Neuro-Controller Design for the Line of Sight Stabilization System Containing Nonlinear Friction

비선형 마찰이 존재하는 조준경 안정화 시스템의 신경망 제어기 설계

Jun Oh Jang, Byung Gyoon Jeon, Gi Joon Jeon
(장 준 오, 전 병 균, 전 기 준)

요 약 : 본 논문에서는 비선형 마찰이 존재하는 조준경 안정화 시스템에 대해서 마찰력 보상과 성능개선을 위한 신경망제어기의 설계방법을 제시한다. 제안한 신경망제어기는 비례, 적분, 진상(PI/LEAD)제어기와 신경회로망과의 병렬로 구성되며, 제어 목적은 비선형 마찰과 외란이 존재하여도 안정거울의 각속도 추적성능과 안정화 성능의 향상에 있다. 신경회로망의 입력으로 안정거울의 각속도 추적오차와 추적오차의 적분, 제어입력이 필터를 통과한 신호가 사용되며, 신경회로망은 간접학습구조에 의해 학습된다. 조준경 시스템의 비선형 마찰력인 쿨롱마찰력의 크기가 외부환경에 따라 변하는 경우와 시스템으로 외란이 인가되는 경우에 대하여도 제안한 병렬제어기는 기존의 PI/LEAD 제어기보다 추적과 안정화 성능 면에서 우수함을 컴퓨터 모의실험으로 확인한다.

Keywords: line of sight stabilization system, coulomb friction, PI/LEAD controller, neuro-controller

I. Introduction

Frictional forces in mechanical systems can be divided into two categories, linear and nonlinear friction according to relationship with velocity. Viscous friction represents a retarding force that is proportional to the velocity. However, it does not give much trouble to the design of a control system mainly due to its linear relationship with velocity. It rather makes contribution to improve system stability with desirable property such as break. On the other hand, the coulomb friction is nonlinear friction that has a constant amplitude with respect to the change in velocity, but the sign of the coulomb frictional force changes with the reversal of the direction of velocity. So, the coulomb friction, which is always present to some degree in mechanical systems, causes difficulties in the design and analysis of a control system. Therefore, the performance of a real control system cannot be accurately predicted by simulation or analysis of the system without considering the nonlinear friction. Its presence is often responsible for the inability of the system to keep low values of the steady state error.

The linearization method, which is often used as a design method of a nonlinear control system, gives rise to serious problems to a position control system with nonlinear friction because the sign of the coulomb frictional force changes abruptly in the vicinity of zero velocity. Hence, a practical method for canceling the friction in the position control system would entail on-line estimation of the friction present in the system and addition of the estimated value to the controller output. But, the degradation of the system may occur when the value of friction is not estimated properly.

To overcome this problem, an adaptive friction compensation method has been studied. The use of recursive least square algorithm for estimation of the parameters in a nonlinear friction model was considered by Canudas et al.[1]. Friedland and Park[2] presented another adaptive friction compensation scheme which was based upon a Lyapunov-like argument involving the position error. Also, many researches on friction compensation have been carried out and reported in a survey paper[3]. However, these methods are based on the characteristics of the nonlinearity and the knowledge on some of the parameters. Friction is a natural phenomenon that is quite hard to model and not yet completely understood.

Recently, advances in the area of artificial neural networks have provided the potential for new approaches to the control of nonlinear systems through learning process. In robotics, Kawato et al.[4] used a hierarchical neural network model as add-on component to the conventional linear controller in order to control the movement of a robot. Lewis et al.[5] proposed multilayer neural-net robot controller with guaranteed tracking performance. Lightbody et al.[6] proposed a direct model reference adaptive control structure using a linear controller and a neural network in parallel in a chemical process and a missile control system. However, nonlinear friction compensation method using a neural network and a linear controller has not been studied in servo systems.

In this paper, we propose a neural network control method for compensation of the friction in a line of sight stabilization system(LSS) containing nonlinear friction. The proposed neuro-controller consists of a PI/LEAD controller and a neural network controller in parallel. The objective of the neuro-controller is to compensate the nonlinear friction and improve the tracking and stabilization performance under disturbance. Tracking error and its integral are used as inputs to the neural network for control(NNC) in order to sense tracking error and to compensate the frictional force, and a filtered control input is also used as an additional input to the NNC to reduce the overshoot of the output response. We train

접수일자 : 1995. 12. 5., 수정완료 : 1997. 2. 23.

장준오, 전병균 : 경북대학교 전자 · 전기공학부
전기준 : 경북대학교 전자 · 전기공학부, 제어계측신기술연구센터

the NNC by the indirect learning control scheme[7]. We compare the tracking and stabilization performance of the proposed neuro-controller with that of the PI/LEAD controller, when the magnitude of friction is constant and changes with environmental factors. Also, we investigate the stabilization performance of the system under random, sinusoidal and constant disturbances through computer simulations.

We describe the line of sight stabilization system in the next section. In Section Ⅲ, we present the neural network control method for friction compensation and in Section Ⅳ we analyze the tracking error dynamics. We compare the performance of the proposed neuro-controller with that of the conventional PI/LEAD controller in Section Ⅴ. Finally, Section Ⅵ contains conclusions and further work.

Ⅱ. Line of sight stabilization system

The line of sight stabilization system(LSS) is a servomechanism designed to maintain the angular orientation of an object in inertial space against disturbance. Besides isolating the line of sight (LOS) of a sensor from disturbances, the stabilization system is also required to slew the LOS in faithful response to command inputs from either a human or an automatic tracker and/or to provide measurements of the LOS orientation or rate to an external system for the purpose of weapon pointing. The LSS is composed of a gyro for integrating the error signal, a compensator for generating control signals, and a dc motor for driving the gimbal with a stabilized mirror.

The LSS is functionally divided into two subsystems; the LOS stabilization and tracking subsystem. The stabilization function is as follows: If disturbances from a maneuvering vehicle are transmitted to the gimbal via the vehicle, the gyro which is attached to the gimbal senses rotation of the gimbal and the gyro output signal is conditioned and amplified by the servoelectronics. Then, the PI/LEAD controller attempts to create a motor torque equal to and opposite direction to any disturbance torque placed on the gimbal. For the tracking function, a human operator senses the position error between the LOS and the target, and the command signal generated by the operator's handle goes into the gyro to track the maneuvering target. The gyro generates an electrical signal which is proportional to angular momentum, and this signal is transmitted to the compensator via a filter and a demodulator. Finally, the compensator produces a signal to control the LOS movement. Fig. 1 shows the simplified block diagram of the LSS, where ω_{cmd} is the command signal and ω_{los} is the angular velocity of the mirror. Two low pass filters (LPF) with cut off frequencies 200Hz and 83 Hz for noise rejection are placed in front of the controller and the motor, respectively.

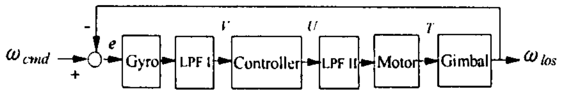


Fig. 1. Block diagram of the line of sight stabilization system.

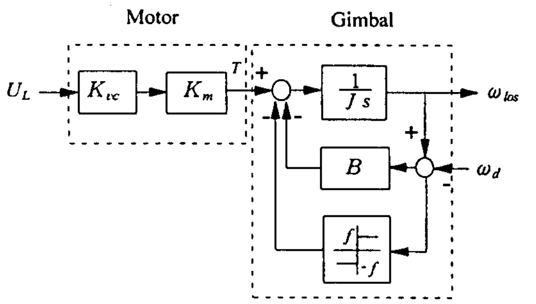


Fig. 2. Block diagram of the motor and gimbal system.

Fig. 2 shows the block diagram of the motor and gimbal system, where a brushless dc motor driven by an amplifier with current feedback is used. At the operational frequency the dc motor/actuator is approximated as the following linear relationship;

$$T = K_{vc}\, K_m\, U_L \tag{1}$$

where, T is the torque developed by the motor, K_{vc} is the gain of the current amplifier, K_m is the torque constant and U_L is the input to the motor and gimbal system. The kinematic equations of general servosystems are reported in [8] and the dynamics of the LSS are explained in [9]. The transfer function of the gimbal system with a stabilized mirror neglecting the coulomb friction can be expressed by

$$G_{gim}(s) = \frac{1}{J\,s + B} \tag{2}$$

where J is the moment of inertia and B is the viscous friction. The parameters of the motor are from the data sheet of the manufacturer and those of the gimbal system have been obtained by experimental measurements and calculations and they are given in Table 1. When the LSS mounted on a vehicle moves on off-load, the disturbance caused by mechanical vibration is fed to the gimbal system and affects the viscous and the coulomb friction. The coulomb friction is modeled as constant times the sign of velocity as shown in Fig. 2. For the LSS, if the angular velocity ω_{los} of the mirror is larger than the disturbance ω_d, the sign of the coulomb friction is positive, otherwise negative.

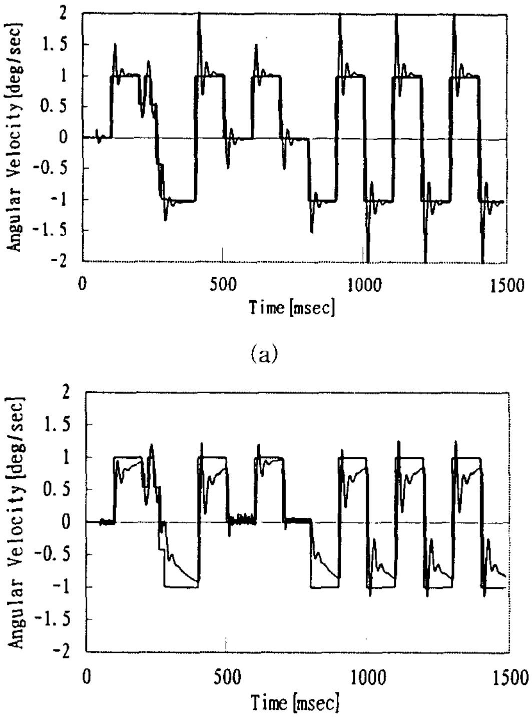
The gyro mounted directly on gimbal system serves to measure the LOS rate. The transfer function of the rate integrating gyro used in the LSS is obtained from the curve fitting to the gyro structure and data provided by the manufacturer as follows;

$$G_g(s) = \frac{9.45 \times 10^{13}}{s(s^3 + 2.5\times 10^3 s^2 + 5\times 10^6 s + 5.6\times 10^9)} \tag{3}$$

The controller should be designed to minimize tracking error and to maintain the LOS orientation under disturbances. The controller receives information from the

Table 1. The parameters of the motor and gimbal system.

Parameter	K_{vc}	K_m	J	B
Nominal value	0.5	9.25	0.1477	0.5685
Unit	A/V	IN-OZ/A	IN-OZ-SEC^2	IN-OZ-SEC



(a)

(b)

Fig. 3. Angular velocity of the LOS mirror by the PI/LEAD controller, (a) without coulomb friction and (b) with coulomb friction.

gyro and calculates control signal. The desired specifications of the LSS are to increase the bandwidth of the closed loop system and to decrease the steady state error in the unit step response. In the LSS a PI/LEAD type controller is implemented as a main controller because the LEAD compensator increases the bandwidth and the PI controller decreases the steady state error. The transfer function of the LEAD compensator, $G_l(s)$, is obtained by the Bode plot as follows:

$$G_l(s)=\frac{s+8.2}{s+4317} \tag{4}$$

The LEAD compensator provides a phase margin of 60° at the gain crossover frequency 30Hz. The integral gain and the proportional gain are adjusted by the Ziegler-Nichols method[10]. After some experimental adjustment of the PI controller by trial and error, finally the transfer function of the PI/LEAD controller, $G_c(s)$, is determined as:

$$G_c(s)=\frac{15.4\ (s+6.28)(s+8.2)}{s(s+4317)} \tag{5}$$

Fig. 3 shows the tracking performance of the command signal of the LSS without and with the coulomb friction. We see in Fig. 3(a) that the overshoot of angular velocity is about 50 percent when we neglect the coulomb friction. Fig. 3(b) shows the angular velocity of the mirror when there exists coulomb friction in the gimbal system. The magnitude ±0.2 of the coulomb friction is added. We conjecture that the torque developed by the motor is not properly applied to the gimbal system due to the frictional force. Also, Fig. 3(b) shows chattering phenomena in the vicinity of zero velocity while maintaining zero angular velocity required by the command signal.

Ⅲ. Neuro-controller design for the line of sight stabilization system

In this section a newly developed neural network control method for friction compensation is presented. Fig. 4 shows the proposed neuro-controller which has the conventional PI/LEAD controller for system stabilization and the NNC for friction compensation in parallel. Control input U to the plant (the motor and gimbal system with LPF) is defined as:

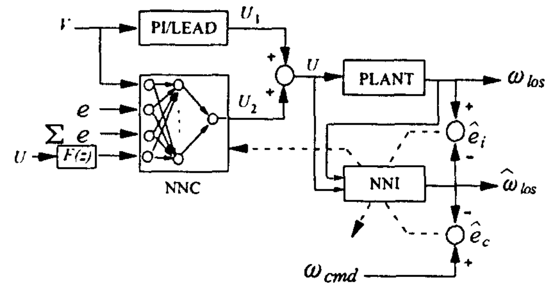


Fig. 4. Structure and learning scheme of the neuro-control system.

$$U=U_1+U_2 \tag{6}$$

where U_1 is the output of the PI/LEAD controller and U_2 is the output of the NNC. The objective of the neural network for identification(NNI) is to provide estimates of the inverse Jacobian of the plant for a use in the training of the NNC. The NNC and the NNI are three layer feedforward neural networks with the error back-propagation learning algorithm[11].

The angular velocity of the mirror, ω_{los}, is an important signal not only as the output of the motor-gimbal system but as a training signal of the NNI and an input signal to the NNC. However, the angular velocity is not accessible because of the specific structure of the rate integrating gyro of the LSS. Therefore, it is estimated from the output of the filter, LPFI as follows:

$$\tilde{\omega}_{los}(k)=\omega_{cmd}(k)-\tilde{e}(k) \tag{7}$$

where $\tilde{\omega}_{los}$ and $\tilde{e}$ are the estimates of ω_{los} and e, respectively. From the transfer functions of the gyro (3) and the low pass filter LPFI,

$$G_{f1}(s)=\frac{400\pi}{s+400\pi} \tag{8}$$

a difference equation on $\tilde{e}(k)$ could be obtained by the Tustin's method. Unfortunately, the gyro and filter system transformed by the Tustin's method is non-minimum phase. Therefore, to avoid divergence we reduce the model of the gyro and filter system to a simple integrator neglecting far left poles of the gyro and filter system. The Tustin's transformation of the integrator with a gain $K=1718$ gives the following difference equation on $\tilde{e}(k)$:

$$\tilde{e}(k) = \frac{2}{TK}[V(k) - V(k-1)] - \tilde{e}(k-1) \tag{9}$$

where the sampling time $T=0.001$ second is executed.

The three layer neural network for the NNI with the number of nodes from the input layer 4-10-1, which is shown in Fig. 5, has a net output given by

$$\hat{\omega}_{los} = \sum_{m=1}^{10} [w^i_{m1} \cdot \sigma^i (\sum_{l=1}^{4} v^i_{lm} \cdot x^i_l)] \tag{10}$$

with $\sigma^i(\cdot)$, the hyperbolic tangent function, v^i_{lm}, the first to second layer interconnection weights, w^i_{m1}, the second to third layer interconnection weights, and the superscript i implies NNI. The inputs to the NNI, $x^i = [x^i_1, x^i_2, x^i_3, x^i_4]^T = [U(k), U(k-1), \tilde{\omega}_{los}(k-1), \tilde{\omega}_{los}(k-2)]^T$, are composed of the control input and the plant output. The performance index for training the NNI is

$$\hat{E_i}(k) = \frac{1}{2}\tilde{e}_i(k)^2 = \frac{1}{2}[\tilde{\omega}_{los}(k) - \hat{\omega}_{los}(k)]^2 \tag{11}$$

where $\tilde{e}_i(k)$ is defined as the error between the estimated plant output and the output of the NNI. The second to third layer weights of the NNI, w^i_{m1} are adjusted by back propagating the estimated error by the following equation:

$$\begin{aligned} w^i_{m1}(k+1) &= w^i_{m1}(k) - \eta \cdot \frac{\partial \widehat{E_i}(k)}{\partial w^i_{m1}(k)} \\ &= w^i_{m1}(k) + \eta \cdot \frac{\partial \ \hat{\omega}_{los}(k)}{\partial w^i_{m1}(k)} \cdot \ \tilde{e}_i(k) \\ &= w^i_{m1}(k) + \eta \cdot \sigma^i(\sum_{l=1}^{4} v^i_{lm}(k) \cdot x^i_l(k)) \cdot \ \tilde{e}_i(k) \end{aligned} \tag{12}$$

where η is the step size. Similarly, the first to second layer weights of the NNI, v^i_{lm} are adjusted by following equation:

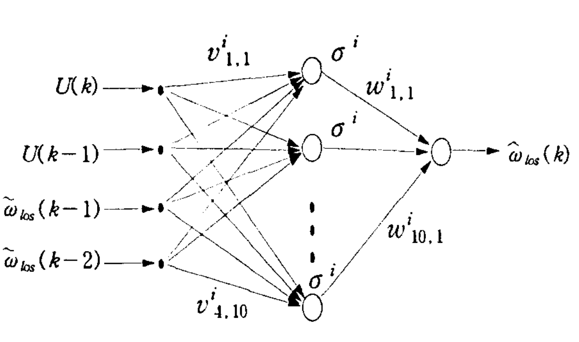


Fig. 5. Structure of the neural network for identification(NNI).

$$\begin{aligned} v^i_{lm}(k+1) &= v^i_{lm}(k) - \eta \cdot \frac{\partial \widehat{E_i}(k)}{\partial v^i_{lm}(k)} \\ &= v^i_{lm}(k) + \eta \cdot \frac{\partial \ \hat{\omega}_{los}(k)}{\partial v^i_{lm}(k)} \cdot \ \tilde{e}_i(k) \\ &= v^i_{lm}(k) + \eta \cdot x^i_l(k) \cdot \ \dot{\sigma}^i(\sum_{l=1}^{4} v^i_{lm}(k) \cdot x^i_l(k)) \\ &\cdot w^i_{m1}(k) \cdot \ \tilde{e}_i(k) \end{aligned} \tag{13}$$

where $\dot{\sigma}^i(\cdot)$ is the time derivative of the $\sigma^i(\cdot)$.

Since the structure of the NNC are the same as that of the NNI, the output of the NNC with the number of nodes from the input layer 4-10-1 is given by

$$U_2 = \sum_{i=1}^{10} [w^c_{i1} \cdot \sigma^c (\sum_{h=1}^{4} v^c_{hi} \cdot x^c_h)] \tag{14}$$

where v^c_{hi} is the first to second layer connection weights, w^c_{i1}, the second to third layer connection weights, $x^c = [x^c_1, x^c_2, x^c_3, x^c_4]^T = [V(k), e(k), \Sigma e(k), F(U(k))]^T$ the inputs to the NNC.

Friction can be rarely determined *a priori* with an accuracy required for its cancellation and cannot be readily measured or controlled. Therefore, tracking error e,command angular' velocity minus angular velocity of the mirror, and its integral are fed back to the NNC in order to sense tracking error caused by the frictional force. The control input passing through a filter $F(z)$, which is explained later, is also used as an additional input to the NNC to reduce the overshoot of the response. Since the NNC can not directly learn the nature of friction, the connection weights of the NNC in Fig. 4 are adjusted by an indirect learning control scheme[6]. The performance index for training the NNC is

$$\hat{E_c}(k) = \frac{1}{2}\hat{e}_c(k)^2 = \frac{1}{2}[\omega_{cmd}(k) - \hat{\omega}_{los}(k)]^2 \tag{15}$$

where $\hat{e}_c(k)$ is defined as the estimated tracking error between the command signal and the output of the NNI. The indirect learning control scheme of the NNC is as follows: The motor and gimbal dynamics are identified by the NNI through a learning process, during which connection weights are adjusted in a direction to minimize the sum of squared errors between the desired output, ω_{los} and neural network output, $\hat{\omega}_{los}$. On the other hand, the second to third layer weights of the NNC, w^c_{i1} , are adjusted by back propagation of the estimated tracking error through the NNI by the following equation:

$$\begin{aligned} w^c_{i1}(k+1) &= w^c_{i1}(k) - \eta \cdot \frac{\partial \widehat{E_c}(k)}{\partial w^c_{i1}(k)} \\ &= w^c_{i1}(k) + \eta \cdot (\frac{\partial U_2(k)}{\partial w^c_{i1}(k)} + \frac{\partial U_1(k)}{\partial w^c_{i1}(k)}) \cdot \frac{\partial \ \hat{\omega}_{los}(k)}{\partial U(k)} \cdot \widehat{e_c}(k) \\ &= w^c_{i1}(k) + \eta \cdot \sigma^c(\sum_{h=1}^{4} v^c_{hi}(k) \cdot x^c_h(k)) \cdot \sum_{m=1}^{10} [v^i_{1m}(k) \cdot \\ &\dot{\sigma}^i(\sum_{l=1}^{4} v^i_{lm}(k) \cdot x^i_l(k)) \cdot w^i_{m1}(k)] \cdot \ \hat{e}_c(k) \end{aligned} \tag{16}$$

where the facts that $\frac{\partial U_1(k)}{\partial w^c_{i1}(k)} = 0$ and $\frac{\partial U_2(k)}{\partial w^c_{i1}(k)} =$

$\sigma^{\cdot c}(\sum_{h=1}^{4} v^c_{hi}(k) \cdot x^c_h(k))$ from (14) are utilized.

The first to second layer weights of the NNC, v^c_{hi} are similarly adjusted by following equation:

$$\begin{aligned} v^c_{hi}(k+1) &= v^c_{hi}(k) - \eta \cdot \frac{\partial \widehat{E_c}(k)}{\partial v^c_{hi}(k)} \\ &= v^c_{hi}(k) + \eta \cdot \frac{\partial U_2(k)}{\partial v^c_{hi}(k)} \cdot \frac{\partial\ \widehat{\omega}_{los}(k)}{\partial U(k)} \cdot \widehat{e_c}(k) \\ &= v^c_{hi}(k) + \eta \cdot x^c_h(k) \cdot \dot{\sigma}^c(\sum_{h=1}^{10} v^c_{hi}(k) \cdot x^c_h(k)) \cdot w^c_{i1}(k) \cdot \\ &\sum_{m=1}^{10}[v^i_{1m}(k) \cdot \dot{\sigma}^i(\sum_{l=1}^{4} v^i_{lm}(k) \cdot x^i_l(k)) \cdot w^i_{m1}(k)] \cdot \hat{e}_c(k) \end{aligned} \tag{17}$$

The large overshoot and subsequent oscillations in Fig 3(a) are due to the excessive amount of the control input generated by the PI/LEAD controller at the instant of reference input change. In the PI/LEAD control system, the excessive amount of the control input is inevitable because of the bandwidth of the closed loop system. But large overshoot would be reduced if the output of the NNC has the opposite sign to that of the PI/LEAD controller output. The following filter output is used as an additional input to the NNC to reduce the overshoot of the control system:

$$F(z) = (1 - a\ z^{-1}) \tag{18}$$

where z^{-1} is a delay operator. When the value of a is 1, the difference of control input is fed back to the NNC via the filter. Computer simulations show that as the value of a increases from 1 the output of the NNC increases in magnitude but with opposite sign to the output of the PI/LEAD controller. Thus, we select a proper value of a to reduce the magnitude of the original control input generated by the PI/LEAD controller. Hence, the control input by the NNC combined with the control input by the PI/LEAD controller makes the mirror eventually track the command signal with smaller overshoots. Practically, we select the value of a by trial and error.

Ⅳ. Analysis on error dynamics of the neuro-control system

In this section, the system error dynamics are analyzed and a bound on the tracking error is derived. The dynamics of the plant (motor and gimbal system) may be expressed from (1) and (2) by

$$J\ \dot{\omega}_{los} + B\,\omega_{los} + h(\cdot) + \omega_d = T \tag{19}$$

where $h(\cdot)$ is a nonlinear function of the angular velocity, ω_d is bounded unknown disturbance and T is control input torque from the motor. It is assumed that $|\omega_d| < \tau_d$ with τ_d known positive constant value. Given the command signal ω_{cmd}, the tracking error is $e = \omega_{cmd} - \omega_{los}$. Differentiating tracking error and using (19), the dynamics of the plant may be written in terms of the tracking error as:

$$J\dot{e} = -Be - T + f + \omega_d \tag{20}$$

where the nonlinear plant function f is

$$f = J\ \dot{\omega}_{cmd} + B\,\omega_{cmd} + h(\cdot) \tag{21}$$

Define now a control input torque as:

$$T = \hat{f} + K_f\,e \tag{22}$$

where the feedback controller (gyro and PI/LEAD) gain, K_f, and an estimate of f, $\hat{f}$, will be provided later by some means not yet disclosed. Then, the closed loop system becomes

$$J\dot{e} = -(K_f + B)e + \hat{f} + \omega_d \tag{23}$$

where the functional estimation error $\hat{f}$ is given by

$$\hat{f} = f - \hat{f} \tag{24}$$

(23) is an error system wherein the tracking error is driven by the functional estimation error. In the remainder of the paper we shall use (23) to focus on selecting a neural network training algorithm that the neural network approximates the nonlinear plant function f.

A three layer neural network(NN), which is similar as shown in Fig. 5, has a net output given by

$$y = \sum_{i=1}^{N_2}[w_{i1} \cdot \sigma(\sum_{h=1}^{N_1} v_{hi} \cdot x_h)] \tag{25}$$

with new notations; $\sigma(\cdot)$,the activation function, v_{hi}, the first to second layer interconnection weights, w_{i1},the second to third layer interconnection weights, N_1, the number of the first layer neuron, and N_2, the number of the second layer neuron. The NN equation may be conveniently expressed in vector format by defining $x = [x_0, x_1, \cdot\cdot\cdot, x_{N_1}]^T$, $\widehat{W}^T = [w_{1,1}, w_{1,2}, \cdot\cdot\cdot, w_{1,N_2}]$, and matrix format by defining $\widehat{V}^T = [v_{hi}]$. Then,

$$y = \widehat{W}^T\sigma(\widehat{V}^T x)\ . \tag{26}$$

A general function f can be modeled by a neural network as:

$$f = W^T\sigma(V^T x) + \varepsilon \tag{27}$$

where W and V are constant ideal weights of the current weights $\widehat{W}$ and $\widehat{V}$ so that ε is bounded by a known constant ε_N, and ε is reconstruction error due to the neural network structure. For practical situation, we assume that the ideal weights are bounded by known positive values so that $\|W\| < W_M$, $\|V\| < V_M$, where $\|\cdot\|$ is a matrix norm. Define the weight deviation or weight estimation error as:

$$\widehat{W} = W - \widehat{W}\ ,\ \widehat{V} = V - \widehat{V}, \tag{28}$$

and the second layer output error for a given x as:

$$\tilde{\sigma} = \sigma - \hat{\sigma} = \sigma(V^T x) - \sigma(\widehat{V}^T x)\ . \tag{29}$$

The Taylor series expansion of the second layer output for a given x may be written as:

$$\sigma(V^T x) = \sigma(\widehat{V}^T x) + \dot{\sigma}(\widehat{V}^T x)\,\widehat{V}^T x + O(\widehat{V}^T x) \tag{30}$$

with $\dot{\sigma}(\hat{z}) \equiv d\sigma(z)/dz|_{z=\hat{z}}$ and $O(\cdot)$ denoting sum of high order terms. Denoting that $\dot{\hat{\sigma}} = \dot{\sigma}(\widehat{V}^T x)$, we have

$$\begin{aligned}\tilde{\sigma} &= \dot{\sigma}(\widehat{V}^T x)\widehat{V}^T x + O(\widehat{V}^T x) \\ &= \dot{\hat{\sigma}}\widehat{V}^T x + O(\widehat{V}^T x)\end{aligned} \tag{31}$$

Now, define the neural network functional estimate of (27) by

$$\hat{f} = \widehat{W}^T \sigma(\widehat{V}^T x) \tag{32}$$

with $\widehat{V}$, $\widehat{W}$ the current (estimated) values of the ideal weights V, W as provided by the training algorithms subsequently to be discussed. Ignoring the actuator dynamics, select a control input torque using (22) and (32) as:

$$T = K_f e + \widehat{W}^T \sigma(\widehat{V}^T x) \tag{33}$$

Using (27) and (33), the closed loop error dynamics (23) become

$$\begin{aligned}J\dot{e} = &-(K_f+B)e + W^T\sigma(V^T x) - \widehat{W}^T\sigma(\widehat{V}^T x) \\ &+ \varepsilon + \omega_d\end{aligned}. \tag{34}$$

Adding and subtracting $W^T \hat{\sigma}$ yields

$$J\dot{e} = -(K_f+B)e + \widehat{W}^T \hat{\sigma} + W^T \tilde{\sigma} + \varepsilon + \omega_d. \tag{35}$$

Adding and subtracting again $\widehat{W}^T \tilde{\sigma}$ yields

$$\begin{aligned}J\dot{e} = &-(K_f+B)e + \widehat{W}^T \hat{\sigma} + \widehat{W}^T \tilde{\sigma} + \widehat{W}^T \tilde{\sigma} \\ &+ \varepsilon + \omega_d\end{aligned}. \tag{36}$$

Using the Taylor series approximation for $\tilde{\sigma}$, the closed loop error system becomes

$$J\dot{e} = -(K_f+B)e + \widehat{W}^T \hat{\sigma} + \widehat{W}^T \dot{\hat{\sigma}} \widehat{V}^T x + d \tag{37}$$

where the disturbance terms d is

$$\begin{aligned}d &= \widehat{W}^T \dot{\hat{\sigma}} \widehat{V}^T x + W^T O(\widehat{V}^T x) + \varepsilon + \omega_d \\ &= \delta + \varepsilon + \omega_d\end{aligned}. \tag{38}$$

The higher order terms in the Taylor series, δ, is bounded by positive constant δ_N, i.e., $|\delta| < \delta_N$. It is important to note that the neural network reconstruction error ε, the plant disturbance ω_d, and the higher-order terms δ in the Taylor series expansion of f all have exactly the same influence as disturbances in the error system.

For the neural network training algorithm to improve the tracking performance of the closed loop system it is required to demonstrate that tracking error e is suitably small. A bound on the tracking error is derived by the following theorem.

Theorem 1 : Let the command signal be bounded. Take the control input for (19) as (33). Let NNC weight training algorithmbe provided by (16) and (17). Then, the tracking error e evolves within a practical bound

$$|e| \leq \frac{\delta_N + \varepsilon_N + \tau_d}{K_f + B} \tag{39}$$

Proof : Define the Lyapunov function candidate for the error dynamics (23)

$$L = \frac{1}{2}Je^2 + \frac{1}{2}(\widehat{W}^T \widehat{W}) + \frac{1}{2}tr(\widehat{V}^T \widehat{V}) \tag{40}$$

where $tr(\cdot)$ is trace. Differentiating yields

$$\dot{L} = J\dot{e}e + \frac{1}{2}\dot{J}e^2 + (\widehat{W}^T \dot{\widehat{W}}) + tr(\widehat{V}^T \dot{\widehat{V}}) \tag{41}$$

whence substitution from(37) yields

$$\begin{aligned}\dot{L} = &-(K_f+B)e^2 + \frac{1}{2}\dot{J}e^2 + \widehat{W}^T(\dot{\widehat{W}} + \hat{\sigma}e) \\ &+ tr\widehat{V}^T(\dot{\widehat{V}} + xe\widehat{W}^T \dot{\hat{\sigma}}) + ed.\end{aligned} \tag{42}$$

Since $\dot{\widehat{W}} = -\dot{\widehat{W}}$ with W constant (and similarly for $\dot{\widehat{V}}$) and if the dynamics of the plantare identified by the NNI, then $\hat{e}_c = e$. We can derive $\dot{\widehat{W}} = -\hat{\sigma}e$ and $\dot{\widehat{V}} = -x(\dot{\hat{\sigma}}^T \widehat{W} e)^T$ from the training rules (16) and (17). The training rules and the assumption $|\dot{J}| = 0$ gives

$$\dot{L} = -(K_f+B)e^2 + ed \tag{43}$$

and

$$\begin{aligned}\dot{L} \leq &-(K_f+B)|e|^2 + |e|(\delta_N + \varepsilon_N + \tau_d) = \\ &-|e|[(K_f+B)|e| - (\delta_N + \varepsilon_N + \tau_d)]\end{aligned} \tag{44}$$

Thus, $\dot{L}$ is negative as long as the term in brace is positive, which implies

$$|e| > \frac{\delta_N + \varepsilon_N + \tau_d}{K_f + B}. \tag{45}$$

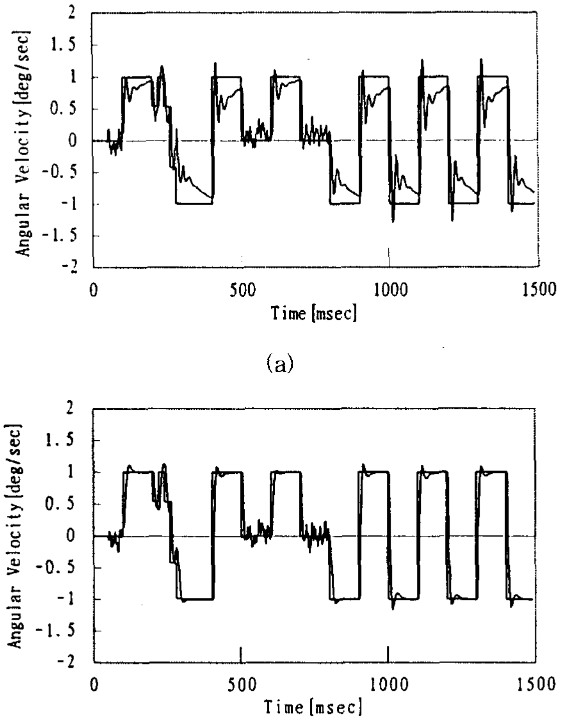
According to the standard Lyapunov theorem, the tracking error evolves within the right hand side of (45). ■

The neural network reconstruction error ε, the bounded disturbance ω_d, and the higher order Taylor series terms δ increase the bound on $|e|$. However, a small tracking error bound may be achieved by reducing δ by training the NNC and reconstruction error ε by properly selecting the structure of the NNC. Notice that since the conventional controller gain K_f is determined according to the design of the PI/LEAD controller, K_f can not be increased arbitrarily. However, large K_f may decrease the tracking error bound as long as the PI/LEAD controller maintains the stability of the control system.

V. Performance analysis of the proposed control method

In this section, the performance of the proposed control system under Gaussian, sinusoidal and constant disturbances are analyzed. Also, effects of a coulomb friction varying with time and a disturbance measured at a field test are investigated through computer simulations. All the simulations in this analysis have been executed under digital environments. Continuous transfer functions were converted to the equivalent discrete transfer functions by the Tustin's method with sampling time T=0.001 second. The response of the discrete system was very close to that of the analog system without the digital neural network controller.

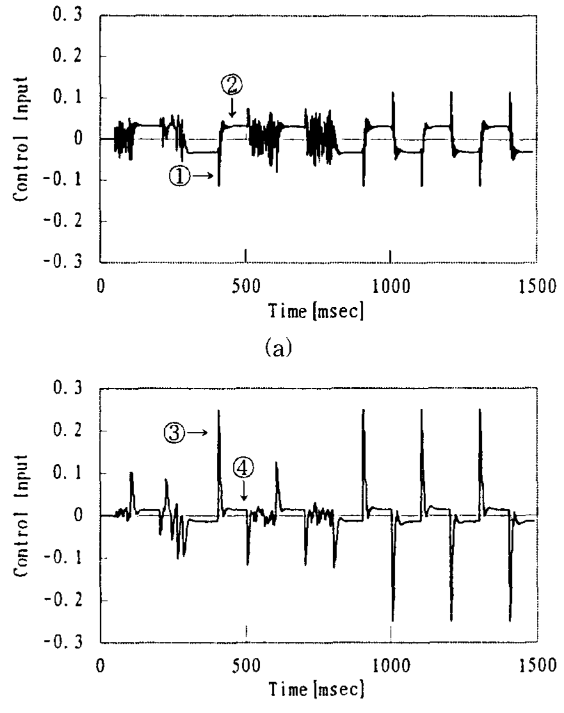
When a vehicle with the LSS mounted on it moves on bad environment such as off-load, the disturbance caused by the mechanical vibration is transmitted to the gimbal system. Fig 6(a) shows the angular velocity of the mirror controlled by the PI/LEAD controller alone when a random disturbance gets into the gimbal system. The magnitude of the random disturbance is assumed to be uniformly distributed between 0.5 and -0.5 deg/sec with variance 0.083. The NNC was trained as described in the previous section in order to minimize the tracking error. The overshoot of the angular velocity of the mirror with the neuro-controller is decreased when *a* varies from 1 to 5. At the value of 5, the angular velocity of the mirror is shown in Fig. 6(b) and the control inputs generated by the neuro-controller is shown in Fig. 7. At the instant that the command signal changes from -1 to 1 deg/sec, we see that the sign of control input signal ① is opposite to that of signal ③. Therefore, it is presumable that the output of the NNC counteracts the output of the PI/LEAD controller. Whereas the control input signal ④ generated by the PI/LEAD of the neuro-controller does not properly compensate the frictional force, the sum of control input signals ② and ④ seems to compensate the frictional force properly.



(a)

(b)

Fig. 6. Angular velocity of the mirror with coulomb friction and random disturbance, (a) by the PI/LEAD controller and (b) by the neuro-controller.



(a)

(b)

Fig. 7. Control inputs (a) by the NNC and (b) by the PI/LEAD of the neuro-controller.

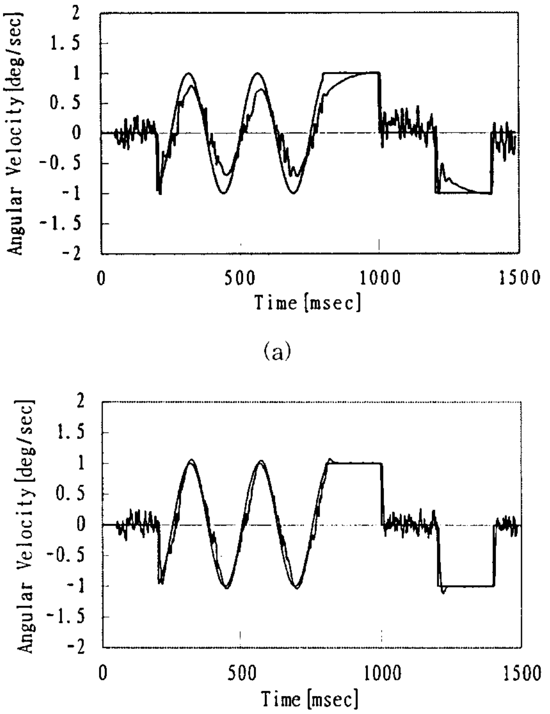
Generalization capability is also tested with a different command signal as shown in Fig. 8. We see that the tracking performance of the neuro-controller is superior to that of the PI/LEAD controller. Also, we investigate the angular velocity of the mirror when the magnitude of the coulomb friction changes with time as follows.

$$f = \begin{cases} 0.15, & t < 0.75 \\ 0.25, & 0.75 \leq t < 1.1 \\ 0.2, & t \geq 1.1 \text{ (sec)} \end{cases} \tag{46}$$

The response in Fig. 9(b) illustrates the adaptation capability of the neuro-controller for friction compensation, while the response in Fig. 9(a) shows a poor tracking performance. When the magnitude of friction changes with time, the mirror of the LSS controlled by the neuro-controller tracks fast to the command signal even if the tracking error is large with the PI/LEAD controller.

On the zero velocity command signal, the LOS mirror should maintain the orientation under any disturbance. The result in Fig. 10 shows the stabilization performance with a sinusoidal disturbance with magnitude of 1deg/sec and frequency of 5Hz and a coulomb friction with the magnitude 0.2. When the polarity of sinusoidal wave is reversal, all the responses in Fig. 10 show high peak overshoot because the sign of the coulomb friction changes with the polarity of the sinusoidal wave. But, the trajectory (ii) converges fast to zero velocity than the trajectory (i) because the neuro-controller minimizes the velocity error by means of friction compensation.

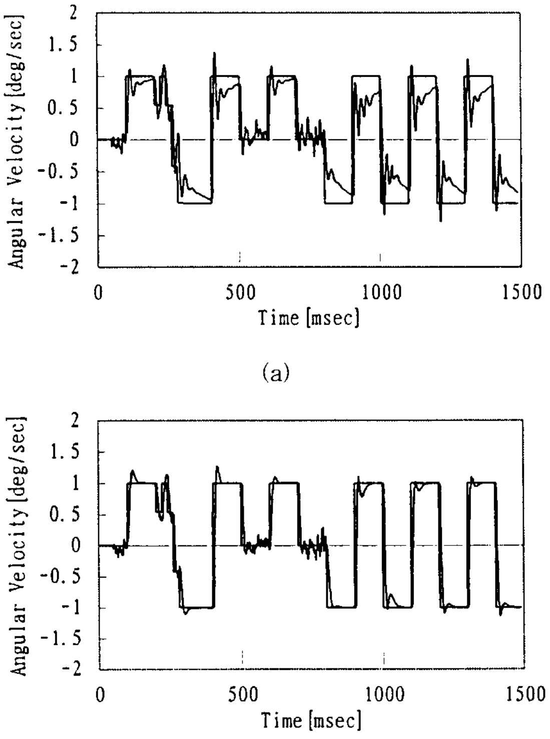
We also investigate the angular velocity of the mirror with a constant disturbance in the time interval between 0.7 and 0.88 second and a random disturbance in the rest of time intervals. The magnitude of the constant disturbance is 0.2 deg/sec and the random disturbance is uniformly distributed between 0.2 and -0.2 deg/sec. Fig. 11(a) shows positive angular velocity at time interval between 0.7 second and 0.88 second (indicated by an arrow), but the response in Fig. 11(b) converges to the zero velocity since the neuro-controller compensates the frictional force.



(a)

(b)

Fig. 8. Angular velocity of the mirror with the different command signal, (a) by the PI/LEAD controller and (b) by the neuro-controller.



(a)

(b)

Fig. 9. Angular velocity of the mirror with different magnitude of coulomb friction, (a) by the PI/LEAD controller and (b) by the neuro-controller.

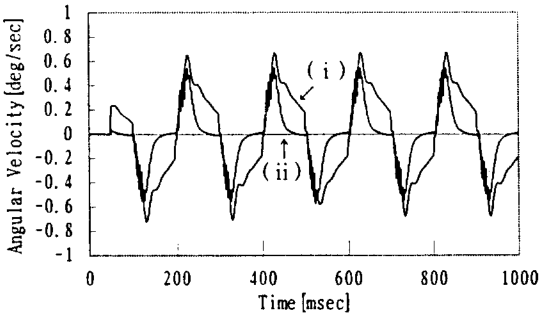
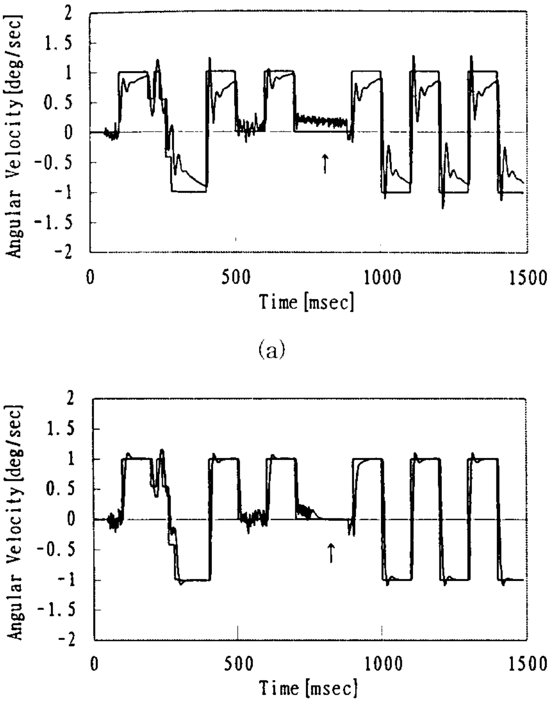


Fig. 10. Angular velocity of the mirror with a sinusoidal disturbance and a coulomb friction. (i) by the PI/LEAD controller and (ii) by the neuro-controller.



(a)

(b)

Fig. 11. Angular velocity of the mirror with random and constant disturbance, (a) by the PI/LEAD controller and (b) by the neuro-controller.

Finally, we investigate the angular velocity of the mirror when there exist coulomb friction with width parameter D as in Fig. 12 and a disturbance measured at a field test (Fig. 13). The simulation result shows good tracking and disturbance rejection properties of the neuro controller, compared with the result of the PI/LEAD controller. The neuro-controller properly compensates the friction even if the coulomb friction has the width parameter D.

Ⅵ. Conclusion

In this paper, we have developed a new neuro-controller for friction compensation of the LSS. The neuro-controller consists of a conventional PI/LEAD controller and an NNC. Indirect learning control scheme for training the NNC is adopted. The tracking error and

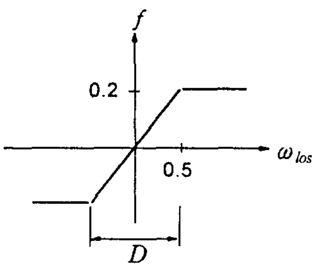


Fig. 12. Coulomb friction with width parameter D.

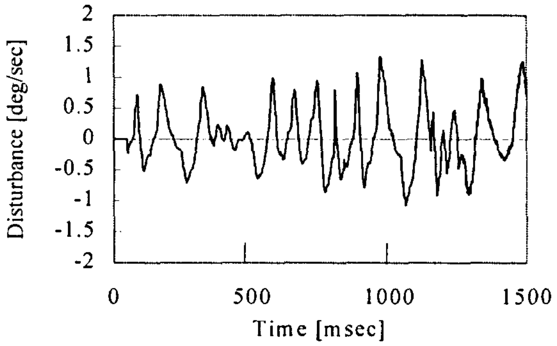
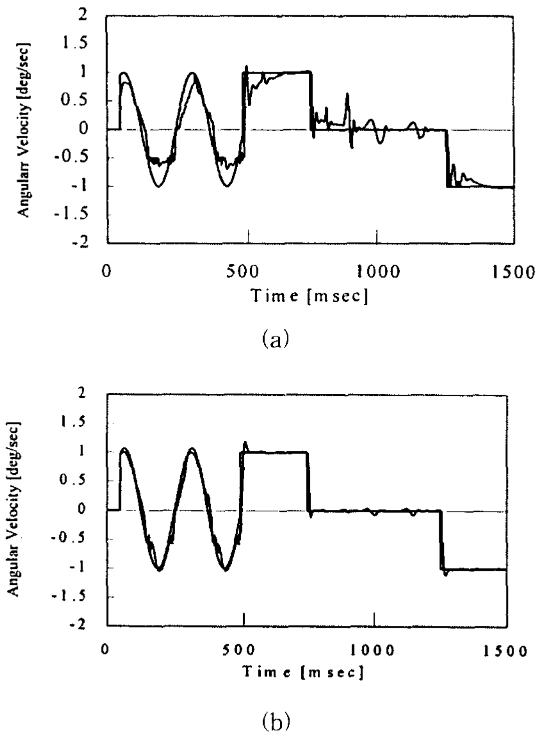


Fig. 13. The disturbance measured at a field test.



(a)

(b)

Fig. 14. Angular velocity of the mirror with coulomb friction with width parameter D and disturbance by field measurements, (a) by the PI/LEAD controller and (b) by the neuro-controller.

its integral are used as inputs to the NNC to compensate frictional force, while the filtered control input is also used as an additional input to the NNC to reduce the overshoot of the control system. It has been verified by simulations that the neuro-controller senses the tracking error and compensates the frictional force adaptively when the magnitude of coulomb friction is piecewise constant and that the NNC predicts the large overshoot ahead of time and makes proper correcting efforts before the overshoot actually occurs. The stabilization performance of the neuro-controller is excellent compared with that of the PI/LEAD controller under random, sinusoidal and constant disturbances. Future research on this subject is to find an optimum value of the filter constant so that the best tracking and stabilization performance can be guaranteed.

References

[1] C. Canudas, K. J. Astrom and K. Braun, "Adaptive friction compensation in DC-motor drives," *IEEE J. Robot. Automat.*, vol. RA-3, no. 6, pp. 681-685, Dec., 1987.

[2] B. Friedland and Y. J. Park, "On adaptive friction compensation," *IEEE Trans. on Auto. contr.*, vol. 27, no. 10, pp. 1609-1612, Oct., 1992.

[3] B. Armstrong-Helouvry, P. Dupont and C. Canudas De Wit, "A survey of models analysis tools and compensation method for the control of machines with friction," *Automatica*, vol. 30, no. 7, pp. 1083-1138, 1994.

[4] M. Kawato, Y. Uno, M. Isobe and R. Suzuki, "A hierarchical model for voluntary movement and its application to robotics," *IEEE Contr. Syst. Mag.*, vol. 8, pp. 8-15, Apr., 1988.

[5] F. L. Lewis, A. Yesildirek and K. Liu, "Multilayer neural-net robot controller with guaranteed tracking performance," *IEEE Trans. on Neural Networks*, vol. 7. no. 2, pp. 388-399, March, 1996.

[6] G. Lightbody and G. W. Irwin, "Direct neural model reference adaptive control, "*IEE proc. D*, vol. 142, no. 1, pp. 31-43, Jan., 1995.

[7] D. H. Nguyen and B. Widrow, "Neural networks for self-learning control systems," *IEEE Contr. Sys. Mag.* vol. 10, pp. 18-23, Apr., 1990.

[8] A. K. Rue, "Stabilization of precision electrooptical pointing and tracking systems," *IEEE. Trans. on Aerospace and electronic systems*, vol. AES 5, no. 5, pp. 805-819, Sep., 1969.

[9] B. G. Jeon et al., "Dynamic analysis of gimbal structure of LOS stabilization system," *Technical report*, Agency for Defence Development, 1994.

[10] J. G. Ziegler and N. B. Nichols, "Optimum settings for automatic controllers," *Trans. ASME*, vol. 64, pp. 759-768, 1942.

[11] D. E. Rumelhart, G. E. Hinton and G. E. Williams, "Learning internal representations by error propagation," in D. E. Rumelhart and J. McClelland(eds.), *Parallel Distributed processing*, Cambridge, MIT Press, vol. 1, ch 8. 1986.

장준오

1988년 경북대학교 전자공학과 졸업. 1992년 동 대학원 전자공학과 졸업(공학석사). 1992년 ~ 현재 동 대학원 박사과정. 주관심분야는 시스템 모델링, 지능제어, 공장자동화등임.

전병균

1985년 경북대학교 전자공학과 졸업. 1987년 동 대학원 전자공학과 졸업(공학석사). 1995년 ~ 현재 동 대학원 박사과정. 1987년 ~ 현재 국방과학연구소 선임연구원. 주관심분야는 시스템 모델링, 지능제어, 정밀모터제어등임.

전기준

제어·자동화·시스템공학회 논문지 제 2권 제 1호 참조.