ 1, & 400<k\le 600\\ -1, & 600<k\le 800\end{cases},\quad \eta=0.1,\quad \eta_c=0.001,\quad n_j=m_j=10$$

$$N=10,\quad A_m(z^{-1})=1-0.9z^{-1},\quad B_m(z^{-1})=0.1.$$

Figs. 4 and 5 show the response and control signal of the model-reference neural predictive controller under set-point changes. Fig. 4 reveals that the proposed controller is capable of giving a much better setpoint tracking performance in comparison with the result controlled by the method given in [12].

In order to explore the effect of load disturbance on the performance of the proposed controller, we add an external load change for the system, that is,

$$y(k)=y(k-1)/(1+y^2(k-1))+5u(k-6)+u(k-7)+\xi(k)$$

where $\xi(k)$=-0.2 at $300\le k<500$ and $\xi(k)$=-0.1 at $k\ge 500$. Fig. 6 depicts the simulation result for the proposed controller with the external load.

6. CONCLUSIONS

This paper has presented a systematic design methodology for developing a model-reference neural predictive control for a class of nonlinear SISO systems with time-delay. The proposed controller is composed of a multi-step-ahead predictor and a model-reference neural predictive controller. The set-point tracking and load disturbance rejection capabilities of the proposed method can be improved by adjusting the parameters in the criterion function.

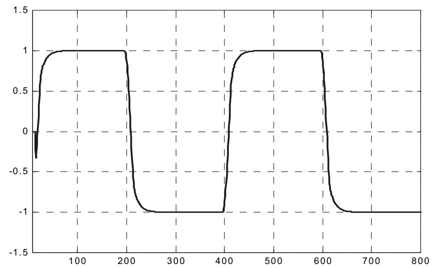

Fig. 4. Set-point tracking simulation result using the model-reference neural predictive controller

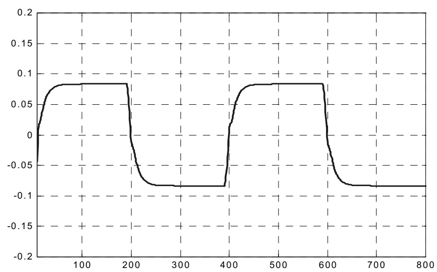

Fig. 5. Simulated control signal of the controller

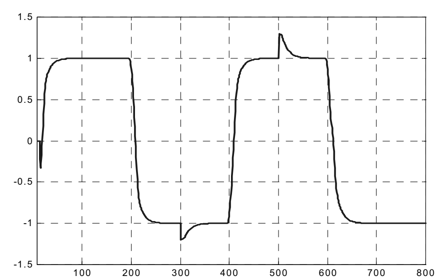

Fig. 6. Simulation tracking result using the controller in the presence of the load changes

The proposed control algorithm has been successfully applied to achieve tracking and regulation performance specifications for two illustrative nonlinear systems. Through computer simulations results, the proposed method has been proven useful and effective under the conditions of set-point and load changes.

7. REFERENCES

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

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

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

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

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

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

[7] S. Piche, B. Sayyar-Rodsari, D. Johnson,

and M. Gerules, “Nonlinear Model Predictive Control Using Neural Networks,,” *IEEE Contr. Syst. Mag., 2000*, pp. 53-61.

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

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

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

[11] W. Ren, and P. R. Kumar, “Stochastic Adaptive Predictive and Model Reference Control,” *IEEE Trans. Automatic Control*, 1994, vol. 39, no. 10, pp. 2047-2060.

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

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

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

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

[16] C. H. Lu, and C. C. Tsai, “Adaptive Predictive Control with Neural Prediction for a Class of Nonlinear Systems with Time-Delay,” *Proceeding of the 2004 American Control Conference*, 2004, pp. 908-913.