Unknown Parameter Identifier Design of Discrete-Time DC Servo Motor Using Artificial Neural Networks

Dong-Seog Bae and Jang-Myung Lee

Abstract: This paper introduces a high-performance speed control system based on artificial neural networks(ANN) to estimate unknown parameters of a DC servo motor. The goal of this research is to keep the rotor speed of the DC servo motor to follow an arbitrary selected trajectory. In detail, the aim is to obtain accurate trajectory control of the speed, specially when the motor and load parameters are unknown. By using an artificial neural network, we can acquire unknown nonlinear dynamics of the motor and the load. A trained neural network identifier combined with a reference model can be used to achieve the trajectory control. The performance of the identification and the control algorithm are evaluated through the simulation and experiment of nonlinear dynamics of the motor and the load using a typical DC servo motor model.

Keywords: ANN, terminal voltages, electric driving system, unknown nonlinear dynamics, reference model, time-varying

I. Introduction

In a high performance electric driving system, the position of a rotor or the speed of a rotor follows the predetermined trajectory over an entire region. Such a high-performance driving system is essential where precise motion is required such as in the areas of robotics, actuation, and guided manipulation. A high-speed controller is indispensable in a driving system. The aim of a high-speed controller is to adjust terminal voltages so that the speed of a rotor can pursue the given trajectory with minimum error, and this can be accomplished by supplying suitable control signals from a general converter.

In an electric driving system, one of the important problems of traditional tracking controller is that unknown load parameters which have wide operating point cannot be acquired. This coincides with the problems of each controllers[1]-[4]. There are many techniques to overcome such problems. For example adaptive controller overcomes them by identifying entire behavior of a DC servo motor using the linear parametric(ARMAX) over given time intervals. But, generally, a load torque is a nonlinear function of the combination of variable numbers such as the position or the speed of a rotor. Therefore identifying the whole nonlinear system through a linearized model around the widely varying operating point with a fast-switching frequency can cause an error inducing unstable or inaccurate performance[5].

The application of neural network for training non-linear function is well-known[6][7]. Neural networks are trained to emulate non-linear plant dynamics by implementing sets of input-output pattern properly. If system dynamics is identified by ANN, traditional control techniques can be applied in order to reach their particular aim. One specific technique the indirect model reference adaptive control(MRAC)[6][7], is useful for the application of trajectory control. In this paper, we propose a method on control topology based on MRAC techniques for trajectory control and identification of DC servo motor based on ANN. To have no previous information the load dynamics is to have no previous information on parameters of DC servo motor. We limited experiment to the identification process having time-invariant model structure because training of ANN is performed through static back-propagation. In the case of time-varying system, recurrent network method[9][10] can be used. This paper consists of the following sections: In section 2, dynamics of DC servo motor and necessary functions for ANN training are described. In section 3, simple introduction for ANN simulator and how ANN emulates function of DC servo motor are presented. In section 4, the performance of identification of DC servo motor is measured. In section 5, the controller topology about identified model is introduced and experiments are presented. In section 6, results and ideas on future research are presented.

Manuscript received: June 7, 2000., Accepted: Aug. 29, 2000.
Dong-Seog Bae: Dept. of Automatic Production, Chang -Won Polytechnic Collage
Jang-Myung Lee: Dept. of Electronics Engineering, Pusan National University

II. DC servo motor model

DC servo motor provides advanced algorithm for stable electrical drives and the omni-directional characteristics of linearity. And, as introduced in references[1-3], it is an ideal factor for trajectory control. In the point of control system, DC servo motor can be considered as SISO plant. Therefore, complications related to multi-input system are discarded.

1. Motor dynamics

Dynamics of DC servo motor is calculated by using the following two equations :

$$K\omega_p(t) = -R_a i_a(t) - L_a[di_a(t)/dt] + V_t(t) \quad (1)$$

$$Ki_a(t) = J[d\omega_p(t)/dt] + D\omega_p(t) + T_L(t) \quad (2)$$

where,

$\omega_p(t)$: rotor angular velocity, rad/s

$V_t(t)$: input voltage, V

$i_a(t)$: armature current, A

$T_L(t)$: load torque, Nm

J : rotor inertia, Nm2
K : torque & back emf constant, NmA-1
D : damping constant, Nms
R_a : armature resistor, Ω
L_a : armature inductance, H.

Load torque, $T_L(t)$, can be expressed as function of a rotor angular velocity as follows:

$$T_L(t) = \psi(\omega_p). \tag{3}$$

Function $\psi(\omega_p)$ is dependent on load and accurate function is assumed to be unknown.

2. Discrete-time DC servo motor model

In order to acquire training data of ANN and apply control algorithm, discrete time DC servo motor model is necessary. Therefore, we define the nonlinear model is defined as follows :

$$T_L(t) = \xi\omega_p^2(t)\,[\mathrm{sign}\,(\omega_p(t))] \tag{4}$$

where, ξ is constant. In equation (4), the direction of load torque is always opposite to the direction of rotation. The reason for choosing this special function is that this function presents general characteristics on driving propeller and load shaped like a pan. But the choice of load torque is completely arbitrary and this function doesn't influence the proposed control algorithm.

Discrete-time model can be established by substituting all continuous differential as finite difference after combining equations (1),(2),(4). State-space representation is as follows:

$$\begin{aligned}\omega_p(k+1) = {} & \alpha\omega_p(k) + \beta\omega_p(k-1) \\ & + \gamma[\mathrm{sign}(\omega_p(k))]\omega_p^2(k) \\ & + \delta[\mathrm{sign}(\omega_p(k))]\omega_p^2(k-1) + \zeta V_t(k)\end{aligned} \tag{5}$$

where, α, γ, ζ are constants based on motor parameters J, k, D, R_a, L_a and sampling time T ,and δ, β are functions of ξ. The value k denotes the kth time step. The value of DC motor parameter with name plate ratings of 1HP, 220[V], 1100[rpm] are as follows:

$J = 0.068\ Kgm^2$
$K = 3.475\ NmA^{-1}$
$R_a = 7.56\ \Omega$
$L_a = 0.055\ H$
$D = 0.03475\ Nms$
$\xi = 0.0039\ Nms^2$
$T = 40\ ms.$

III. ANN and back propagation learning

The structure of typical 3-layer feed-forward ANN is shown in Fig. 1. It consists of input-layer, output-layer and two hidden layers and a set of node is arranged in these layers. Active signals are converted into weak or amplified signals and transferred to next layers through link.

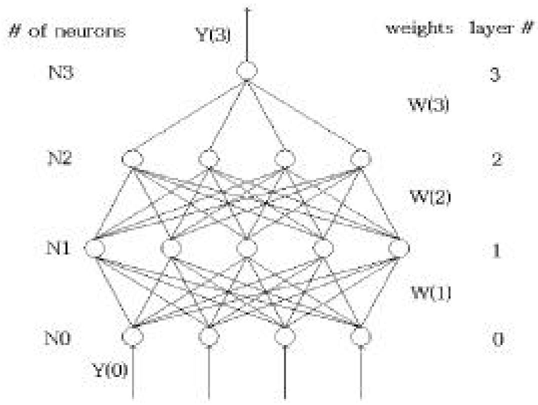

Fig. 1. Topology of a 3 layer feedforward ANN.

1. The structure of multi-layer feed-forward network

Each ANN nodes consisting of H layer is expressed by the following two equations:

$$u_i(h+1) = \sum_{j=1}^{Nh} w_{ij}(h+1)Y_j(h) + \theta_i(h+1) \tag{6}$$

$$Y_i(h+1) = f[u_i(h+1)] \tag{7}$$

where,

$w_{ij}(h+1)$: the weight between i^{th} neuron of h+1 layer and j^{th} neuron of h layer
$\theta_i(h+1)$: the threshold of i^{th} neuron of $h+1^{th}$ layer
$u_j(h+1)$: input for i^{th} neuron in the layer of $h+1^{th}$
$Y_i(h)$: input for i^{th} the action of neuron in the layer of h^{th}
$f[\cdot]$: the active function of sigmoid
N_h : the number of neuron in the layer of h^{th}
i, j, h is $1 \le i \le N_{h+1}$, $1 \le j \le N_h$ & $0 \le h \le H-1$.

2. Back-Propagation Learning

ANN imitates the functions by implementing the set on the patterns of input-output function. The back-propagation learning method minimizes the difference between the desired output and the real output for all the given patterns to adjust the weight in the link and the threshold in the node. The reason for this is that Pth training pattern $(P = 1, \ldots\ldots P)$ minimizes the following energy function for all the weight and threshold.

$$E_p = (1/2)\sum_i (T_i - Y_i(H))^2 \tag{8}$$

$Y_i(H)$ corresponds to the action of i^{th} neuron in output layer H. Also, T_i is desired value. Update value corresponding to weight value is computed by gradient descent technique.

$$w_{ij}^{new}(h) = w_{ij}^{old}(h) + \eta\,\partial E_p/\partial w_{ij}(h) + \alpha\nabla w_{ij}(h) \tag{9}$$

The quantity for $\partial E_p / \partial w_{ij}(h)$ is calculated by the following equation:

$$\partial E_p / \partial w_{ij}(h) = \delta_i(h) f[u_i(h)] \cdot Y_j(h-1) \quad (10)$$

$$\delta_i(h) = \sum_j \delta_j(h+1) f[u_i(h+1)] \cdot w_{ji}(h+1) \quad (11)$$

where, $\delta_i(H) = -(T_i - Y_i(H))$.

Generally, the above algorithm is known for the error back-propagation algorithm. The constant η is the learning step and υ is momentum gain. $\nabla w_{ij}(h)$ presents the variation of weight for the repetition beforehand. The weight is improved for all P pattern and the learning process desires a lot of modification. The enough learning is accomplished when the sum of the total error function that is included in the collection of all the P training pattern is decreased under the threshold ε that is selected beforehand. Namely,

$$E_{total} = \sum_p E_p < \varepsilon, \qquad p = 1, \ldots, P. \quad (12)$$

More details about back-propagation algorithm are mentioned in reference[6]. In designing and training ANN to imitate function, the only fixed parameters are input and output for ANN. These are is based on input-output variables of function, and two hidden layers learn arbitrary nonlinear factor. But the number of hidden neurons and parameters η, υ, ε and P are independent of statistical learning, not supported by settled standard and the selection is normally based on the experience. The final object is to find the combination of parameters yielding whole errors in the extent of reasonable learning numbers.

IV. ANN for system identification and control

Fig. 2 shows the basic concept of identification of DC servo motor and control. This is similar to indirect model reference adaptive control[5][7], where, DC servo motor is first identified by the combination of ANN input and output variables. Weights from trained ANN identifier are used to calculate terminal voltage and this will derive the DC servo motor velocity $\omega_p(k)$ to move asymptotically toward

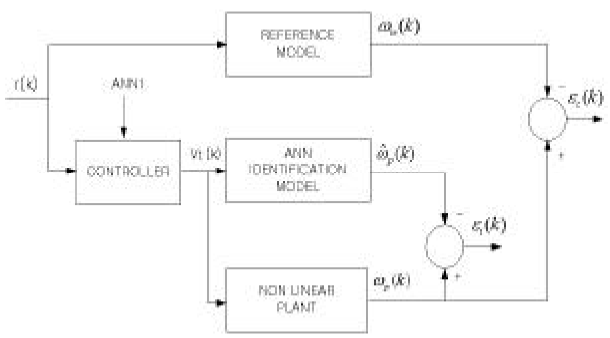

Fig. 2. ANN Based identification and control system.

reference model output $\omega_m(k)$. $\varepsilon_i(k)$ and $\varepsilon_c(k)$ of Fig. 2 are defined as discrimination and tracking error, respectively. The aim of discrimination is to minimize the error

$$[\varepsilon_i(k)]^2 \;\forall\; kT \in [0, t_f].$$

Or, we minimize $[\varepsilon_i(k)]^2$ in $kT \in [0, t_f]$. This is the window in time $[0, t_f]$. During this period, ANN is learning, it finishes when the value of tracking error meet minimum value that we want to and then, Non linear plant is identified.

Here, the control strategy is to calculate the most optimal terminal voltage $V_t(k)$, and this minimizes tracking error. The behavior of desired DC servo motor is described through the adjacent reference model. And the bounded control sequence $r(K)$ to desired velocity trajectory $\omega_m(k)$ can be derived by using the reference model. It is used as an active signal of control system.

1. Identification of DC servo motor

The characteristics of DC servo motor are identified by making the set of input and output pattern for ANN and properly setting the weighting value with error back-propagation. The intensity of training depends on degree of complexity of dynamics. When it comes to training of ANN, one of the work done first is to define the operation area to input and output variables of ANN. The researchers confined the operation space as follows to unite the limit of mechanical and electric hardware agree with hypothetical scenario

$$-30.0 < \omega_p(k) < 30.0 \text{ rad/s}$$

$$|\omega_p(k-1) - \omega_p(k)| < 1.0 \text{ rad/s}$$

$$|V_t(k)| < 100 \text{ [V]}.$$

The performance of ANN discrimination is evaluated by comparing the estimated output and real output of motor to typical arbitrary excitation signal.

2. Topology

Equation (5) becomes equation (13).

$$V_t(k) = g[\omega_p(k+1), \omega_p(k), \omega_p(k-1)]. \quad (13)$$

Here, $g[\cdot]$ is derived from equation (11) and we suppose that is unknown.

$$\begin{aligned} g[\omega_p(k+1), \omega_p(k), \omega_p(k-1)] = {} & [(\omega_p(k+1) - \alpha\omega_p(k) \\ & - \beta\omega_p(k-1) - \lambda[\mathrm{sign}(\omega_p(k))]\omega p^2(k) \quad (14) \\ & - \delta[\mathrm{sign}(\omega_p(k))]\omega_p^2(k-1)/\xi \end{aligned}$$

ANN learns to imitate $g[\cdot]$. $\omega_p(k)$, $\omega_p(k-1)$ $\omega_p(k+1)$, are independent variables of $g[\cdot]$ and inputs of ANN. The corresponding desired output $g[\omega_p(k+1), \omega_p(k), \omega_p(k-1)]$ is calculated from equation (14). This is also equal to $V_t(k)$ of equation (13). Desired output $V_t(k)$ corresponding to input pattern of randomly raising $[\omega_p(k+1), \omega_p(K), \omega_p(K-1)]$ is used for off-line training. Training data comes from operation area defined in 4.1.

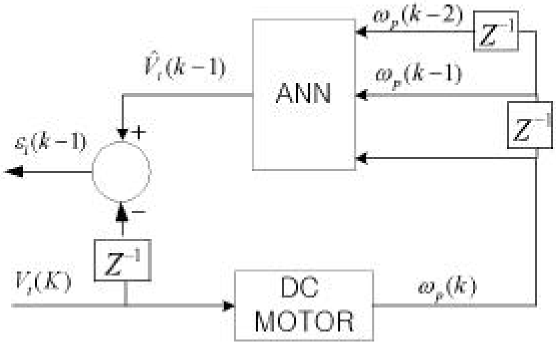

Fig. 3. Structure of the ANN identifier.

Fig. 3 shows a model structure of serial-parallel ANN discriminator. Z^{-1} is unit time-delay. The estimated terminal voltage of DC servo motor is derived from equation (15).

$$\hat{V}_t(k-1) = N[\omega_p(k), \omega_p(k-1), \omega_p(k-2), \omega_p(k-3)] \quad (15)$$

where, "^"represents the estimated voltage, and N[·] is output to the input "·".

The following statistics describes the simple concept including training result and ANN Topology:

No. of input : 3
No. of output : 1
No. of hidden layer : 1
No. of hidden neuron : 6
No. of training Patterns P : 1500
No. of training Sweep : 5000
Training Step η : 0.1
Momentum gain α : 0.4
Total threshold ε of E : 0.01 .

The performance of trained ANN discriminator is estimated through the excitation of the DC servo motor model by using the following voltage sequence and compared with output of equation (15) and $V_t(k)$.

$$V_t(k) = 50\sin(2\pi kT/6) + 45\sin(2\pi kT/4) \quad (16)$$
$$\forall\ kT \in [0, 20/\mathrm{sec}].$$

Here, it is restricted in $kT \in [0, 20/\mathrm{sec}]$. Fig. 4 shows the result. The highest error of estimator is 2.3[V](2.3%) under the highest application voltage 100[V]. In the Topology, for the discrimination model it is very important not to suppose the capability of using the parameter of load and DC servo motor. As a result, the training function is very complex. So, the result of calculation needed for training ANN is a large value, the same as in the training statistics.

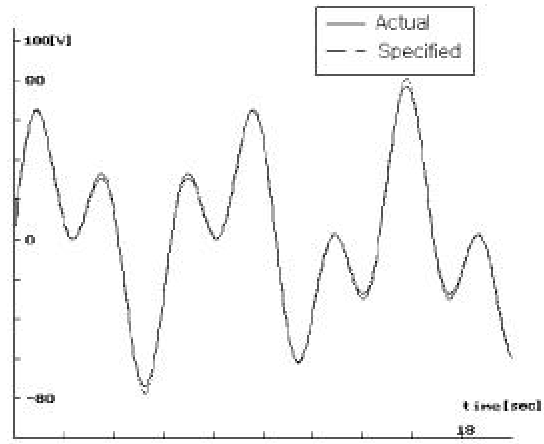

Fig. 4. Actual and estimated terminal voltages of the DC servo motor.

V. Experiment of DC servo motor using learned ANN

1. Trajectory control of DC servo motor

The purpose of this control system is that the velocity of motor tracks the desired trajectory $\omega_m(k)$ by using DC servo motor. This is accomplished by making DC servo motor track the output of chosen reference model. The following second-order reference model is chosen.

$$\omega_m(k+1) = 0.6\omega_m(k) + 0.2\omega_m(k-1) + r(k) \quad (17)$$

where, $r(k)$ is the bounded input to the reference model. The coefficients are chosen to assure that the poles are in the unit circle and that is the response type that can be accomplished from DC servo motor. The control sequence $r(k)$ corresponding to desired sequence $\omega_m(k)$ can be calculated by using equation (17). In section 4, learned ANN is used for Topology to track input voltage $V_t(k)$ of DC servo motor, and assures that DC servo motor velocity, $\omega_p(k)$, correctly pursues trajectory. In order to evaluate the performance of controller the velocity trajectory is simulated, $\omega_m(k)$, selecting arbitrarily. Indicated figure comparison and actual velocity trajectory are as follows : Because selected reference model is stabilized asymptotically, it can be supposed that tracking error moves toward "0", and predict velocity at $(K+1)$th sampling time in the equation (18):

$$\hat{\omega}_p k+1 = 0.6\omega_p(k) + 0.2\omega_p + r(k). \quad (18)$$

The result can be feed back to ANN that was learned in Topology in oder to estimate the control input at Kth sampling time

$$\hat{V}_t(k) = N[\hat{\omega}_p(k+1), \omega_p(k), \omega_p(k-1)] \quad (19)$$

Overall structure and control system for identification based on ANN is shown in Fig.5. The ability of the model tracking control is verified through the experiment on two arbitrarily selected trajectory. As previously described, first denoting $\omega_m(k)$, the researchers induced corresponding [r(k)]. For the above trajectory, the corresponding [r(k)] is derived by using equation (18). [r(k)] is applied to model in Fig. 4 and the matrix $\propto_m^T$ corresponds to reference model coefficient[0.6 0.2]. Fig. 6 describes tracking performance on the same sinusoidal reference velocity trajectory as indicated in the previous section, while Fig. 7 compares actual performance on sigmoid velocity trajectory. Fig. 6 and Fig. 7 show that tracking performance is better. In addition, suggested algorithm has the advantage that DC servo motor is entirely independent of the load parameter. Here, improved performance can be acquired by learning more of ANN.

Electric drive system must show the superior characteristics in a noisy environment. Noise is caused by several reasons, and the main reason is resolution error and position error for DC servo motor, drift of load parameter and quantization velocity. High performance operating controller should be strong enough robust to maintain accurate tracking performance regardless of the operating environment. The controller based on ANN has its own ability to reduce noise. Therefore, the controller will show the better performance in a noisy operating environment. In a noisy environment, to inspect the controller performance, after mixing uniform random variable of ∓ 2 rad/sec with the content in Fig. 7, the velocity is repeatedly measured,

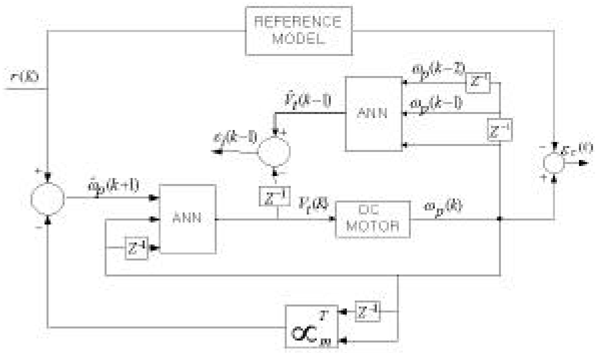

Fig. 5. Overall structure of the controller.

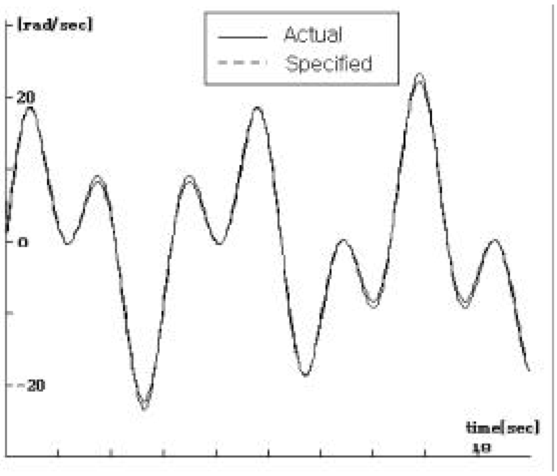

Fig. 6. Tracking performance for a sinusoidal reference trajectory.

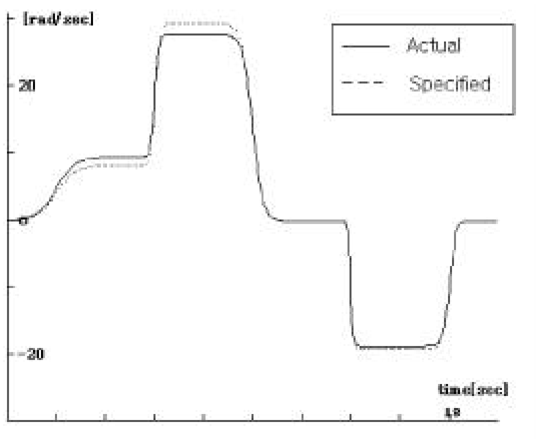

Fig. 7. Tracking performance for a sigmoid reference.

$\omega_p(K)$ of DC servo motor. Here, the velocity $\omega_p(K)$ is used as the input of ANN controller. Fig. 8 shows the tracking performance in environment with noise continuously generated in Topology. As compared to Fig. 7, tracking accuracy is continuous under active mode including noise and a stabilized controller performance is observed.

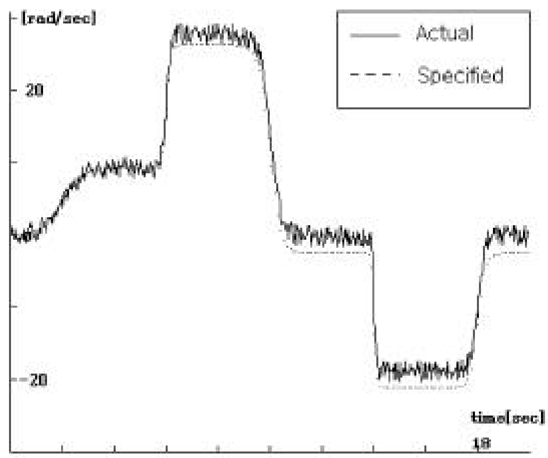

Fig. 8. Tracking performance for a sigmoid reference track with measurement noise.

2. The experiments in hardware

Fig. 9 shows the overall structure of controller used in the experiment. The system is composed of output control signal in itself by programming in the personal computer(PC). The purpose is to get a faster and more accurate control signal by not using micro controller specifically. The specific structure is as follows: The control output signal(0V 10V) is made by the 12bit D/A convertor in the PC, and is converted to bipolar signal of 0[V] 5[V], -5[V] 0[V] in the level shift circuit, and then, used absolute circuits which made (-)voltage when the motor is backward, and (+)voltage when forward.

In the velocity control of motor PWM method is used, and in the switching of it IGBT, the element of electric power is used. In addition, in the extraction of rotor speed the noise of tacho generator through LPF(Low Pass Filter) is reduced. and converted into digital voltage through 12-bit A/D convertor, and then used for feedback signal to rotor speed. The specifications of DC servo motor and tacho generator used in experiment are as follows(Table 1):

Table 1. Spec. of motor and tacho generator used in experiment.

Motor	Maximum input voltage	40VDC
	Maximum torque	12.5kg · cm
	Maximum speed	1,200rpm
	Maximum current	5.3A
Tacho Generator	Output	3V/1,000rpm

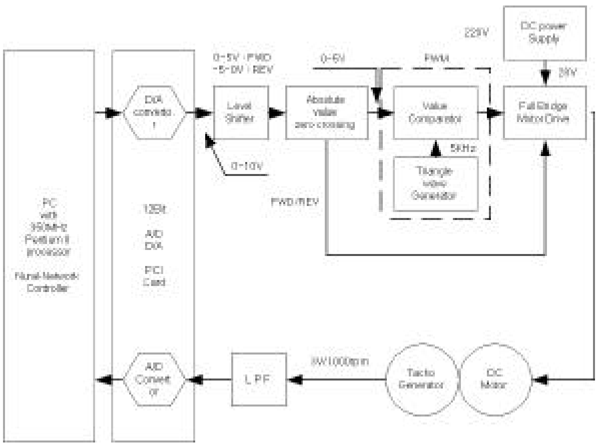

Fig. 9. Overall structure of experimented control system.

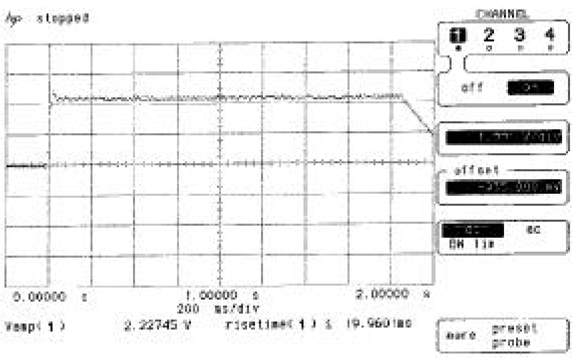

Fig. 10. Tracking performance of control algorithm to DC servo motor.

In the experiment, the proposed algorithm in this paper was applied to DC servo motor to evaluate the performance of controller. This time, the desired value of rotor speed was set to sinusoidal signal of 890[rpm], and the sampling period T was 1.675[msec]. Furthermore the object of 1[kg] is connected to rotor so that random torque can be generated in case of variation of load torque. According to voltage shown by Fig 10, it indicates about 2.2[V]. At Table 1, it is 1000[rpm] under 3[V], therefore, rotor speed of motor is 890[rpm]. This is rotor speed of motor which we want to track. At the below graph of Fig. 10, it exists about 0.1[V]. It indicates 25[rpm] of rotor speed. therefore, It shows that there is only tracking error of 2.8[%] about 890[rpm]. The steady state error in the experiment was 25[rpm](2.8%) [Fig.10]. The algorithm proposed in this paper produced a superior performance in the identification of parameters despite the load variation of the motor.

VI. Conclusion

The trajectory control and experiment on unknown discrete time DC servo motor were performed successfully. Nonlinear operating characteristics and load of unknown time-variant discrete time DC servo motor could be found by ANN, and this result is very important factor in experiment, so enough learning was required. In order to acquire better performance of trajectory control of discrete time DC servo motor velocity, this paper used ANN adding the concept of model reference adaptive control.

Through the experiment, the tracking performance of the proposed control algorithm was more superior. Also, the researchers conducted the experiment in a noisy condition to study the degree of robust controller. The researchers suggest future studies on controllers which use intelligent control algorithm application.

Reference

[1] D. S. Bae, "Parameter identification and trajectory control of a DC motor using neural networks," *Proceedings of the fifth international symposium on artificial life and robotics*, vol. 2, pp. 833-837, Jan., 25-29, 2000.

[2] D. Miche, "Learning from humans strategies of motor control: a challenge for robotic systems design," *5th IEEE International Workshop on Robot and Human Communication*, pp. 223-228, Nov., 1996.

[3] K. S. Narendra and K. Parthasarathy, "Identification & control of dynamical systems using neural network" *IEEE Transaction on Neural Network*, vol. 1, pp. 4-27, Mar., 1990.

[4] S. Weerascooriya and M. A. El-Sharkawi, "Adaptive tracking control for high performance DC drives," *IEEE Transaction on Energy Conversion*, vol. 5, pp. 122-128, Sept., 1989.

[5] S. Weerasooriya and M. A. El-Shakawi, "Identification and control of a DC motor using back-propagation neural network," *IEEE Transaction on Energy Conversion*, vol. 6, pp. 663-669, Dec., 1991.

[6] J. M. Zurada, Introduction to Artificial Neural Systems, pp. 256-358.

[7] Mohammad Bahrami Keith E. Tate, "Neural networks as complementary direct controller of nonlinear plants," Univ. of New South Wales. Proc. of the 1993 *IEEE International Symposium on Intelligent Control*, pp. 122-126, , Chicago, Illinois, USA, Aug., 25-27, 1993

[8] Cincotti S, Marchesi M, and Serri M, "A neural

network model of parametric non-linear hysteretic inductors," *IEEE Transactions on Magnetics*, vol. 34, no. 5, P.1, pp. 3040-3053, Italy Sept., 1998.
[9] M. A. Brdys, and G. J. Kulawski, "Dynamic neural controllers for induction motor," *IEEE Transactions on Neural Networks*, vol. 10, no. 2, pp. 340-355, Mar., 1999.
[10] K. Pakdamam, J. F. Vibert, Eric Boussard, and N. Azmy, "Single neuron with recurrent excitation: Effect of the transmission delay," *Neural Networks*, vol. 9, no. 5, pp. 797-818.

Dong Seog Bae
was born in 1967 in Kyungnam, Korea. He received the B.S. degree in Electronics at Bukyung Univ., the M.S. degree in Electronics at Dong-a Univ., and he has been a Ph.D. student in Electronics at Pusan Nat'l Univ. His research interests are control using nural networks, digital audio system design, manufacturing automation, and intelligent control.

Jang Myung Lee
(S'90-M'90) was born in Kyungnam, Korea, in 1957. He received the B.S. degree and the M.S. degree in electronics engineering from Seoul National University, Seoul, Korea, in 1980 and 1982, respectively, and the Ph.D. degree in computer engineering from the University of Southern California, Los Angeles, in 1990. Since 1992, he has been with the Department of Electronics Engineering, Pusan National University, Pusan, Korea. His research interests are robotics, integrated manufacturing systems, and intelligent control.