Nonlinear Networked Control Systems with Random Nature using Neural Approach and Dynamic Bayesian Networks

Hyun Cheol Cho and Kwon Soon Lee*

Abstract: We propose an intelligent predictive control approach for a nonlinear networked control system (NCS) with time-varying delay and random observation. The control is given by the sum of a nominal control and a corrective control. The nominal control is determined analytically using a linearized system model with fixed time delay. The corrective control is generated online by a neural network optimizer. A Markov chain (MC) dynamic Bayesian network (DBN) predicts the dynamics of the stochastic system online to allow predictive control design. We apply our proposed method to a satellite attitude control system and evaluate its control performance through computer simulation.

Keywords: Dynamic Bayesian network, NCS, neural network, random time-delay.

1. INTRODUCTION

An NCS is currently popular in the industry because many dynamic systems progressively involve a complicated structure and are controlled by remote controllers via communication networks. Such a network controlled framework has multiple economical and technological advantages. However, a challenging issue in a NCS is the effect of the time-varying delay between the remote controller and the targeted plant. Such delay is caused by system complexity and the use of a randomly varying communication network. Time delay often degrades control performance and can even cause instability. Thus, it is necessary to consider the effect of the delay on control design for a NCS.

Recently, engineers have actively investigated control solutions for such a problem [1-3]. Recent publications also include tutorial articles that introduce the control and stability analysis for linear time-invariant NCSs. The simplest design approach is to use an augmented model to represent the effect of the time delay [2,3]. In [4], the authors presented a control design based on an augmented discrete-time system model for periodic delay and extended it to non-identical delay in [5]. In [6,7], a queuing theory approach was applied to construct a state predictor and the state probability was calculated using a First-Input-First-Output (FIFO) model. In [8-10], the authors applied optimal stochastic control, robust control, and system perturbation theory, respectively, to overcome the effects of the delay in an NCS. In [11], sampling time scheduling where the sampling interval is arbitrarily changed online was used to maintain a stable NCS. This concept was extensively utilized for multi-dimensional NCSs [12]. Proportional-Integral (PI) control was designed for time-delayed systems in [13], and its parameters were adaptively updated to cope with time-varying delay.

More recent research has considered more complex NCSs and more sophisticated control methodologies. In [14], the authors studied a Lyapunov-based control for a nonlinear MIMO NCS in which the system model included a fixed time delay and the controller law was derived based on Lyapunov stability theory. State feedback control was utilized for a linear continuous NCS in [15,16]. Finally, in [17] the authors used stochastic design of a random NCS whose two time delays (sensor and actuator) were estimated using two homogeneous MCs.

In most studies to date, the authors usually dealt with NCSs characterized by linear time-invariant, fixed time delay, and deterministic behavior. Although the proposed methodologies were successfully implemented in simulation experiments, real-time control errors are unavoidable in practice due to the nonlinear and random nature of NCSs. A recent survey [1] recognized the need to change the controller design framework to obtain more practical adaptive controller designs. However, practical NCS

Manuscript received March 13, 2007; revised July 21, 2007 and October 25, 2007; accepted February 19, 2008. Recommended by Editorial Board member Won-jong Kim under the direction of Editor Young-Hoon Joo. This work was supported by the Korea Science and Engineering Foundation (KOSEF) through the National Research Lab. Program funded by the Ministry of Science and Technology (No. M1030000030306J000030310).

Hyun Cheol Cho and Kwon Soon Lee are with the Dept. of Electrical Engineering, Dong-A University, 840, Hadan 2-dong, Saha-gu, Pusan 604-714, Korea (e-mails: hyunccho @gmail.com, kslee@dau.ac.kr).
* Corresponding author.

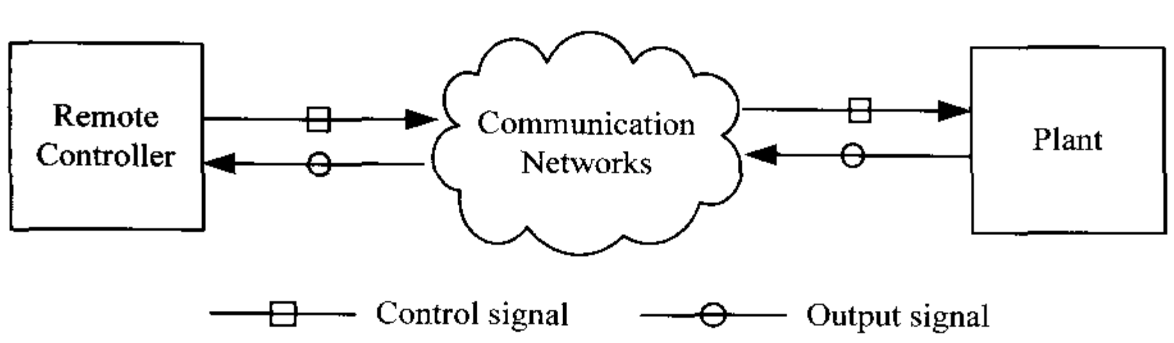


Fig. 1. Networked control systems.

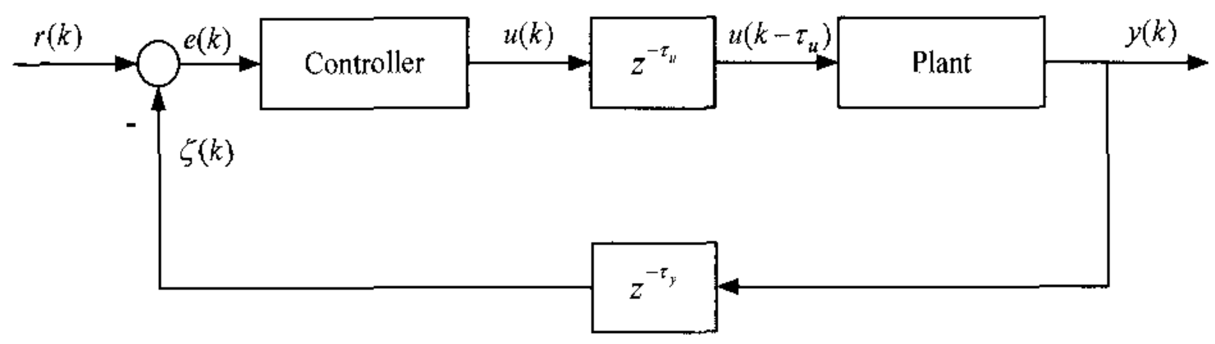


Fig. 2. Typical control systems with time delays.

dynamics are difficult to analytically model. Moreover, changes in the system environment are not completely predictable and cannot be easily accounted for in the design procedure.

This paper contributes an intelligent control approach for a nonlinear NCS with time-varying delay and random observation, which is hardly dealt with to date. The control is given by the sum of a nominal control and a corrective control. The nominal control is determined analytically using a linearized system model with fixed time delay. The corrective control is generated online by a neural network iteratively trained using data from the actual nonlinear stochastic system. The neural network comprises a nonlinear Infinite Impulse Response (IIR) module whose input vector includes: the current and time-lagged errors, the recurrent outputs, and the constant bias, whose output is the control vector. In addition, we construct a DBN model to predict the dynamics of the stochastic system online for allowing predictive control design. Thus, a predict signal generated from the DBN is forwarded to the neural network tuner as input information. We apply our proposed method to a satellite attitude control with nonlinear stochastic dynamics and Poisson distributed time delay. Simulation experiments demonstrate the superiority of the proposed neural approach to traditional NCS controllers.

This paper is organized as follows: In Section 2, we describe a nonlinear stochastic NCS with random time delay. We propose a control design for the system in Section 3. DBN modeling and neural predictive control are derived in Section 4 and 5 respectively. A simulation example is provided in Section 6 and conclusions are given in Section 7.

2. NONLINEAR STOCHASTIC NCS

We consider a discrete nonlinear NCS with random time delay and stochastic observations. A general mathematical expression for single-input-single-output (SISO) systems with control and observation time delay τ_u, τ_y is given by

$$\begin{cases} x(k+1) = f(x(k), u(k-\tau_u), k) \\ y(k) = g(x(k), \omega, k) \\ \zeta(k) = y(k-\tau_y), \end{cases} \qquad (1)$$

where $x \in \Re^n$ is the state vector, $y, \zeta \in \Re$ are the output and measured output, respectively, f and g are continuous nonlinear functions of all their arguments, and ω is a scalar Gaussian noise process. Clearly, the system outputs are non-Gaussian due to their nonlinearity. The system error, $e(k) = r(k) - \zeta(k)$, propagated to the controller, is obtained by comparing the observation ζ with the reference value r. A block diagram of a typical nonlinear NCS is indicated in Fig. 2. Here, we assume two bounded random delays τ_u and τ_y with compact support

$$\tau_u \in [\tau_{u,\min}, \tau_{u,\max}], \; \tau_y \in [\tau_{y,\min}, \tau_{y,\max}], \qquad (2)$$

where system stability is obtained within the bounds. Applying an expectation operator $E(\tau_u)$ and $E(\tau_y)$, we alternatively express (2) as

$$\tau_u = T_u - \Delta\tau_u, \; \tau_y = T_y - \Delta\tau_y, \qquad (3)$$

where

$$\begin{aligned} &T_u := E(\tau_u), && T_y := E(\tau_y), \\ &\Delta\tau_u := E(\tau_u) - \tau_u, && \Delta\tau_y := E(\tau_y) - \tau_y. \end{aligned} \qquad (4)$$

The expected values of the delays T_u and T_y in (4) can be estimated from experimental data and used to design a nominal controller. However, the perturbations in the time delays require a more complex design for satisfactory NCS performance.

3. CONTROLLER DESIGN

We express the nonlinear state and output of (1) as the sum of a nominal state and a nonlinear perturbation, i.e.,

$$\begin{cases} x(k+1) = x^*(k+1) + \Delta x(k+1) \\ y(k) = y^*(k) + \Delta y(k), \end{cases} \qquad (5)$$

where x^* is the state of a nominal model and y^* is the nominal output, while Δx and Δy represent nonlinear perturbations. The nominal state x^* corresponds to fixed time delay T_u in (4) and the perturbed state Δx results from the random time delay $\Delta\tau_u$. Thus, the state equation is rewritten as

$$x(k+1) = F(x^*, u^*(k-T_u), k) + \Delta f(x, u(k-\tau_u)), \qquad (6)$$

where F is a linear function of all its arguments and ΔF is a nonlinear perturbation. Similarly, for the

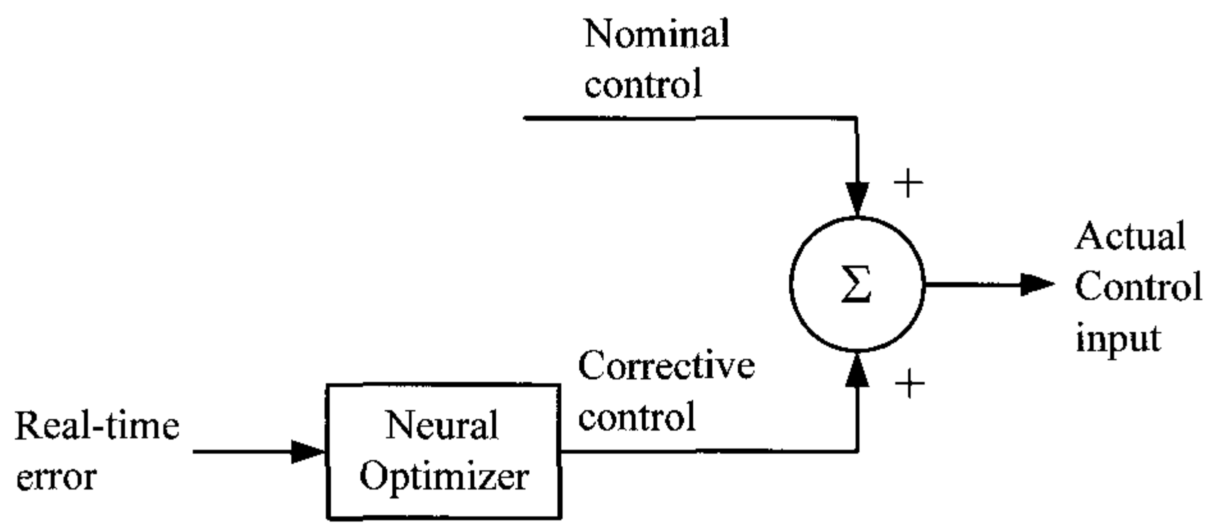


Fig. 3. NCS control configuration.

output model of (5), we have

$$y(k)=G(x^*,w^*,k)+\Delta g(x,w,k), \tag{7}$$

where G is linear and Δg is nonlinear. We also separate the observation into a nominal observation $\zeta^*(k)$ and a nonlinear perturbation $\Delta\zeta(k)$ as

$$\zeta(k)=\zeta^*(k)+\Delta\zeta(k). \tag{8}$$

Based on the observation equation (1), we have

$$\zeta(k)=y^*(k-\tau_y)+\Delta y(k-\tau_y). \tag{9}$$

In terms of the expected delay and the random perturbation of (4), we have the observation equation

$$\begin{aligned}\zeta(k)&=y^*(k-T_y+\Delta\tau_y)+\Delta y(k-\tau_y)\\&=y^*(k-T_y)+\Delta\tilde{y}(k-\tau_y).\end{aligned} \tag{10}$$

In summary, a nonlinear NCS model is separated into a nominal linear model with fixed time delay and a nonlinear stochastic perturbation including random time delay. Correspondingly, we adopt a two-step control design procedure. We firstly design a state feedback control for the nominal model and then adaptively correct its control online using a neural network. Hence, the control vector is the sum of the nominal and corrective control vectors as illustrated in Fig. 3.

3.1. Linear control for nominal time delay model

We use the deterministic nominal model

$$\begin{cases}x^*(k+1)=Ax^*(k)+Bu(k-T_u)\\ y^*(k)=Cx^*(k)\\ \zeta^*(k)=y^*(k-T_y)\end{cases} \tag{11}$$

to design a state feedback control based on pole placement for two cases $T_y=0$, $T_u\neq 0$, and $T_u=0$, $T_y\neq 0$, where A is a n-by-n nominal state matrix, B is a n-by-1 control matrix, and C is a n-by-1 output matrix. Then, we determine the appropriate control parameter by seeking a common region for the two solutions.

3.1.1 Control delay only ($T_y=0$, $T_u\neq 0$)

For $T_y=0$ in (11), we have

$$\begin{cases}x^*(k+1)=Ax^*(k)+Bu(k-T_u)\\ \zeta^*(k)=Cx^*(k).\end{cases} \tag{12}$$

An augmented state-space model [18] is expressed as

$$\begin{cases}\mathrm{X}_c(k+1)=\Pi_c\mathrm{X}_c(k)+\Gamma_c u(k)\\ \zeta(k)=\Sigma_c\mathrm{X}_c(k),\end{cases} \tag{13}$$

where the augmented state vector is

$$\mathrm{X}_c(k)=\begin{bmatrix}x^*(k) & x_1(k) & x_2(k) & \cdots & x_{T_u}(k)\end{bmatrix}^T \in\Re^{(n+Tu)}$$

and the corresponding matrices are given by

$$\Pi_c=\begin{bmatrix}A & B & \mathbf{0}_{n\times(T_u-1)}\\ \mathbf{0}_{(T_u-1)\times n} & \mathbf{0}_{(T_u-1)\times 1} & \mathbf{I}_{(T_u-1)}\\ \mathbf{0}_{1\times n} & 0 & \mathbf{0}_{1\times(T_u-1)}\end{bmatrix}\in\Re^{(n+T_u)\times(n+T_u)},$$

$$\Gamma_c=\begin{bmatrix}0 & \cdots & 0 & 1\end{bmatrix}^T\in\Re^{(n+T_u)},$$

$$\Sigma_c=\begin{bmatrix}C & 0 & \cdots & 0\end{bmatrix}\in\Re^{1\times(n+T_u)}.$$

We have the linear state feedback control law

$$u(k)=-\mathrm{K}_c\,\mathrm{X}_c(k) \tag{14}$$

with the gain matrix

$$\mathrm{K}_c=\begin{bmatrix}\kappa_c & \kappa_{c,n+1} & \cdots & \kappa_{c,n+T_u}\end{bmatrix}\in\Re^{1\times(n+T_u)}. \tag{15}$$

Substituting (14) into (13), we have the new state matrix

$$\Pi^c=\Pi_c-\Gamma_c\mathrm{K}_c=\left[\begin{array}{ccc}A & B & \mathbf{0}_{n\times(T_{u-1})}\\ \mathbf{0}_{(T_{u-1})\times n} & \mathbf{0}_{(T_{u-1})\times 1} & 1\\ \hline & -\mathrm{K}_c & \end{array}\right] \tag{16}$$

and its characteristic equation

$$zI-\Pi^c=0. \tag{17}$$

We select a stable control gain K_c satisfying $|z|<1$ with appropriate closed-loop dynamics.

3.1.2 Output delay only ($T_y\neq 0$, $T_u=0$)

Similarly, for $T_u=0$, we have

$$\begin{cases} x^*(k+1) = Ax^*(k) + Bu(k) \\ y^*(k) = CX^*(k) \\ \zeta^*(k) = y^*(k-\tau_y) \end{cases} \quad (18)$$

and its augmented model [18] is given by

$$\begin{cases} X_o(k+1) = \Pi_o(k) + \Gamma_o u(k) \\ \zeta(k) = \Sigma_o X_o(k), \end{cases} \quad (19)$$

where similarly we have a new state vector

$$X_o(k) = \begin{bmatrix} x^*(k) & x_1(k) & x_2(k) & \cdots & x_{T_y}(k) \end{bmatrix}^T \in \Re^{(n+T_y)},$$

$$\Pi_o = \begin{bmatrix} A & \mathbf{0}_{n\times(T_y-1)} & \mathbf{0}_{n\times 1} \\ C & \mathbf{0}_{1\times(T_y-1)} & 0 \\ \mathbf{0}_{(T_y-1)\times n} & \mathbf{I}_{T_y-1} & \mathbf{0}_{(T_y-1)\times 1} \end{bmatrix} \in \Re^{(n+T_y)\times(n+T_y)},$$

$$\Gamma_o = \begin{bmatrix} B & 0 & \cdots & 0 \end{bmatrix}^T \in \Re^{(n+T_y)},$$

$$\Sigma_o = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \in \Re^{1\times(n+T_y)}.$$

Likewise, we have the state feedback control

$$U_o(k) = -K_o X_o(k), \quad (20)$$

where a control gain matrix

$$K_o = \begin{bmatrix} \kappa_o & \kappa_{o,n+1} & \cdots & \kappa_{o,n+T_y} \end{bmatrix} \in \Re^{1\times(n+T_y)}. \quad (21)$$

Finally, a new state matrix including the control parameter is given by

$$\Pi^o = \Pi_o - \Gamma_o K_o$$
$$= \begin{bmatrix} A - B\kappa_o & -B\kappa_{o,n+1} & -B\kappa_{o,n+2} & \cdots & -B\kappa_{o,n+T_y-1} & -B\kappa_{o,n+T_y} \\ C & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}. \quad (22)$$

Similarly, we determine K_o for stability based on its characteristic equation

$$zI - \Pi^c = 0. \quad (23)$$

3.1.3 Solution of control for a nominal time delay

From each control solution of the state feedback above, we finally select an appropriate control gain which satisfies both models by seeking a common region for the two solutions, i.e.,

$$K \in \{K_c \cap K_o\}, \quad (24)$$

where K is the control matrix applied to the nominal system with both control and observation delays.

3.2. Correction of the control gain by neural network

In practice, control design using a nominal system model becomes suboptimal due to model perturbations. We solve this problem using a neural network approach. The key idea is to optimally determine a corrective control through learning using a set of n neural networks as shown in Fig. 4. Each network has the recurrent structure shown in Fig. 5 where the output signal is feedback to an input neuron. This neural model is IIR framework appropriate to construct a stable controller due to its feedback realization [18]. The input vector of each network commonly includes the system error, the time-lagged error, and a constant bias. The output is the corrective control. In Fig. 5, each network output is expressed as

$$\tilde{\kappa}_i = \phi_i(w_i E + v_i \Omega_i + b_i), \quad i = 1, \cdots, (n+T), \quad (25)$$

where we let $T := T_u = T_y$ for simplicity in offline design procedure. We also have the network weight matrices,

$$w_i = [w_{i0}, \cdots, w_{il}], \quad v_i = [v_{i1}, \cdots, v_{im}], \quad (26)$$

the input vector,

$$E = [e(k), \cdots, e(k-1)]^T, \quad \Omega_i = [\tilde{\kappa}_i(k-1), \cdots, \tilde{\kappa}_i(k-m)]^T \quad (27)$$

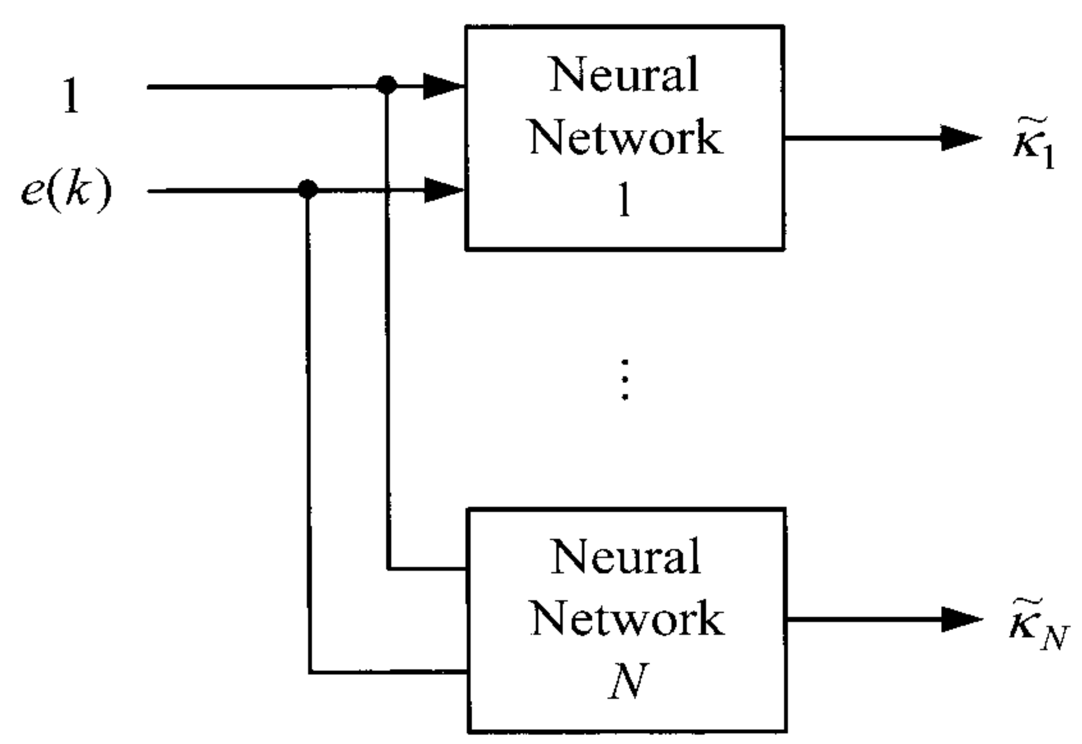


Fig. 4. A neural module for an optimizer.

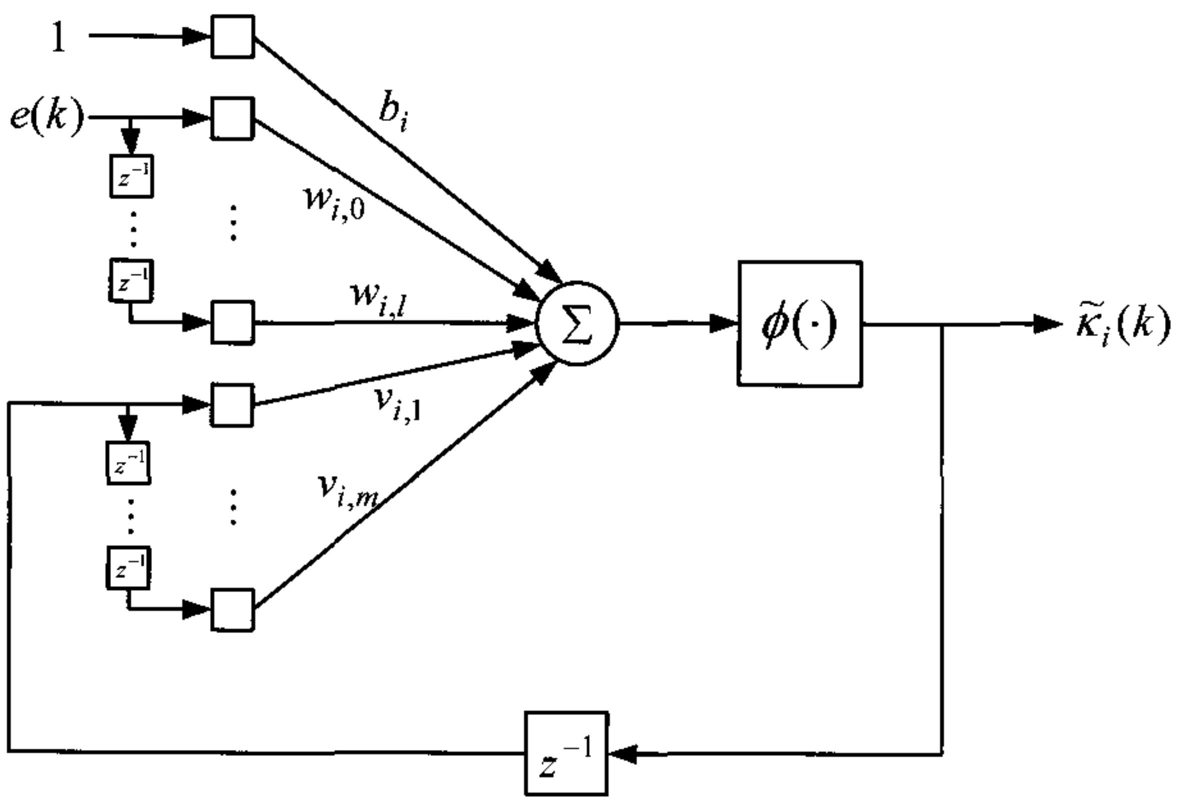


Fig. 5. Structure of a nonlinear IIR network.

and the bias b_i. For simplicity, the activation function of the ith network is

$$\phi_i = \alpha_1^i \tanh(\alpha_2^i \tilde{\kappa}_i), \quad \alpha_1^i, \alpha_2^i > 0. \tag{28}$$

The control objective is to minimize the system error with respect to the network parameters θ_i = elements of $\{w_i, v_i, b_i\}$. To derive a learning rule of the parameters, we first define an objective function

$$J = \arg\min_{\theta_i} \frac{1}{2} e^2(k) = \frac{1}{2}\left(r(k) - \zeta(k)\right)^2. \tag{29}$$

Applying steepest-descent optimization, the adjustment rule of the parameter vectors are given by

$$\theta_i(k+1) = \theta_i(k) - \eta \frac{\partial J}{\partial \theta_i}, \quad i = 1, \cdots, n+T, \tag{30}$$

where $\eta \in (0,1)$ is the learning rate. Using the chain rule, we expand the partial derivative in (30) as

$$\frac{\partial J}{\partial \theta_i} = \frac{\partial J}{\partial e} \frac{\partial e}{\partial \zeta} \frac{\partial \zeta}{\partial u} \frac{\partial u}{\partial \kappa_i} \frac{\partial \kappa_i}{\partial \tilde{\kappa}_i} \frac{\partial \tilde{\kappa}_i}{\partial \theta_i}. \tag{31}$$

Solving each differential term in (31), we have

$$\frac{\partial J}{\partial \theta_i} = e \frac{\partial \zeta}{\partial u} x_i \frac{\partial \tilde{\kappa}_i}{\partial \theta_i}, \tag{32}$$

where

$$\frac{\partial \tilde{\kappa}_i}{\partial \theta_i} = \begin{cases} \phi_i'(\cdot) E_i, & \text{if } \theta_i = w_i \\ \phi_i'(\cdot) \Omega_i, & \text{if } \theta_i = v_i \\ \phi_i'(\cdot), & \text{if } \theta_i = b_i, \end{cases} \tag{33}$$

and the system *Jacobian* is approximated [19] as

$$\frac{\partial \zeta}{\partial u} = \frac{\zeta(k) - \zeta(k-1)}{u(k) - u(k-1)}. \tag{34}$$

Finally, the adjustment rules of the network parameters in (30) are respectively obtained as

$$w_i(k+1) = w_i(k) + \eta e \left(\frac{z(k) - z(k-1)}{u(k) - u(k-1)} \right) \phi'(\cdot) \zeta_1, \quad i = 1, \cdots, n+T, \tag{35}$$

$$v_i(k+1) = v_i(k) + \eta e \left(\frac{z(k) - z(k-1)}{u(k) - u(k-1)} \right) \phi'(\cdot) \zeta_2, \quad i = 1, \cdots, n+T, \tag{36}$$

$$b(k+1) = b(k) + \eta e \left(\frac{z(k) - z(k-1)}{u(k) - u(k-1)} \right) \phi'(\cdot). \tag{37}$$

4. DBN MODELING

We model the random output signal in (1) using a DBN to design a predictive control system. In Section 2, we observe that the output is a non-Gaussian random variable. DBN is particularly suited to modeling stochastic systems with non-Gaussian statistics [20]. In this paper, we adopt the DBN modeling scheme of [21] with a discrete MC model. As a first step in DBN modeling, we discretize the amplitude of the analog observation in (1) as

$$\zeta(k) \in \{\zeta_1(k), \zeta_2(k), \cdots, \zeta_N(k)\}. \tag{38}$$

The corresponding probability is given by

$$p(\zeta(k)) \in \{p(\zeta_1(k)), p(\zeta_2(k)), \cdots, p(\zeta_N(k))\} \tag{39}$$

subject to the constraint

$$\sum_{i=1}^{N} p(\zeta_i(k)) = 1. \tag{40}$$

We represent the random observation using the MC model of Fig. 6. The parameters a_{ij} of this model are the conditional probabilities of transition from the observation at k-1 to the observation at k, defined by

$$a_{ij}(k) = p(\zeta_i(k) \mid \zeta_j(k-1)), \quad i, j = 1, \cdots, N. \tag{41}$$

These parameters are constrained by

$$\sum_{i=1}^{N} a_{ij}(k) = 1, \quad j = 1, \cdots, N. \tag{42}$$

This DBN parameter must be optimally estimated online based on the observation sequence via DBN learning to stochastically model the system dynamics [20]. We adopt the estimation algorithm of [21] for our DBN modeling. We briefly describe the online parameter estimation algorithm of [21]. First, the DBN parameter in (41) is alternatively defined as

$$a_{ij}(k) = \alpha\, m_{ij}(k), \quad i, j = 1, \cdots, N, \tag{43}$$

where m_{ij} is average likelihood values. This is expressed in recursive from as

$$m_{ij}(k)=\left(\frac{k-1}{k}\right)m_{ij}(k-1)+\left(\frac{1}{k}\right)\gamma_{ij}(k), \quad i,j=1,\cdots,N, \tag{44}$$

which is sequentially updated selecting γ_{ij} according to the following rule

$$\gamma_{ij}(k)=\begin{cases} c, & if\ \zeta(k)=i\ \&\ \zeta(k-1)=j \\ 0, & otherwise, \end{cases} \tag{45}$$

where $c>0$. Applying a sliding window to emphasize more recent data, we rewrite (44) as

$$m_{ij}(k)=\left(\frac{N_w-1}{N_w}\right)m_{ij}(k-1)+\left(\frac{1}{N_w}\right)\gamma_{ij}(k), \quad i,j=1,\cdots,N, \tag{46}$$

where $N_w > 0$.

5. PREDICTIVE CONTROL BY DBN

We have the posterior probability vector of the DBN model in Fig. 6.

$$p(\zeta(k))=A(k)p(\zeta(k-1)), \tag{47}$$

where the time-varying stochastic matrix $A=[a_{ij}]$, $i,j=1,\cdots,N$, is updated through the estimation procedure of Section 4. The state probability is recursively computed by multiplying the stochastic matrix by the prior probability vector. Consequently, if the initial probability vector $p(z(k_0))$ where $k_0<k$ is known, we obtain the new probability of the state from recursive calculation based on observation data. Moreover, assuming $A(k+i)=A(k)$ where $i>0$, by recursively applying the posterior probability, we obtain an i-ahead predictive state probability at current time k, i.e.,

$$p(\zeta(k+i))=A^{i+1}p(\zeta(k-1)), \quad i>0. \tag{48}$$

The elements of A converge asymptotically to a stationary distribution in (48) and the posterior probability becomes fixed [20]. We compute future states by seeking the maximum probability in (48), that is,

$$s=\max\{p(\zeta_1(k+i)),p(\zeta_2(k+i)),\cdots,p(\zeta_N(k+i))\}, \tag{49}$$

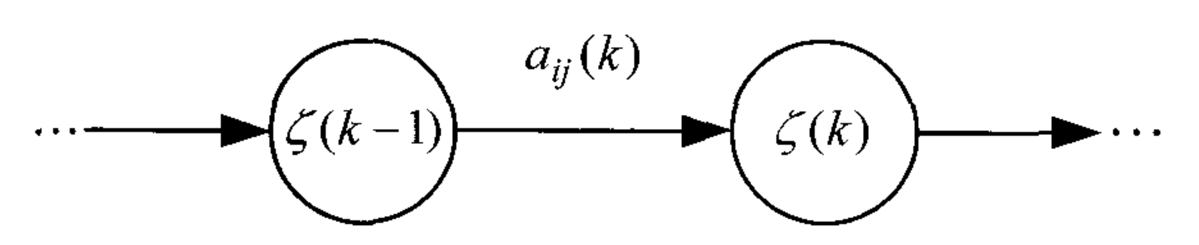


Fig. 6. A DBN model of the random observation.

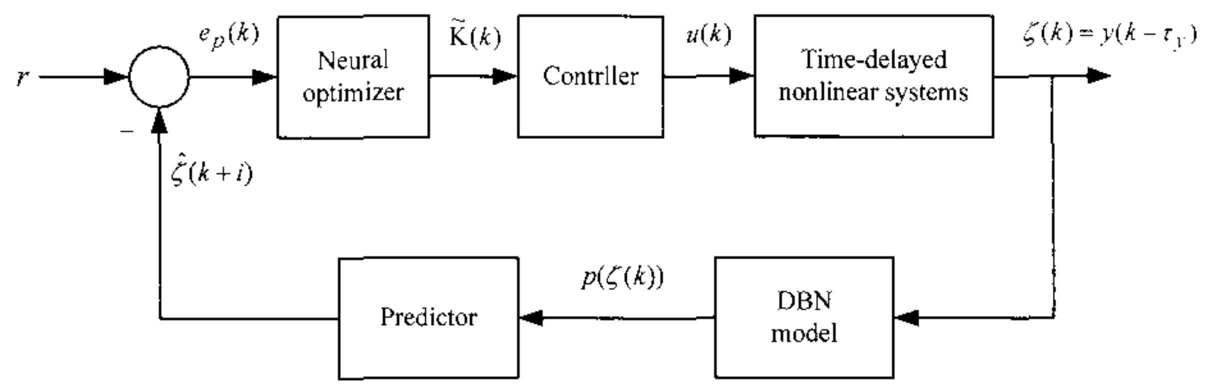


Fig. 7. Predictive control with DBN model.

where $i>0$ and $s\in[1,N]$. The predicted observation $\hat{\zeta}$ from (49) is used to predict the error

$$e_p(k)=r-\hat{\zeta}(k+i). \tag{50}$$

The error estimate is essential for enhancing NCS control performance [1,2]. Fig. 7 shows a block diagram of our proposed predictive control systems using a DBN model for the NCS.

6. NUMERICAL EXAMPLE

We apply our proposed control method to satellite attitude control. A satellite system must have the correct attitude for proper orientation with respect to the Earth. This is remotely controlled, receiving a control signal and sending its state to a control center on the Earth. We refer to the dynamic equations for satellite attitude in [22], but modify them for the purpose of our simulation to

$$\begin{cases} x_1(k+1)=x_1(k)+T_s x_2(k)+c_1\cos(k)+0.5T_s^2 u(k) \\ x_2(k+1)=x_2(k)+c_2 x_1(k)x_2(k)+T_s u(k) \\ y(k)=x_1(k)+c_3(k)\omega^3(k), \end{cases} \tag{51}$$

where the states x_1 and x_2 are the angle and the angular velocity of a satellite, T_s is the sampling period, c_1 and c_2 are constant, c_3 is a random variable, ω is Gaussian noise, and u is control input. The first and second equations of (51) represent dynamics of the angle and the angular velocity of a satellite system respectively, which are composed of linear and nonlinear terms. An output signal in the third term of (51) is deteriorated by non-Gaussian noise. We select two fixed time delays in (4) T_u and T_y identically equal to two. Thus, the nominal model with fixed time delays is given by

$$\begin{bmatrix} x_1^*(k+1) \\ x_2^*(k+1) \end{bmatrix}=\begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix}\begin{bmatrix} x_1^*(k) \\ x_2^*(k) \end{bmatrix}+\begin{bmatrix} 0.5T_s^2 \\ T_s \end{bmatrix}u(k-2),$$
$$y(k)=\begin{bmatrix} 1 & 0 \end{bmatrix}\begin{bmatrix} x_1^*(k) \\ x_2^*(k) \end{bmatrix}, \tag{52}$$
$$\zeta(k)=y(k-2).$$

Our control objective is to maintain zero angle, i.e., $r=0$ under given initial state condition which is not zero. Based on the design guideline of Section 3, we select the nominal control gain $K=[5.2, 1.75, 2.1, 0.9]$. We then design a neural optimizer for the nonlinear system model with time-varying delays.

Case 1: First, we consider deterministic observations, i.e., $\omega(k)=0$ in (51). Since a control gain K is a 1×4 matrix, four neural networks are constructed in (27) along with each input vector

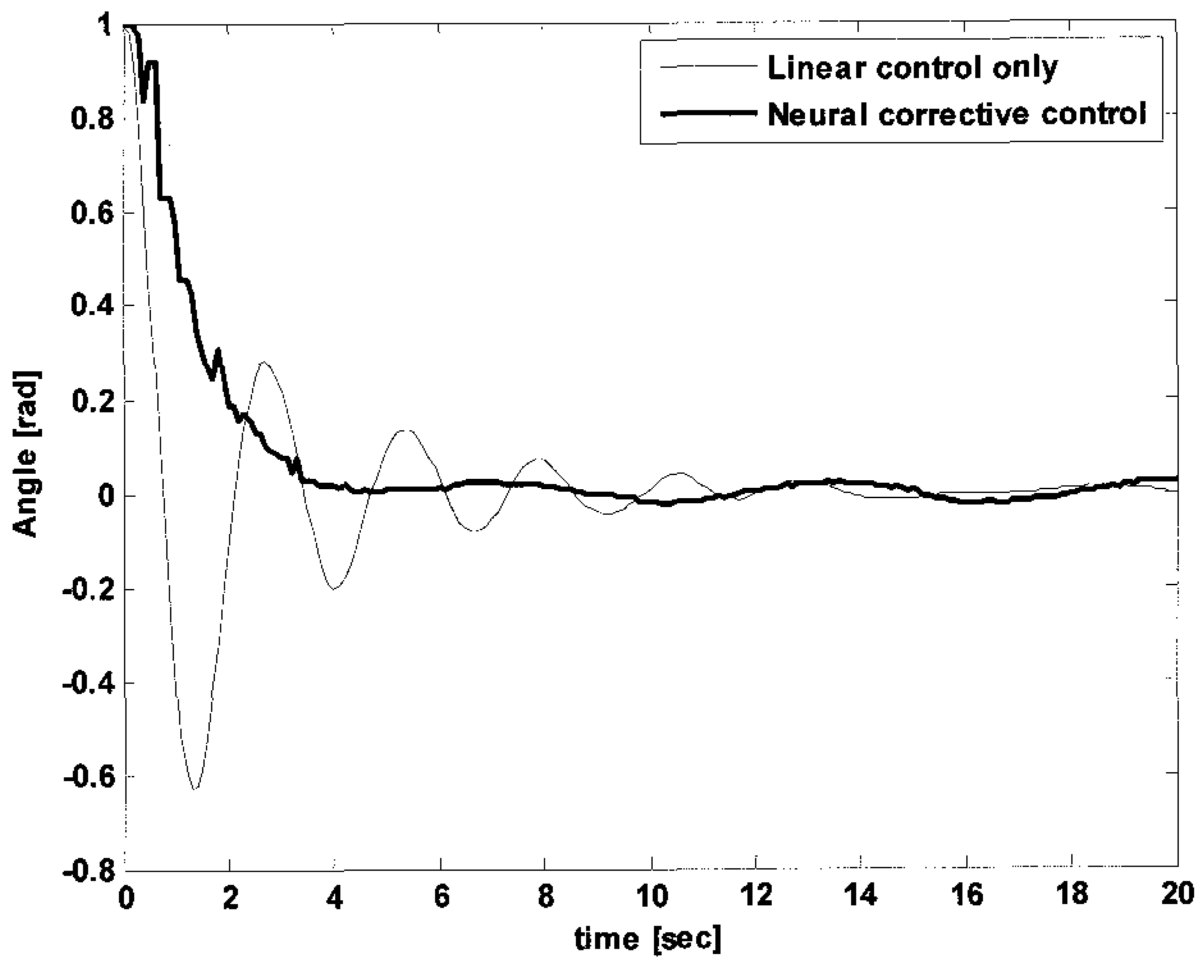
$$[E(k)\ \Omega_i(k)]^T = [1 e(k) e(k-1) e(k-2) \tilde{\kappa}_i(k-1) \tilde{\kappa}_i(k-2)]^T, \quad i=1,\cdots,4.$$

Each initial condition for the network is randomly chosen with uniform distribution in [-0.5, 0.5] and the learning rate is 0.75. The neural network iteratively learns until it reaches satisfactory performance for varying initial values of the networks. This learning is accomplished from offline experiments in which simulation data are generated from the numerical setup in (52). As stated in Section 3.2, an objective function used in this learning procedure is linearly composed of a system error which is the deviation between a reference value and a system output. Training data of the neural networks is the system error made from comparing a system output to a reference value. We select a neural network with the smallest error function through iterative learning routes with 5000 epochs. Fig. 8(a) shows angle trajectory of a satellite system with an initial vector $x(0)=[1\ 0]^T$ for added neural corrective control and for linear control only. The results clearly demonstrate that the corrective control effectively enhances control performance. For linear control we observe large oscillations that corrective control successfully eliminates. Thus, settling time in case of neural control is much faster than linear control. A small ripple in the steady-state region of the neural control is present due to the nonlinearity of the system model. Likewise, the control input in Fig. 8(b) is clearly more efficient for neural control. Initially, linear control uses excessive input force which causes large oscillations, whereas the corrective neural control reduces the average control power.

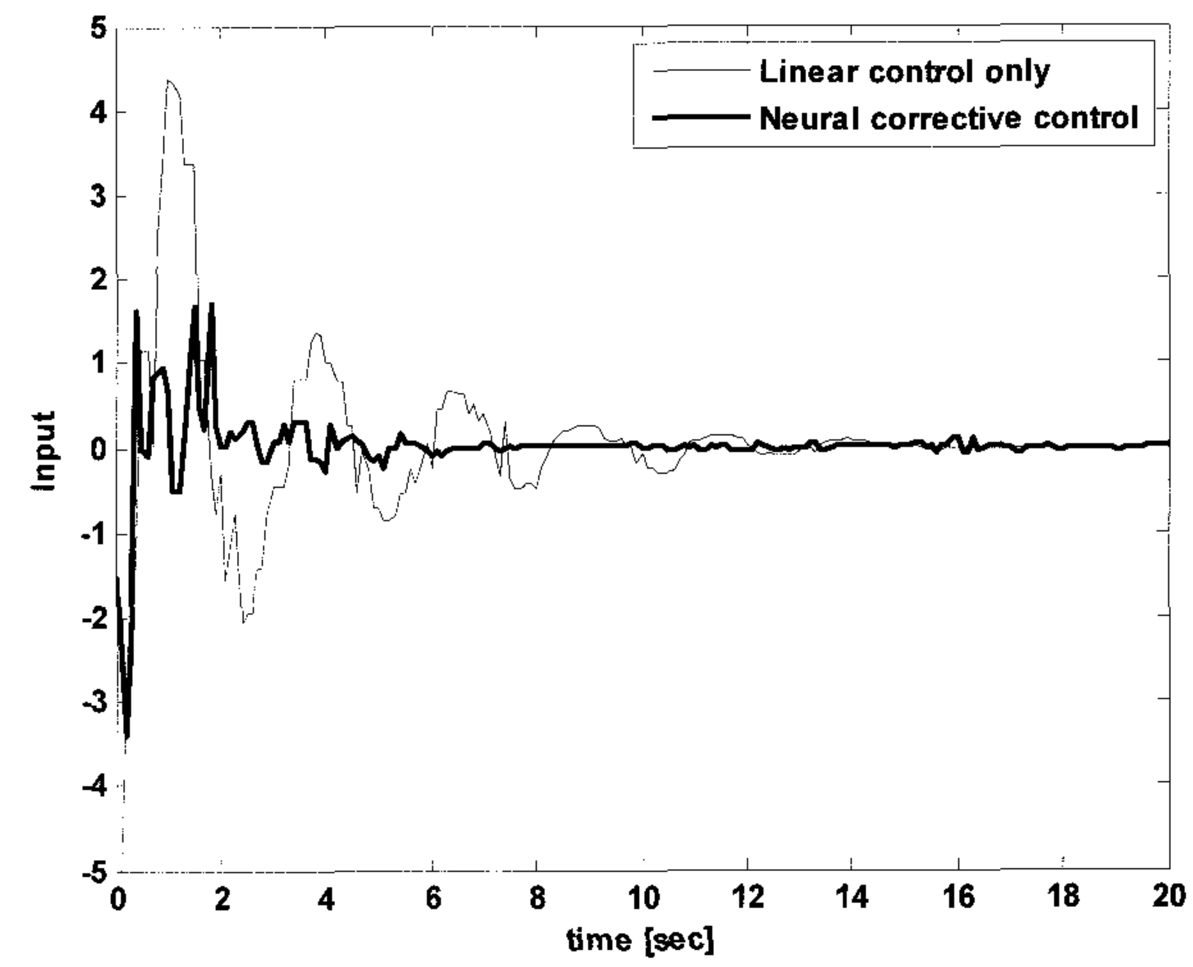
Case 2: Next, we construct a neural predictive control using the nonlinear stochastic model of (51). For DBN modeling of the random observation, we first quantize the continuous observation signal as

$$\begin{aligned}
\zeta_1(k) &= \{ |\zeta(k)| > 0.2) \}, \\
\zeta_2(k) &= \{ |\zeta(k)| \in [0.2, 0.4) \}, \\
\zeta_3(k) &= \{ |\zeta(k)| \in [0.4, 0.6) \}, \\
\zeta_4(k) &= \{ |\zeta(k)| \in [0.6, 0.8) \}, \\
\zeta_5(k) &= \{ |\zeta(k)| \geq 0.8 \}.
\end{aligned} \tag{53}$$

In the output equation of (51), $c_3(k)$ is uniformly distributed random variable and $\omega(k)$ is Gaussian zero-mean with randomly changed variance. We test our predictive control for the system model and compare it with the non-predictive neural control design in Case 1 with the same simulation environment. Simulation results for these two controls are illustrated in Fig. 9. There are large deviations in non-predictive control due to non-stationary random noise in the output model, but the output trajectory for predictive control is approximately zero in the steady state. These results demonstrate the success of our neural predictive control. The corresponding input depicted in Fig. 9(b) shows that neural predictive control energy is significantly smaller than that for non-predictive control.

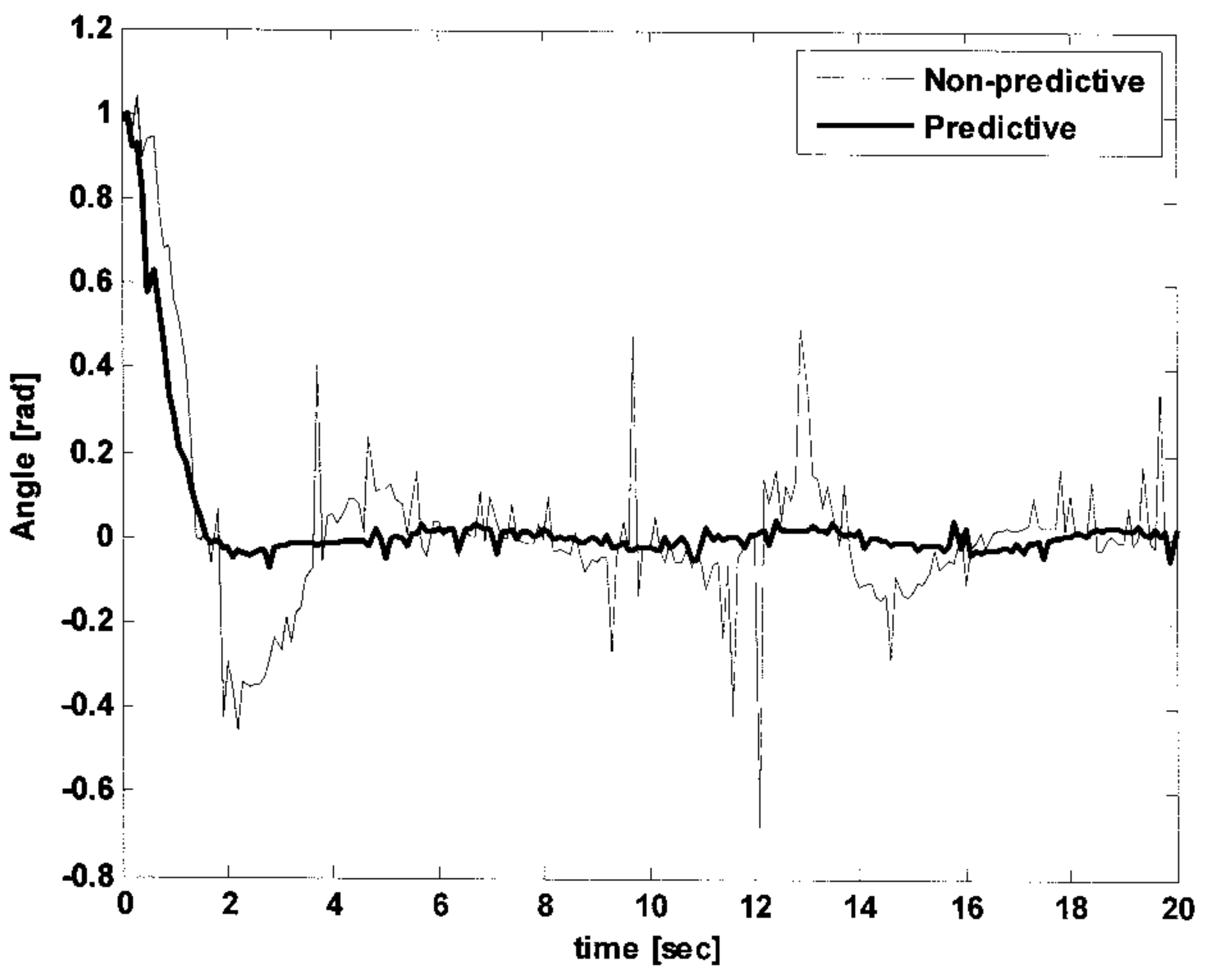


(a) System response.

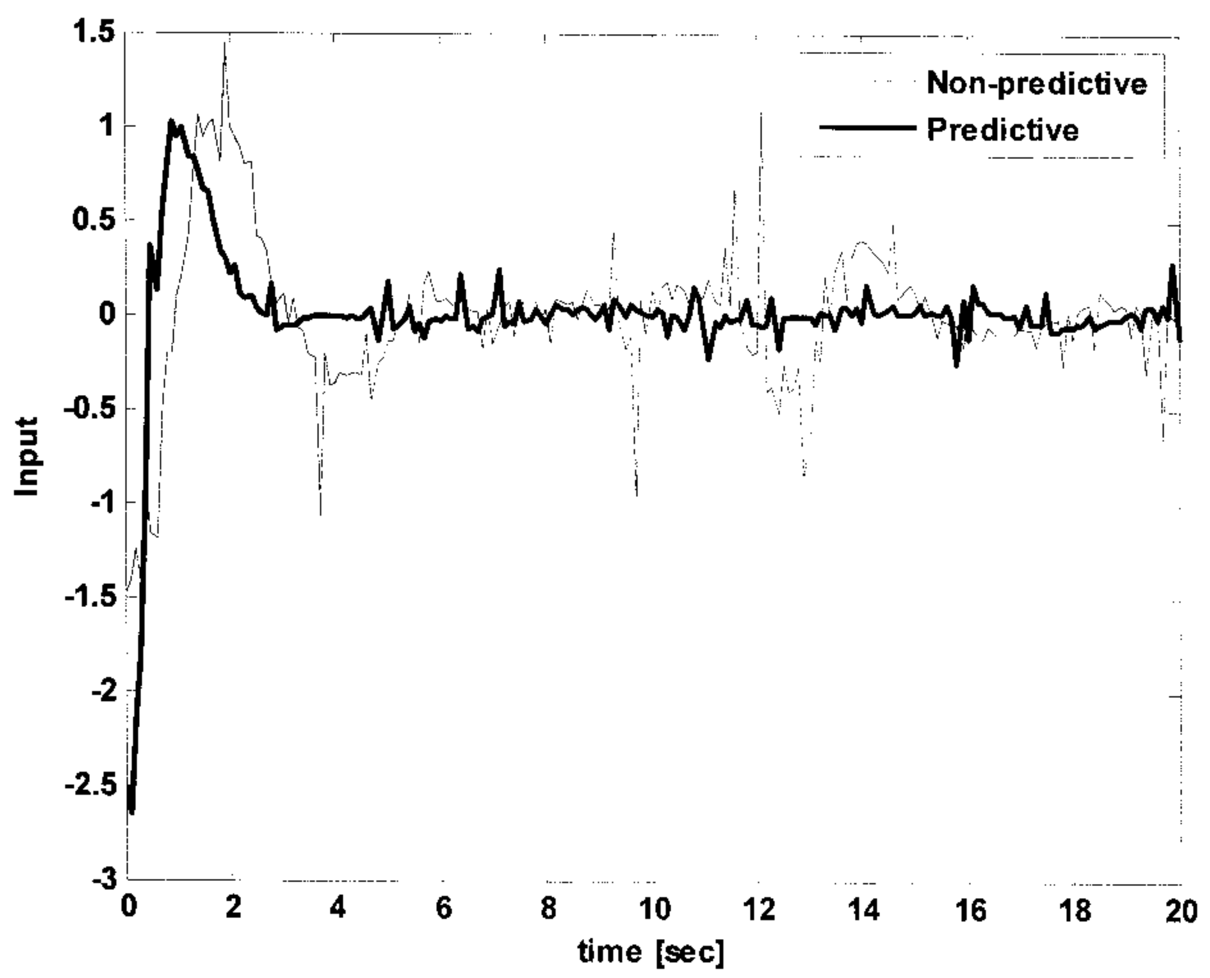


(b) Control input.

Fig. 8. Simulation result of Case 1.



(a) System response.



(b) Control input.

Fig. 9. Simulation result of Case 2.

7. CONCLUSION

We propose a neural control approach for nonlinear stochastic networked systems with time-varying delay. A recurrent neural module is constructed as a gain optimizer for state feedback control. We demonstrate the reliable performance of our control approach by applying it to a nonlinear satellite attitude system through simulation. Simulation results demonstrate that neural corrected control is better than just linear control and neural predictive control outperforms neural corrected control. However, adaptive control requires more data to allow DBN modeling and in some applications corrective neural control may be preferable. In future work, adaptive sampling for stabilization and more complex NCS structures will be examined.

REFERENCES

[1] T. C. Yang, "Networked control systems: A brief survey," *IEE Proc.-Control Theory and Applications.*, vol. 153, no. 4, pp. 403-412, 2006.

[2] Y. Jianyong, Y. Shimin, and W. Haiqing, "Survey on the performance analysis of networked control systems," *Proc. of IEEE Int. Conf. on System, Man and Cybernetics*, pp. 5068-5073, 2004.

[3] Y. Tipsuwan and M.-Y. Chow, "Control methodologies in networked control systems," *Control Engineering Practice*, vol. 11, pp. 1099-1111, 2003.

[4] Y. Halevi and A. Ray, "Integrated communication and control systems: Part I - Analysis," *J. of Dynamic Systems, Measurement and Control*, vol. 110, pp. 367-373, 1988.

[5] L.-W. Liou and A. Ray, "Integrated communication and control systems: Part III - Nonidentical sensor and controller sampling," *J. of Dynamic Systems, Measurement and Control*, vol. 112, pp. 357-364, 1990.

[6] R. Luck and A. Ray, "Experimental verification of a delay compensation algorithm for integrated communication and control systems," *Int. J. of Control*, vol. 59, no. 6, pp. 1357-1372, 1994.

[7] H. Chan and U. Ozguner, "Closed-loop control of systems over a communications network with queues," *Int. J. of Control*, vol. 62, no. 3, pp. 493-510, 1995.

[8] J. Nilsson, *Real-time Control Systems with Delays*, Ph.D. Dissertation, Lund Institute of Technology, 1998.

[9] G. C. Walsh, O. Beldiman, and L. Bushnell, "Error encoding algorithms for networked control systems," *Proc. of the IEEE Conf. on Decision and Control*, vol. 5, pp. 4933-4938, 1996.

[10] F. Goktas, *Distributed Control of Systems over Communication Networks*, Ph.D. Dissertation, University of Pennsylvania, 2000.

[11] S. H. Hong, "Scheduling algorithm of data sampling times in the integrated communication and control systems," *IEEE Trans. on Control Systems Technology*, vol. 3, no. 2, pp. 225-230, 1995.

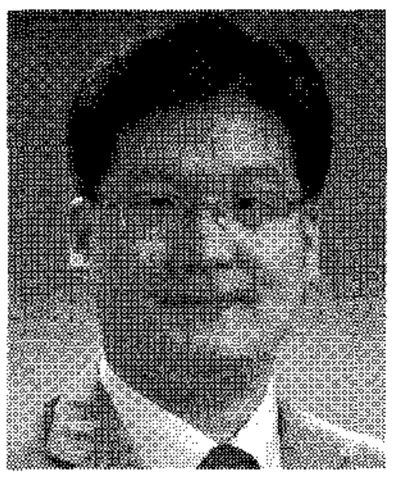
[12] S. H. Hong and W.-H. Kim, "Bandwidth allocation scheme in CAN protocol," *IEE Proc.-Control Theory and Applications*, vol. 147, no. 1, pp. 37-44, 2000.

[13] N. B. Almutairi, M.Y. Chow, and Y. Tipsuwan, "Network-based controlled DC motor with fuzzy compensation," *Proc. of the 27th Annual Conf. of the IEEE Industrial Electronics Society*, vol. 3, pp. 1844-1849, 2001.

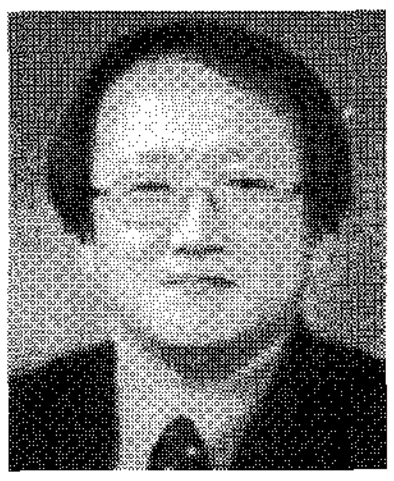
[14] G. C. Walsh, O. Beldiman, and L. Bushnell, "Asymptotic behavior of nonlinear networked control systems," *IEEE Trans. on Automatic Control*, vol. 46, no. 7, pp. 1093-1097, 2001.

[15] L. A. Montestruque and P. J. Antsaklis, "On the

model-based control of networked systems," *Automatica*, vol. 39, pp. 1837-1843, 2003.
[16] D. Yue, Q.-L. Han, and C. Peng, "State feedback controller design of networked control systems," *IEEE Trans. on Circuits and Systems*, vol. 51, no. 11, pp. 640-644, 2004.
[17] L. Zhang, Y. Shi, T. Chen, and B. Huang, "A new method for stabilization of networked control systems with random delays," *IEEE Trans. on Automatic Control*, vol. 50, no. 8, pp. 1177-1181, 2005.
[18] S. Haykin, *Neural Networks*, Prentice Hall, Upper Saddle River, NJ, 1999.
[19] M. Saerens and A. Soquet, "Neural controller based on back-propagation algorithm," *IEE Proceedings - F*, vol. 138, no. 1, pp. 55-62, 1991.
[20] K. Murphy, *Dynamic Bayesian Networks: Representation, Inference and Learning*, Ph.D. Dissertation, University of California-Berkeley, 2002.
[21] H. C. Cho and M. Sami Fadali, "Online estimation of dynamic Bayesian network parameter," *Proc. of IEEE World Congress on Computational Intelligence*, pp. 6395-6402, 2006.
[22] G. F. Franklin, J. D. Powell, and M. L. Workman, *Digital Control of Dynamic Systems*, Prentice Hall, 1997.



Hyun Cheol Cho received the B.S. from the Pukyong National University in 1997, the M.S. from the Dong-A University, Korea in 1999, and his Ph.D. from the University of Nevada-Reno, USA in 2006. He is currently a Post-doc Researcher in the Dept. of Electrical Engineering, Dong-A University. His research interests are in the areas of control systems, neural networks, stochastic process, and embedded systems.



Kwon Soon Lee received the B.S. degree from Chungnam National University, Daejeon in 1977, the M.S. degree in Electrical Engineering from Seoul National University, Seoul, Korea, in 1981, and the Ph.D. degree in Electrical and Computer Engineering from Oregon State University, USA, in 1990. He joined the Department of Electrical Engineering, Dong-A University, Busan, Korea, as an Assistant Professor from 1982 to 1994. Since October 1, 1994, he has been with the Division of Electrical, Electronic, and Computers at Dong-A University, Busan, Korea, where he is currently a Professor. He has authored or coauthored over 130 articles in archival journals and conference proceedings. His research interests include all aspects of port automation systems, intelligent the control theory, and application of immune algorithm, etc. Prof. Lee is a member of the National Research Laboratory nominated by the Korean Ministry of Science & Technology, Team Leader of the New University for Regional Innovation (NURI) at Dong-A University, and Director of the International Cooperation Department of the Regional Research Center (RRC) in Korea.