Design of Nonlinear Compensator for Positioning Control in a Mass Storage System*

(A Neural-Network-Based Controller)

Kazuhiko TAKAHASHI**, Makoto TAKAYANAGI***,
Ichiro YAMADA**** and Steffen TORP*****

Mass storage systems (MSSs) with automatic mediahandling mechanisms have become widely used in the field of information processing, and faster media handling is desired. Although there are various handling mechanisms, in all cases, disturbances such as solid friction and inertia variations are major obstacles to faster handling. This paper investigates a neural-network-based controller that compensates for these disturbances. A direct-type neural controller is proposed. The capability and characteristics of the controller are numerically and experimentally investigated by testing the positioning control of a rotary storehouse mechanism in an MSS. Experimental results, compared with those obtained using a conventional disturbance observer, indicate that the proposed neural-network-based controller can compensate effectively for nonlinear disturbances in the mechanisms.

Key Words: Neural Network, Disturbance Observer, Mass Storage System, Solid Friction, Inertia Variation

1. Introduction

The amount of information handled by information processing systems is increasing rapidly with progress towards the information society, creating an increasing need for mass storage systems (MSSs) that can accumulate large amounts of data and automatically exchange recording media such as magnetic tapes or optical disks. At the same time, users demand faster access, and thus a shorter handling time. Although there are various handling mechanisms, all of them are subjected to system disturbances such as

solid friction and inertia variations, which prevent faster handling.

To compensate for disturbances such as solid friction, researchers have examined systems that are less sensitive to disturbances⁽¹⁾, model reference adaptive control⁽²⁾, disturbance observers⁽³⁾, and neural networks⁽⁴⁾. The authors earlier proposed a disturbance observer that models the nonlinear characteristics of solid friction⁽³⁾, and have shown its effectiveness in positioning control for a media-handling mechanism. However, the neural network is expected to offer flexibility, learning ability, and nonlinear mapping ability. It can therefore be used to design a controller for an uncertain system, which involves unknown factors or is subject to nonlinear disturbances. Therefore, many studies have applied neural networks to control systems⁽⁴⁾⁻⁽¹²⁾ such as for a robot manipulator. However, most of these studies have only used computer simulation, and there are few examples of applying the neural-network-based controller (hereafter called a neural controller) to actual mechanisms such as media-handling mechanisms.

This paper presents a method of using a neural

* Received 13th December, 1993.

** NTT Interdisciplinary Research Laboratories, Nippon Telegraph and Telephone Corporation, 3-9-11 Midori-cho, Musashino, Tokyo 180, Japan

*** NTT Tokai Telecommunication Systems Department, Nippon Telegraph and Telephone Corporation, 13-3, Higashisakura 1, Higashi-ku, Nagoya 461, Japan

**** Technology Research Department, Nippon Telegraph and Telephone Corporation, 1-6 Uchisaiwai-cho 1-Chome, Chiyoda-ku, Tokyo 100-19, Japan

***** Danish Institute of Technology, Gregersensvej, P.O. Box 141, DK-2630 Taastrup, Denmark

controller to compensate for nonlinear disturbances in a rotary storehouse mechanism in an optical mass storage system (OMSS)⁽¹³⁾. In section 2, a neural controller design method using a state-space representation of the target system is proposed and its stability conditions are derived. The proposed neural controller is tested by simulation in section 3. In section 4, use of the neural controller to compensate for both solid friction and inertia variations is examined for application to the positioning control of a rotary storehouse mechanism. The experimental results using the neural controller are compared with those obtained using a disturbance observer.

2. Neural Controller for a Rotary Storehouse

2.1 State equation of a rotary storehouse

For designing the neural controller, we consider the model of a rotary storehouse mechanism shown in Fig. 1. The equation of motion of the rotary storehouse considering solid friction is expressed as⁽³⁾

$$J_H \ddot{\theta}(t) = K_T i_a(t) + T_f(t). \quad (1)$$

θ : angular displacement, i_a : electric current

J_H : inertia moment converted at motor shaft

K_T : motor torque constant, T_f : solid friction

Assuming zero friction, we obtain the discrete-time state-space equations :

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{b}u(k), \quad (2)$$

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k), \quad (3)$$

where

$$\mathbf{x}(k) = [\theta(k) \ \omega(k)]^T, \quad u(k) = i_a(k),$$

$$\mathbf{A} = \begin{bmatrix} 1 & \Delta \\ 0 & 1 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} K_T \Delta^2 / 2J_H \\ K_T \Delta / J_H \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$

ω is the angular velocity, and Δ is the sampling time.

Fig. 1 Overview of optical mass storage system with the storage capacity of 250 GB

2.1 Neural controller design

Recently, several types of controllers using neural networks have been proposed. In this section, a simple direct neural controller that determines the inverse dynamics of the target system after learning is designed using the discrete-time state-space equations, Eqs. (2) and (3).

To design the direct neural controller, we used a three-layer PDP-type neural network, which has neither an inner feedback loop nor a direct connection from the input layer to the output layer. Also, all the thresholds were assumed to be zero. To allow for the nonlinear effect, a nonlinear function f , such as a sigmoid function, was added to the designed neural network as the neuron activation function. The output of the neural network, which is used as a control input $u(k)$, can be calculated using the equation

$$u(k) = f\{W^{oh} f[W^{hi} \mathbf{I}_{NN}(k)]\}, \quad (4)$$

where W^{hi} is the weight matrix ($q \times s$) from the input layer to the hidden layer, W^{oh} is the weight matrix ($1 \times q$) from the hidden layer to the output layer, and $\mathbf{I}_{NN}(k)$ is the neural-network input vector ($s \times 1$). The learning of the neural network (i.e., the tuning of the weights between neurons) is performed according to the generalized δ -rule⁽¹⁴⁾, which is a gradient descent method that minimizes the cost function. Neural controllers can be classified into two types^{(11),(12)} according to the learning scheme. One performs the learning after every sample. As this type is in many ways similar to adaptive control, we refer to it as an adaptive-type neural controller. The other type performs the learning only after every trial. As this type is analogous to learning control, we call it a learning-type neural controller.

In an adaptive-type neural controller, the weights are adjusted after every sample so as to minimize the cost function J_a :

$$\mathbf{W}(k+1) = \mathbf{W}(k-1) - \eta \partial J_a(k) / \partial \mathbf{W}(k-1), \quad (5)$$

$$J_a(k) = (1/2) \{ \mathbf{e}^T(k) \mathbf{Q} \mathbf{e}(k) + R u^2(k-1) \}, \quad (6)$$

where $\mathbf{W}$ is one of the weight matrices, W^{hi} or W^{oh} , η is a convergence factor that determines the learning speed of the network, $\mathbf{Q}$ is the weight of the output error vector $\mathbf{e}(k)$ ($= \mathbf{y}_d(k) - \mathbf{y}(k)$, where $\mathbf{y}_d(k)$ is the desired system output) and R is the weight of the control input $u(k)$. The output error vector $\mathbf{e}(k)$ can be rewritten in the following way using Eqs. (2) and (3).

$$\mathbf{e}(k+1) = \mathbf{a}\mathbf{I}(k) - \mathbf{C}\mathbf{b}u(k) \quad (7)$$

Here,

$$\mathbf{a} = [\mathbf{E} \quad -\mathbf{C}\mathbf{A}^2 \quad -\mathbf{C}\mathbf{A}\mathbf{b} \quad -\mathbf{D}], \quad (8)$$

$$\mathbf{I}(k) = [\mathbf{y}_d(k+1) \quad \mathbf{x}(k-1) \quad u(k-1) \quad r]^T, \quad (9)$$

$\mathbf{E}$ is a unit matrix, $\mathbf{D}$ is a disturbance, and r is an arbitrary constant. Assuming that the vector $\mathbf{I}(k)$ is used as the neural network input vector $\mathbf{I}_{NN}(k) = \mathbf{I}(k)$,

$$+ \mathbf{I}_{NN}(k) \mathbf{I}_{NN}^T(k) \mathbf{W}^{hiT}(k) \mathbf{W}^{hi}(k) \} \quad (28)$$

Substituting Eq. (27) into Eq. (16) yields

$$\boldsymbol{\sigma}_b(i+1) = \boldsymbol{\sigma}_b(i) \{ \mathbf{E} - \eta \chi \boldsymbol{\xi}_b(i) \} \quad (29)$$

We use the following non-negative definite function as a candidate for the Lyapunov function.

$$V(i) = \boldsymbol{\sigma}_b(i) \boldsymbol{\Phi} \boldsymbol{\sigma}_b^T(i) \geq 0. \quad (30)$$

The stability condition of the parameter error vector $\boldsymbol{\sigma}_b(k)$ is

$$\Delta V(i) = -\boldsymbol{\sigma}_b(i) \boldsymbol{\Xi}_b(i) \boldsymbol{\sigma}_b^T(i) \leq 0, \quad (31)$$

$$\boldsymbol{\Xi}_b(i) = \eta \chi \{ \boldsymbol{\xi}_b(i) \boldsymbol{\Phi} + \boldsymbol{\Phi} \boldsymbol{\xi}_b^T(i) - \eta \chi \boldsymbol{\xi}_b(i) \boldsymbol{\Phi} \boldsymbol{\xi}_b^T(i) \} \geq 0. \quad (32)$$

In the same manner as described in section 2.3.1, if $\boldsymbol{\xi}_b(i)$ is a real symmetric matrix, the stability condition is

Table 1 Physical parameters

Inertia moment converted at motor shaft: J_H	0.0482 to 0.1155 kgm ²
Torque constant of motor: K_T	0.1911 Nm/A
Magnitude of solid friction: T_{fo}	0.1911 to 0.3822 Nm

Fig. 3 Simulation results for adaptive-type neural controller (linear NN, $\eta = 4.2 \times 10^{-3}$, $Q = \text{diag}(1000, 1)$, $R = 0$)

$$0 < \eta \chi \leq 2\lambda_{ob}^{-1}(i), \quad (33)$$

where $\lambda_{ob}(i)$ is the maximum eigenvalue of the matrix $\boldsymbol{\xi}_b(i)$. Consequently, if η satisfies Eq. (33), the stability of the parameter error vector $\boldsymbol{\sigma}_b(i)$ is guaranteed in the learning-type neural controller.

3. Simulation Study of Neural Controller

In this section, the neural controller is simulated using the mathematical model of the rotary storehouse, Eq. (1), to verify the controller's feasibility and discuss its characteristics. The nonlinear characteristics of solid friction T_f are approximately formulated by introducing the sign function of the angular velocity $\omega^{(3)}$. The physical parameters of the rotary storehouse are shown in Table 1. These parameters are measured from the experimental set-up, which is described in section 4. In the simulation, the sampling time is 10 ms.

Figure 3 shows the learning process of the adaptive-type neural controller where both the target system and the neural network are linear ($T_f = 0$, $f(x) = x$). Figure 3(a) shows the angular position, Fig. 3(b) shows tracking error, Fig. 3(c) shows the cost function $J_a(k)$, and Fig. 3(d) is the maximum eigenvalue $\lambda_{oa}(k)$. Here the convergence factor is $\eta = 4.2 \times 10^{-3}$. The system output y converges to the desired output y_d . The cost function $J_a(k)$ also converges as learning progresses. These results show the feasibility of the adaptive-type neural controller. As shown in the bottom curve, the maximum value of $\lambda_{oa}(k)$ is 2.64×10^5 at the beginning of the learning and varies as the learning progresses. As here the parameter χ equals 1.61×10^{-3} , if we choose a convergence factor η in the interval $0 < \eta \leq 4.71 \times 10^{-3}$, the local stability condition, Eq. (26), is satisfied.

Figure 4 shows the learning process of the learning-type neural controller, where both the target system and the neural network are linear. Figure 4(a) shows the relationship between either the cost function $J_b(i)$ or the maximum eigenvalue $\lambda_{ob}(i)$ and the iteration number; Fig. 4(b) shows the angular position and tracking error of both the first iteration and the 20th iteration. Here the convergence factor is $\eta = 2.0 \times 10^{-5}$. As shown in Fig. 4, the cost function $J_b(i)$ converges and the system output converges to the desired output as the learning progresses. These results show the feasibility of the learning-type neural controller. As shown in the top figure, the maximum value of $\lambda_{ob}(i)$ is 4.93×10^7 at the beginning of the learning. Therefore, if we choose a convergence factor η in the interval $0 < \eta \leq 2.52 \times 10^{-5}$, the local stability condition, Eq. (33), is satisfied.

To clarify the compensation effects of the neural controller with respect to the nonlinear characteris-

tics of the target system, simulation using a nonlinear neuron activation function is performed for a nonlinear target system. Here, the magnitude of solid friction is assumed to be 0.3 Nm in Eq. (1). In the neural controller, the neuron activation function is Eq.

Fig. 4 Simulation results for learning-type neural controller (linear NN, $\eta = 2.2 \times 10^{-5}$, $Q = \text{diag}(1000, 1)$, $R = 0$)

Fig. 5 Effect of nonlinear function for controlling nonlinear system

(16). Figure 5 shows the effect of the nonlinear function. The horizontal axis of Fig. 5 is the parameter X_{n1} ; the vertical axis is the normalized cost function averaged for one iteration. As shown in Fig. 5, the nonlinear function can reduce the normalized cost function for both types of neural controller.

4. Experimental Study of Compensation for Nonlinear Disturbances

In this section, we describe experiments that investigate the feasibility of the neural controller in an actual system. Figure 6 shows the configuration of the experimental equipment. A DC servomotor drives the rotary storehouse, whose inertia varies widely with the number of optical disk cartridges stored. The position of the rotary storehouse is detected by an optical encoder. However, there is no tachogenerator in the experimental equipment, so the velocity signal was calculated. The neural controller is implemented by software on an HP workstation. The sampling time is 10 ms. In the experiment, the desired velocity curve, ω_d , is a trapezoidal velocity curve whose maximum velocity is $2\pi/3$ rad/s and the desired position curve, θ_d , is its integral.

4.1 Experimental results with disturbance observer

As a reference for comparing the results of the neural controller, we designed a tracking P-controller with feedforward compensation using a disturbance observer⁽³⁾. The disturbance observer shown in Fig. 7 can estimate both the angular velocity, ω , and the solid friction, T_f , simultaneously by detecting the angular position, θ . In this case, the observer is a third-order system whose three gains can be set by using the Butterworth pole placement to choose the poles of the observer. The feedback gains of the P-controller are designed for a bandwidth of 7 Hz. A

Fig. 6 Configuration of experimental equipment

bang-bang feedforward current is also applied. As this controller is only used in the tracking control of the rotary storehouse positioning control, the control mode changes to a precise positioning control with a conventional LQI controller when the position error is less than $5.5 \times 10^{-4} \pi$ rad.

Figure 8 shows the experimental results using the tracking controller with the disturbance observer for the positioning control of the rotary storehouse. The tracking of both position and velocity is accurate with feed-forward compensation.

4.2 Experimental results with neural controller

Figure 9 shows the relationship between the

Fig. 7 Block diagram of disturbance observer

Fig. 8 Experimental results for disturbance observer

cost function $J_b(i)$ and the iteration number; Fig. 10 shows the response of the system controlled by the learning-type neural controller under a control input weight of $R=0$. The cost function $J_b(i)$ for each η converges and the convergence becomes faster if we increase η . These results show that the learning-type neural controller can control an actual system without a priori information on the system. The effect of using a control input weight R is to reduce the high-frequency components in the control input signal, but it takes more iterations before the cost function $J_b(i)$ converges. However, there is still one problem. If the position of the rotary storehouse is changed after the neural network has been trained, the controller will fail to compensate for the solid friction and so a

Fig. 9 Convergence of cost function

Fig. 10 Experimental results for learning-type neural controller

positioning error remains. This is because the solid friction in the rotary storehouse has a weak dependence on the storehouse position. Therefore the learning-type neural controller must be trained again if the storehouse is moved.

To solve the above problem, the adaptive-type neural controller was investigated, since it performs

Fig. 11 Experimental results for adaptive-type neural controller

on-line learning. Figure 11 shows the response of the system using the adaptive-type neural controller. The system output converges to the desired output rapidly. Even if we change the storehouse position, the control performance of the adaptive-type neural controller is better than that of the learning-type neural controller. The control input weight R has the same effect as in the learning-type controller, but unfortunately a small positioning error remains at the end of the desired trajectory. Because the control input weight R reduces the increase in control input, the control input is slightly too small to compensate for solid friction.

We tested the robustness of the control system to parameter variations by changing the inertia of the rotary storehouse, without changing the parameters of each controller. Figure 12 shows the response of the system for the positioning control of the rotary storehouse having an inertia variation of 52.34%. When a disturbance observer or either neural controller is used, the inertia has a strong influence on the deceleration. On the other hand, from Fig. 12, it can be seen that the current profiles are almost the same for all the controllers. These results show that the neural controller can compensate for nonlinear disturbances as well as the disturbance observer can.

5. Conclusions

To compensate for nonlinear disturbances such as solid friction and inertia variation in the mechanism of an OMSS, we have examined both a neural controller and a disturbance observer. The effectiveness of the neural controller was experimentally verified.

First, we proposed a design method for a direct-

Fig. 12 Comparison of compensation effects for rotary storehouse having an inertia variation of 52.34% (150 optical disk cartridges)

type neural controller using a state-space form of the target system, and discussed its local stability. Next, to investigate the performance of the neural controller, it was simulated using the mathematical model of the rotary storehouse with solid friction. The simulation results showed both the feasibility and the characteristics of the proposed controller. Finally, the controller's ability to compensate for nonlinear disturbances was investigated for use in the positioning control of a rotary storehouse in an OMSS. The experimental results confirmed that the neural controller can control a practical system by using the learning ability of the neural network without precise a priori knowledge of the system, even under existing nonlinear disturbances.

Acknowledgement

The authors wish to express their thanks to Mr. Takayuki Yamada, NTT R&D Center, for his helpful advice.

References

- (1) Bai, S., Tsujisawa, T., and Tagawa, R., *Trans. of SICE*, (in Japanese), Vol. 19, No. 8, (1983), p. 636.
- (2) Tamura, K., Ichikawa, K., Kotani, K., and Amano, Y., *Trans. of SICE*, (in Japanese), Vol. 20, No. 3, (1984), p. 227.
- (3) Yamada, I., Kaneko, K., and Takayanagi, M., *J. of Robotics and Mechatronics*, Vol. 3, No. 4, (1991), p. 334.
- (4) Ozaki, T., Suzuki, T., Furuhashi, T., Okuma, S., and Uchikawa, Y., in *Proc. on 2nd Symposium on Robotics and Automation Systems*, (in Japanese), Tokyo (1990), p. 153.
- (5) Psaltis, D., Siders, A., and Yamamura, A., in *Proc. of IEEE Int. Conference on Neural Networks*, Vol. IV, (1987), p. 551.
- (6) Kawato, M., Uno, Y., Isobe, M., and Suzuki, R., in *Proc. of IEEE Int. Conference on Neural Networks*, Vol. IV, (1987), p. 573.
- (7) Narendra, K.S., and Parthasarathy, K., *IEEE Trans. on Neural Networks*, Vol. 1, No. 1, (1990), p. 4.
- (8) Fukuda, T., Shibata, T., Tokita, M., and Mitsuka, T., *Trans. of JSME*, (in Japanese), Vol. 57, No. 535, C (1991), p. 228.
- (9) Nagato H., and Yoshida, K., *Trans. of JSME*, (in Japanese), Vol. 58, No. 545, C (1992), p. 171.
- (10) Chen, F.C., and Khalil, H.K., *Int. J. Control*, Vol. 55, No. 6, (1992), p. 1299.
- (11) Yamada T., and Yabuta, T., in *Proc. of 16th Annual Conference of the IEEE Industrial Electronics Society*, California (1990), p. 1244.
- (12) Yamada, T., Yabuta, T., and Takahashi, K., in *Proc. of Int. Conference on Industrial Electronics, Control and Instrument*, Kobe (1991), p. 1389.
- (13) Yamada, I., Saito, M., Watabe, A., and Itao, K., *IEEE Trans. on Magnetics*, Vol. 29, No. 3, (1993), p. 2172.
- (14) Rumelhart, D.E., McClelland, J.L., and PDP Research Group, *Parallel Distributed Processing*, (1989), p. 322, The MIT Press.