The Research of Sampling Frequency for A DC Servo Motor Speed Control System Based on Neural Networks

Ming-Huei Chu
Department of Mechatronic Technology, Tungnan University
Taipei, Taiwan

Yi-Wei Chen
Department of Mechatronic Technology, Tungnan University
Taipei, Taiwan

Yuan Kang
Department of Mechanical Engineering, Chung Yuan Christian University
Chung Li, Taiwan

Zhi-Wei Chen
Department of Mechatronic Technology, Tungnan University
Taipei, Taiwan

Abstract—This study utilizes the sampling frequency will affect the performance of a direct neural controller (DNC), which is applied to a DC motor speed control system. A direct neural controller of self-tuning strategy is proposed and treated as a speed regulator to keep the motor in constant speed without the specified reference model. A tangent hyperbolic function is used as the activation function, and the back propagation error is approximated by a linear combination of error and error's differential. The simulation results reveal that the proposed speed regulator keeps motor in constant speed with high convergent speed, but the convergent speed is affected by the sampling frequency. In general, the high sampling frequency will make the speed control system have favor performance, but it will take lots of CPU time. This study applies off-line training to evaluate the appropriate initial values of neural connective weights, then the speed control performance will be improved under low sampling frequency condition.

Keywords- DC servo motor, Speed control, Sampling frequency, Neural networks

I. INTRODUCTION

In recent years, the multi-layers neural networks are applied to modern precise DC servo systems for providing the accurate performance under the unknown nonlinear friction, system parameters variations and torque load variations conditions. The neural network controllers have been used in various fields owing to their capability of on-line learning and adaptability. The indirect control strategy has been widely used in DC servo systems [1-3]. It is a two-step process including identification of plant dynamics and control. In the indirect control strategy, a sub-network (called "emulator") is required to be trained before the control phase, and the quality of the trained emulator is crucial to the controlling performance. It is therefore very important that the data sets for training the emulator must cover a sufficiently large range input and output pares, but it is very possible that the future behaviors in on-line control may outside the range that was used during the emulator's training, the back propagation through the emulator fails, causing poor or even unstable control performance.

The direct control strategy can overcome this problem if a priori qualitative knowledge or Jacobian of the plant is available. By this method, the plant can be viewed as an additional but no modifiable layer of the neural network. The error between the actual and desired outputs of the plant is used to update the connective weights. In this sense, the controller learns continuously, and hence it can control plants with time-varying characteristics. But it is usually difficult to approximate the Jacobian of an unknown plant. Zhang and Sen [4] presented a direct neural controller for on-line industrial tracking control application, and a simple sign function applied to approximate the Jacobian of a ship track keeping dynamics. The results of a nonlinear ship course-keeping simulation were presented, and the on-line adaptive control was available. But their schematic is not feasible for high performance motion controls. A motion control system needs a neural controller with faster convergent speed. Lin and Wai [5,6] proposed the δ adaptation law to increase the on-line learning speed. They designed a neural network controller with the δ adaptation law for PM synchronous servo motor drive, and preserved a favorable model-following characteristic under various operating conditions. Lin et al. [7] have proven the convergence of the learning algorithm with the δ adaptation law. Chu et al. [8] proposed a linear combination of error and error's differential to approximate the back propagation error. By this way, the convergent speed will be increased. But the linear combination needs two parameters which are unknown and usually determined by try and error. Kang et al. [9] proposed a neural tuner, which can tune the unknown parameters on line from some initial values, and the appropriate parameters can be obtained by on line tuning.

This study utilizes the direct neural control applied to control the speed of an 18W DC servo motor. The proposed neural controller is treated as a speed regulator to keep the motor of constant speed, and the close loop control system doesn't apply any specified reference model. A tangent hyperbolic function was used as the activation function to improve the convergent speed. The linear combination of error and error's differential with the appropriate parameters is used to approximate the back propagation error. The appropriate parameters of linear combination are obtained by the simple coefficients, which are used to normalize the input variables of neural network.

The simulation results reveal that the proposed speed regulator keeps motor in constant speed with high convergent speed, but the convergent speed is affected by the sampling frequency. In general, the high sampling frequency will make the speed control system have favor performance, but it will take lots of CPU time. This study applies off-line training to evaluate the appropriate initial values of neural connective weights, then the speed control performance will be improved under low sampling frequency condition.The experiment results reveal the direct neural controller enhances the adaptability and performance, and is available to regulate the speed of DC servo motor with high convergent speed.

II. DESCRIPTION OF THE DIRECT NEURAL CONTROL SYSTEM

The application of the direct neural controller for DC servo motor speed regulation is shown in Fig.1, where ω_r is the speed command and ω is the actual output speed.

Cybenko has shown that one hidden layer with sigmoidal function is sufficient to compute arbitrary decision boundaries for the outputs. Although a network with two hidden layers may give better approximation for some specific problems, de Villiers et al. has demonstrated that networks with two hidden layers are more prone to fall into local minima and take more CPU time. In this study, a network with single hidden layer is applied to the speed regulator. Another consideration is the right number of units in a hidden layer. Lippmann has provided comprehensive geometrical arguments and reasoning to justify why the maximum number of units in a single hidden layer should equal to M(N+1), where M is the number of output units and N is the number of input units. Zhang and Sen. [4] have tested different numbers units of the single hidden layer. It was found that a network with three to five hidden units is often enough to give good results. There are 5 hidden neurons in the proposed neural regulator. The proposed DNC is shown in Fig2 with a three layers neural network.

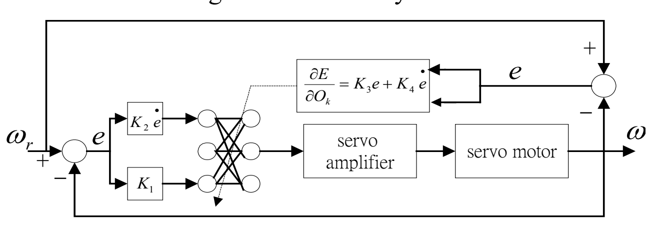


Figure 1. The block diagram of speed control system

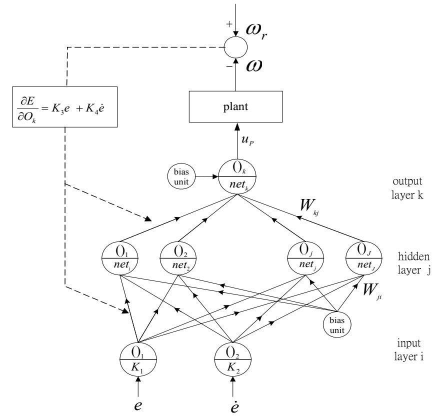


Figure 2. The structure of proposed neural controller

The difference between command speed ω_r and the actual output speed ω is defined as error e. The error e and its differential $\dot{e}$ are normalized between –1 and +1 in the input neurons before feeding to the hidden layer. In this study, the back propagation error term is approximated by the linear combination of error and error's differential. A tangent hyperbolic function is designed as the activation function of the nodes in the output and hidden layers. So that the net output in the output layer is bounded between – 1 and +1, and converted into a bipolar analogous voltage signal through a D/A converter, then amplified by a servo-amplifier for enough current to drive the DC motor. A step speed command is assigned to be the reference command input in order to simulate the step speed response of a DC servo motor.

The proposed three layers neural network, including the hidden layer (j), output layer (k) and input layer (i) as illustrated in Fig. 2. The input signals e and $\dot{e}$ are multiplied by the coefficients K_1 and K_2, respectively, as the normalized signals O_i to hidden neuron. A tangent hyperbolic function is used as the activation function of the nodes in the hidden and output layers. The net input to node j in the hidden layer is

$$net_j = \sum (W_{ji} \cdot O_i) + \theta_j \quad i = 1,2,...I, \quad j = 1,2,...J \tag{1}$$

the output of node j is

$$O_j = f(net_j) = \tanh(\beta \cdot net_j) \tag{2}$$

where $\beta > 0$, the net input to node k in the output layer is

$$net_k = \sum (W_{kj} \cdot O_j) + \theta_k \quad j = 1,2,...J, \quad k = 1,2,...K \tag{3}$$

the output of node k is

$$O_k = f(net_k) = \tanh(\beta \cdot net_k) \tag{4}$$

The output O_k of node k in the output layer is treated as the control input u_P of the system for a single-input and single-output system. As expressed equations, W_{ji} represent the connective weights between the input and hidden layers and W_{kj} represent the connective weights between the hidden and output layers. θ_j and θ_k denote the bias of the hidden and output layers, respectively.

The error energy function at the Nth sampling time is defined as

$$E_N = \frac{1}{2}(X_N - X_{PN})^2 = \frac{1}{2}e_N^2 \tag{5}$$

where X_N and X_{PN} denote the outputs of the reference model and the outputs of the controlled plant at the Nth sampling time, respectively. The weights matrix is then updated during the time interval from N to N+1.

$$\Delta W_N = W_{N+1} - W_N = -\eta\frac{\partial E_N}{\partial W_N} + \alpha\cdot\Delta W_{N-1} \tag{6}$$

where η is denoted as learning rate and α is the momentum parameter. The gradient of E_N with respect to the weights W_{kj} is determined by

$$\frac{\partial E_N}{\partial W_{kj}} = \frac{\partial E_N}{\partial net_k}\frac{\partial net_k}{\partial W_{kj}} = \delta_k O_j \tag{7}$$

and δ_k is defined as

$$\begin{aligned}\delta_k &= \frac{\partial E_N}{\partial net_k} = \sum_n \frac{\partial E_N}{\partial X_P}\frac{\partial X_P}{\partial u_P}\frac{\partial u_P}{\partial O_n}\frac{\partial O_n}{\partial net_k} = \sum_n \frac{\partial E_N}{\partial O_n}\frac{\partial O_n}{\partial net_k} \\ &= \sum_n \frac{\partial E_N}{\partial O_N}\beta(1-O_k^2) \qquad n=1,2,\cdots,K\end{aligned} \tag{8}$$

where $\partial X_P/\partial u_P$ is difficult to be evaluated. The DC servo system is a single-input and single-output control system (i.e., n =1), the sensitivity of E_N with respect to the network output O_k can be approximated by a linear combination of the error and its differential shown as:

$$\frac{\partial E_N}{\partial O_k} = K_3 e + K_4\frac{de}{dt} \tag{9}$$

where K_3 and K_4 are positive constants and assigned as $K_1 = K_3$ and $K_2 = K_4$ in this study, because the proposed neural control system without the reference model, so that the coefficients K_1 and K_2 also can be used as the appropriate coefficients to evaluate the term $\partial E_N/\partial O_k$. Similarly, the gradient of E_N with respect to the weights, W_{ji} is determined by

$$\frac{\partial E_N}{\partial W_{ji}} = \frac{\partial E_N}{\partial net_j}\frac{\partial net_j}{\partial W_{ji}} = \delta_j O_i \tag{10}$$

where

$$\begin{aligned}\delta_j &= \frac{\partial E_N}{\partial net_j} = \sum_m \frac{\partial E_N}{\partial net_k}\frac{\partial net_k}{\partial O_m}\frac{\partial O_m}{\partial net_j} \\ &= \sum \delta_k W_{km}\beta\left(1-O_j^2\right) \qquad m = 1,2,\cdots,J\end{aligned} \tag{11}$$

The weight-change equations on the output layer and the hidden layer are

$$\begin{aligned}\Delta W_{kj,N} &= -\eta\frac{\partial E_N}{\partial W_{kj,N}} + \alpha\cdot\Delta W_{kj,N-1} \\ &= -\eta\delta_k O_j + \alpha\cdot\Delta W_{kj,N-1}\end{aligned} \tag{12}$$

$$\begin{aligned}\Delta W_{ji,N} &= -\eta\frac{\partial E_N}{\partial W_{ji,N}} + \alpha\cdot\Delta W_{ji,N-1} \\ &= -\eta\delta_j O_i + \alpha\cdot\Delta W_{ji,N-1}\end{aligned} \tag{13}$$

where η is denoted as learning rate and α is the momentum parameter δ_j and δ_k can be evaluated from Eq.(11) and (8). The weights matrix are updated during the time interval from N to N+1 :

$$W_{kj,N+1} = W_{kj,N} + \Delta W_{kj,N} \tag{14}$$

$$W_{ji,N+1} = W_{ji,N} + \Delta W_{ji,N} \tag{15}$$

III. DYNAMIC SIMULATIONS

The block diagram of the DC servo motor speed control system with the proposed neural regulator is shown in Fig.1, which consists of a 15W DC servo motor, an tachometer with a unit of 1/150.8 V/rad/s, an 12 bits bipolar D/A converter with an output voltage range of –5V to +5V and a servo amplifier with voltage gain of 2.3. The parameters of DC servo motor are listed in Table 1.

In the designed direct neural controller, the number of neurons is set to be 2, 5 and 1 for the input, hidden and output layers, respectively (see Fig.2). There is only one neuron in the output layer. The output signal of the direct neural controller will be between –1 and +1, which is converted into a bipolar analogous voltage signal by the D/A converter. The output of the D/A converter is between +5V and -5V corresponding to the output signal between +1 and -1 of the neural controller. It means the output of neural controller multiplied by a conversion gain of 5V. Then, the voltage signal is amplified by the servo amplifier to provide high current for driving the DC servo motor. The parameters K_1 and K_2 must be adjusted in order to normalize the input signals for the neural controller. In this simulation, the parameters K_3 and K_4 can be determined, and K_3 = K_1 and $K_4 = K_2$ are assigned.

In this simulations, a step signal of 1V corresponding to 150.8 rad/s is denoted as the speed command, the learning rate η of the neural network is set to be 0.1 and the coefficient β =0.5 is assigned. Since the maximum error-voltage signal is 1V, the parameters K_1 and K_2 are assigned to

TABLE I. TABLE 1. THE PARAMETERS OF MOTOR

Motor resistance R_a	3.18Ω
Motor inductance L_a	$0.53mH$
Inertia of rotor J	$24.3\times10^{-7}kgm^2$
Torque constant K_T	$23\,mNm/A$
Back emf K_B	0.00241V/rpm

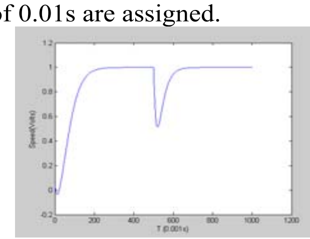
be 0.6 and 0.01, respectively, in order to obtain an appropriate normalized input signals to the neural network. The parameters $K_3 = K_1$ =0.6 and $K_4 = K_2$ =0.01 are assigned for better convergent speed of the neural network. Assumes a disturbance torque load of 0.015 Nm applies to this control system at t=0.5s. The simulation results are shown in Fig.3 (sampling time = 0.001s) and Fig.4 (sampling time = 0.0001s) , where Fig.3 (a) represents the speed response of the DC motor, Fig.3 (b) represents the output signal of the direct neural controller. Fig.4 (a) represents the speed response of the DC motor with neural controller (sampling time = 0.0001s). Fig.4 (b) represents the output signal of the neural controller. Fig.4 (c) shows the convergent time of the connective weights is smaller than 100ms, and the speed response of the DC motor is stable. Consequently, the proposed neural speed regulator enhances the adaptability in speed control system.

In addition, an extra attention should be taken on the disturbing torque load. The neural controller Fig.3 (a) (sampling time = 0.001) does not have fast performance of speed regulation. The neural controller Fig.4 (a) (sampling time = 0.0001) has fast performance of speed regulation. The high sampling frequency will make the speed control system have favor performance, but it will take lots of CPU time. This simulation applies off-line training to evaluate the appropriate initial values of neural connective weights, and then the speed control performance will be improved under low sampling frequency condition. By this method, the simulation results with sampling time =0.001s shown in Fig.5. Fig.5 (a) shows the speed response of the DC motor with neural controller (sampling time = 0.001s). Fig.5 (b) shows the output signal of the neural controller. Fig.5 (c) shows the convergent time of the connective weights.

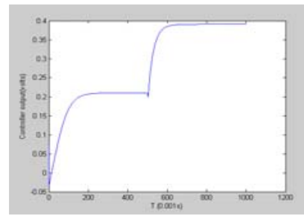
IV. EXPERIMENTAL RESULTS

The experimental apparatus consist of a 15W DC servo motor whose parameters shown in Table 1 are the same as that of simulation, an encoder with a unit of 0.01256 rad/pulse and a servo amplifier with voltage gain of 2.3.

In the designed direct neural controller, the number of neurons is set to be 2, 5 and 1 for the input, hidden and output layers, respectively (see Fig.2). There is only one neuron in the output layer. The output signal of the direct neural controller will be between –1 and +1, which is converted into a bipolar analogous voltage signal by the D/A converter. The output of the D/A converter is between +4.847V and -4.847V corresponding to the output signal between +1 and -1 of the neural controller. Then, the voltage signal is amplified by a servo-amplifier, and which provides sufficient current to drive the DC servomotor. In this experiment, the parameters $K_3 = K_1$ and $K_4 = K_2$ is assigned. A step signal of 120 pulses/0.01s corresponding to 150.8 rad/s is denoted as the speed command. The learning rate η =0.5, the coefficient β =0.5 and the sampling time of 0.01s are assigned.

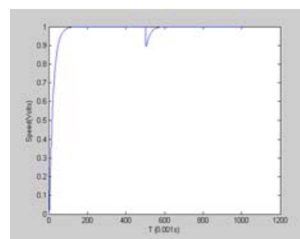


(a) Speed response

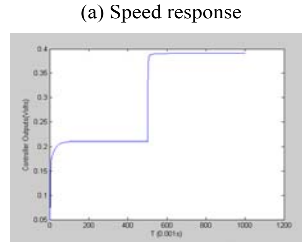


(b) Output of neural controller

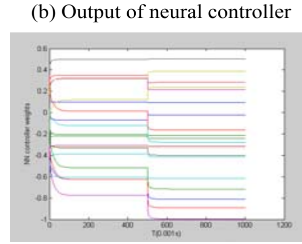
Figure 3. Simulation results for speed control (T=0.001S, torque load of 0.015 Nm applies to this control system at t=0.5s)



(a) Speed response

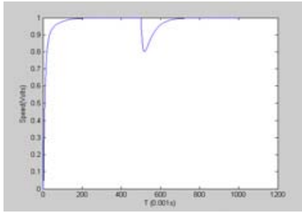


(b) Output of neural controller

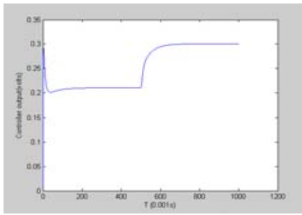


(c) The time responses for connective weights

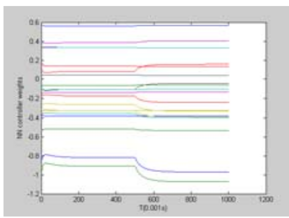
Figure 4.Simulation results for speed control (T=0.0001S, torque load of 0.015 Nm applies to this control system at t=0.5s)



(a) Speed response



(b) Output of neural controller



(c) The time responses for connective weights

Figure 5. Simulation results for speed control (T=0.001s) with the appropriate initial values of neural connective weights by off-line training

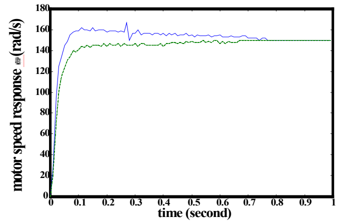
The parameters $K_3 = K_1$=0.003 are defined to normalize the input signals of the neural controller, and $K_4 = K_2$=0.00003 (0.00004) are assigned to provide appropriate damping effect. This experiment applies off-line training to evaluate the appropriate initial values of neural connective weights, and then the speed control performance will be improved under low sampling frequency condition (sampling time=0.01s). The experiment results are shown in Fig.6. It shows that the speed response of a DC motor is stable and accurate but with some overshoot.

Figure 6.shows the parameters K_2 and K_4 exhibit a damping effect.. DC motor speed response with $K_2 = K_4$=0.00003 has more overshoot than that with $K_2 = K_4$=0.00004. The neural control system with $K_2 = K_4$=0.00004 has better damping.

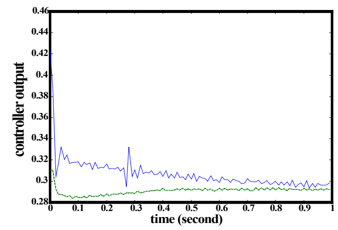
It is a better way for neural controller to apply smaller sampling time, but it will take lots of CPU time. This experiment applies off-line training to evaluate the appropriate initial values of neural connective weights, then the speed control performance will be improved under low sampling frequency condition.

V. CONCLUSION

The proposed direct direct neural speed control is easily implemented, which has been applied to regulate the speed of a DC servomotor successfully. The advantages of this controller are no need of previous knowledge or dynamic model of the plant. The on line learning capability of the



(a) Speed response of DC servo motor



(b) The output of neural controller

Figure 6. Experiment results (Sampling time=0.01s, $K_1 = K_3 = 0.003, \eta = 0.5, K_2 = K_4 = 0.00003$: —, $K_2 = K_4 = 0.00004$: ----)

proposed direct direct neural depends on high sampling frequency. Although the high sampling frequency will make the speed control system have favor performance, but it will take lots of CPU time. This study applies off-line training to evaluate the appropriate initial values of neural connective weights, and then the speed control performance will be improved under low sampling frequency condition.

[1] R. S. Ahmed, K. S. Rattan, and I. H. Khalifa, "Real-Time Tracking Control of A DC Motor Using A Neural Network," IEEE Aerospace and Electronics Conference, Vol. 2, pp.593-600, 1995.

[2] M. A. Hoque, M. R. Zaman, and M.A. Rahman, "Artificial Neural Network Based Controller For Permanent Magnet DC Motor Drives," IEEE Industry Application Conference, Thirtieth IAS Annual Meeting, Vol. 2, pp.1775-1780, 1995.

[3] F. M. EI-Khouly, A. S. Abdel-Ghaffar, A. A. Mohammed and A. M. Sharaf, "Artificial Intelligent Speed Control Strategies for Permanent Magnet DC Motor Drives," IEEE Industry Applications Conference, IAS Annual Meeting, Vol. 1, pp.379-385, 1994.

[4] Y. Zhang, P. Sen, and G. E. Hearn, "An On-line Trained Adaptive Neural Controller," IEEE Control Systems Magazine, Vol. 15, No. 5, pp. 67-75, 1995.

[5] F. J. Lin, R. J. Wai, and H. P. Chen, "A PM synchronous servo motor drive with an on-line trained fuzzy neural-network controller," IEEE Translate Energy Conversion, Vol. 13, pp. 319-325, 1998.

[6] F. J. Lin, and R. J. Wai, "Hybrid Controller Using Neural Network for PM Synchronous Servo Motor Drive," IEE Proceeding of Electric Power Application, Vol. 145, No. 3, pp. 223-230, 1998.

[7] F. J. Lin, R. J. Wai, and C. C. Lee, "Fuzzy Neural Network Position Controller for Ultrasonic Motor Drive Using Push-pull DC-DC Converter," IEE Proceeding of Control Theory Application., Vol. 146, No. 1, pp. 99-107, 1999.

[8] M. H. Chu, Y. Kang, Y. F. Chang, Y. L. Liu and C. W. Chang, "Model-Following Controller Based on Neural Network for Variable Displacement Pump," JSME International Journal, Series C, Vol. 46, No. 1, pp.176-187, 2003.

[9] Y. Kang, M.-H. Chu, Y-L Liu, C.-W. Chang and Shu-Yen Chien, "An Adaptive Control Using Multiple Neural Networks for the Position Control in Hydraulic Servo System", ICNC2005, LNCS3611, pp.291-300, 2005.