Remarks on tracking method of neural network weight change for adaptive type neural network feedforward feedback controller

Takayuki Yamada
Department of Computer and Information Sciences
Faculty of Engineering
Ibaraki University
4-12-1 Nakanarusawa, Hitachi, Ibaraki, 316-8511, Japan

Abstract

A cost function is useful for a confirmation of neural network controller learning performance, but, this confirmation may not be correct for neural networks. Previous papers proposed a tracking method of neural network weight change and simulated it on the application of both learning and adaptive type neural network direct controllers. This paper applies the tracking method to an adaptive type neural network feedforward feedback controller and simulates it. The simulation results confirm that a track of the neural network weight change is separated into two trajectories. They also discuss the relationship between the feedback gain of the feedback controller and the parameter determining the neural network learning speed.

1. Introduction

Many studies have been undertaken in order to apply both the flexibility and the learning capability of neural networks to control systems.[1]-[4] Learning rules of neural network weights are usually designed so as to minimize the error between a plant output (or neural network output) and a desired output (teaching signal). The essence of neural network learning is nothing but the change of the neural network weights. However, in order to examine the performance of the neural network learning, most researchers use a cost function (squared error between the desired output and the neural network output (or the plant output)). This is because it is not practical to examine the huge number of the neural network weights and the cost function is a scalar value which is easily dealt with. However, the neural network weight change is not always reflected in the cost function. This problem is especially serious in neural network controller applications. This is because the performance of the cost function is affected by dynamics of the plant. My previous paper[1] proposed a tracking method of the neural network weight change as an examination of the neural network controller learning performance. This tracking method can realize to observe the neural network weight change directly on the 2D plane. It was also applied to both a learning type and an adaptive type neural network direct controllers and its usefulness was confirmed.[1][2] The neural network direct controller is simplest because the neural network output is directly connected to the plant input. This is suitable to examine basic characteristics, but too simple. On the other hand, a feedforward feedback controller with the neural network was also proposed.[4] The plant input is the sum of the neural network output and the feedback controller output in this feedforward feedback controller. It is expected that this type neural network controller is more practical in comparison with the direct controller. This is because it is based on a biological neural network. However, its learning performance is different from that of the direct controller. This is because the learning rule of the feedforward feedback controller is designed so as to minimize the feedback controller output.

Thus, this paper applies the tracking method of the neural network weight change to the adaptive type neural network feedforward feedback controller. A second order discrete time SISO (single input and single output) plant is selected as an example of dynamical plants. The simulation results confirm the usefulness of the tracking method for the adaptive type neural network feedforward feedback controller. They also confirm that it is observed the neural network weight change trajectories are separated on the 2D plane. This phenomena is observed when the tracking method is applied to the adaptive type neural network direct controller.[2] The feedfoward feedback controller has a feedback gain and it takes effect on the neural network learning performance. Simulation results confirm this effect and its relationship to the parameter determining the neural network learning speed on the 2D plane..

2. Tracking method of neural network weight change

This section explains the tracking method of the neural network weight change briefly. This tracking method is applied to the adaptive type neural network feedforward feedback controller for the SISO plant. In this paper, an output layer of the neural network has one neuron, the weights between the output layer and a hidden layer can be expressed as a vector ω and the weights between the hidden layer and an input layer can be expressed as a matrix W. To simplify, the neuron number of the input layer is equal to that of the hidden layer. That is, the weight matrix W is the square matrix. The tracking method uses the following steps.

(Tracking method of neural network weight change)
(1) We can derive one weight vector Γ from the neural network weight vector ω and weight matrix W as follows:

$$\Gamma^T = [\omega_1 \cdots \omega_n \; W_{11} \cdots W_{1n} W_{21} \cdots W_{2n} \cdots W_{n1} \cdots W_{nn}] \quad (1)$$

Where n is the neuron number both the input layer and the hidden layer.

(2)We must define a standard vector Γ_0. Any vector, which has same order as that of the weight vector Γ, can be selected as this standard vector, for example, the weight vectors derived from the initial neural network weights, the final neural network weights and so on.

(3) We can calculate an inner product of the weight vector Γ and the standard vector Γ_0 because these vectors have same order. We can also calculate an angle between the weight vector Γ and the standard vector Γ_0 as follows:

$$X = |\Gamma| \cos\theta, \quad Y = |\Gamma| \sin\theta \quad (2)$$

$$\theta = \cos^{-1}\left(\frac{\langle \Gamma_0 \cdot \Gamma \rangle}{|\Gamma_0| \cdot |\Gamma|}\right) \quad (3)$$

Where $\langle \Gamma_0 \cdot \Gamma \rangle$ is the inner product between the vector Γ_0 and the vector Γ, and $|\Gamma|$ is the norm of the vector Γ.

(4) We can draw a new weight performance on the 2D plane by the use of X and Y in equations (2) and (3).

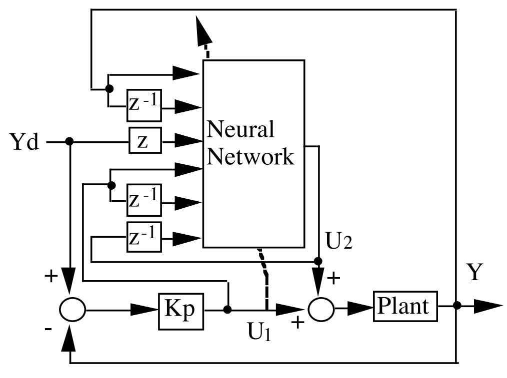

Fig.1 Block diagram of neural network feedforward feedback controller for second order discrete time plant.

3. Simulation

This paper applies the tracking method of the neural network weight change to the adaptive type neural network feedforward feedback controller. The simulated plant is follows:

$$Y(k) = -a_1 Y(k-1) - a_2 Y(k-2) + U(k-1) + bU(k-2) - a_3 Y(k-3) + C_{non} Y^2(k-1) \quad (4)$$

Where Y(k) is the plant output, U(k) is the plant input, k is the sampling number, a_1, a_2 & b are the plant parameters, a_3 is the parasite term and C_{non} is the nonlinear term. For this simulation, a_1=-1.3, a_2=0.3, b=0.7, a_3=-0.03 and C_{non}=0.2 are selected. The rectangular wave is also selected as the desired value Yd. The output error ε is defined as follows:

$$\varepsilon(k) = Yd(k) - Y(k) \quad (5)$$

For this simulated plant, the neuron number n in both the input and hidden layers is 6. The neural network input vector I is defined as the following equation.

$$I^T(k) = [Y_d(k+1) \; Y(k) \; Y(k-1) \; U_2(k-1) \; U_1(k) \; U_1(k-1)] \quad (6)$$

Where U1(k) and U2(k) are the feedback loop output and the neural network output respectively. We select the following sigmoid function f(x) as the input output relation of the hidden layer of the neural network.

$$f(x) = \frac{X_g\{1 - \exp(-4x/X_g)\}}{2\{1 + \exp(-4x/X_g)\}} \quad (7)$$

Where Xg is the parameter which defines the sigmoid function shape. The neural network output U2(k) is composed as follows:

$$U_2(k) = \omega^T(k) f\{W(k) I(k)\} \quad (8)$$

When we use the P control (Proportional control), the feedback loop output U1(k) is composed using the feedback gain Kp as shown in the following equation.

$$U_1(k) = Kp\{Yd(k) - Y(k)\} \quad (9)$$

The plant input U(k) of the feedforward feedback neural network controller is the sum of the neural network output and the feedback loop output as follows:

$$U(k) = U_1(k) + U_2(k) \quad (10)$$

The block diagram of the adaptive type neural network feedforward feedback controller is shown in Fig.1. The learning rule of this neural network controller is designed so as to minimize the feedback loop output. When we apply the δ rule to this learning rule, it is expressed as follows:

$$W_{ij}(k+1) = W_{ij}(k-1)+\eta U_1(k)\omega_i(k-1)I_j(k-1) \times f'\{\sum_{j=1}^{n} W_{ij}(k-1)I_j(k-1)\} \quad (11)$$

$$\omega_i(k+1) = \omega_i(k-1)+\eta U_1(k)f\{\sum_{j=1}^{n} W_{ij}(k-1)I_j(k-1)\} \quad (12)$$

Where η is the parameter to determine the neural network learning speed. We select the weight vector derived from the initial neural network weights as the standard vector Γ_0 of the equations (2) and (3)

Figures 2 and 3 respectively show the plant output k=1~1000 and k=4000~5000. As shown in these figures,

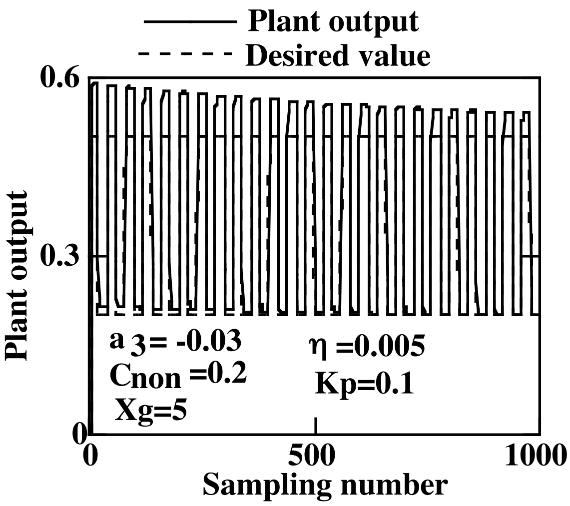

Fig.2 Plant output (k=1~1000).

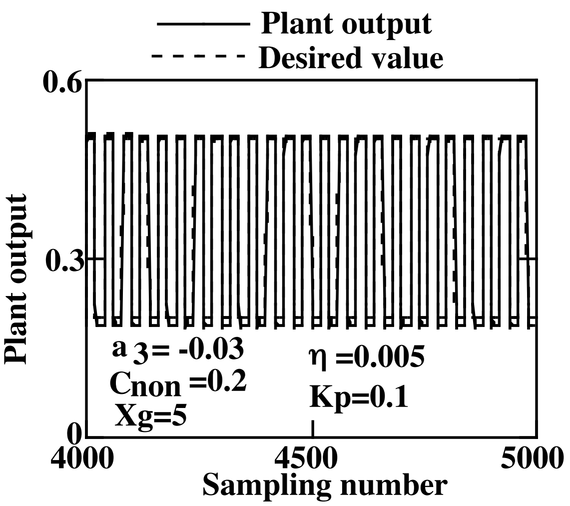

Fig.3 Plant output (k=4000~5000).

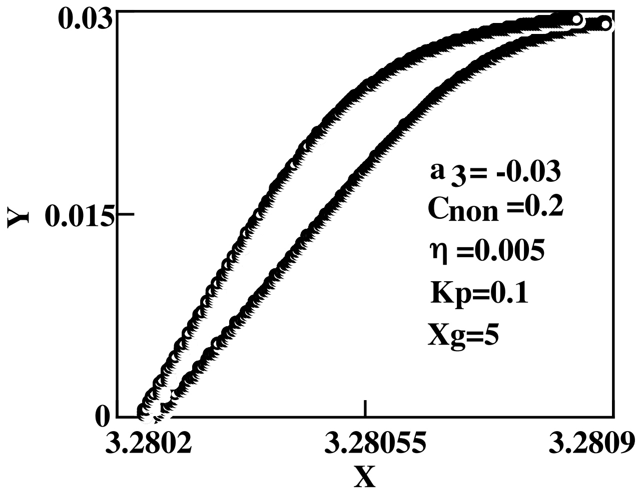

Fig.4 Track of neural network weight change.

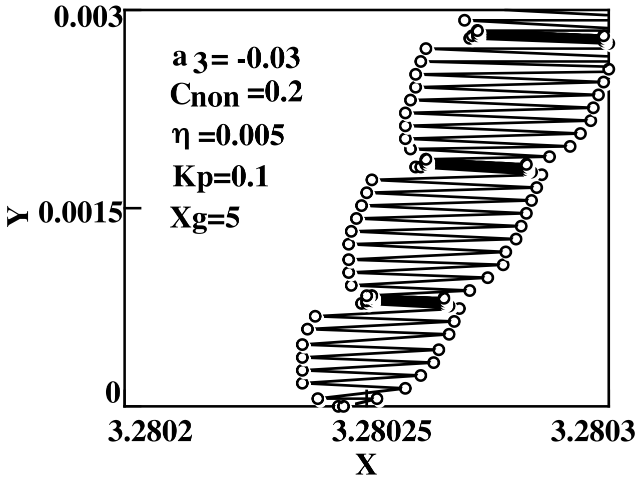

Fig.5 Expansion of track around initial stage of learning.

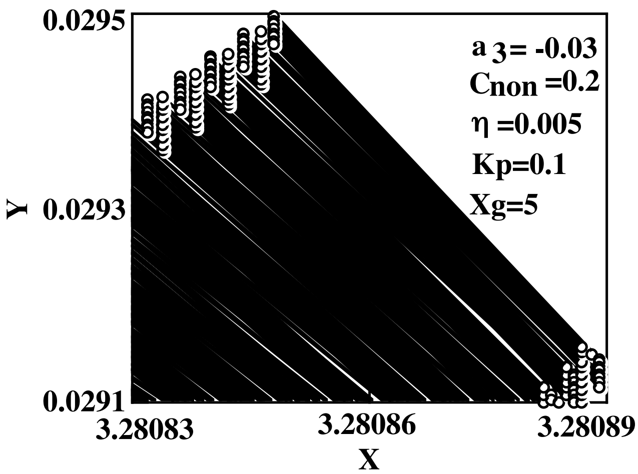

Fig.6 Expansion of track around final stage of learning.

the neural network output decreases as learning progresses and it converges with the desired value. Figure 4 shows the whole track of the neural network weights change (k=1~5000). As shown here, it starts on the X axis and after, it separates to two trajectories. Figure 5 shows the expansion of the track around the initial stage of learning. As shown in this figure, the start point is one, but, soon the track of the neural network weight change is separate to two trajectories. The length between two trajectories seems to be periodic. Figure 6 shows the expansion of the track around the final stage of leaning. The trajectories are still separated and periodic. This phenomena was also observed in the adaptive type neural network direct controller application.

Next, We examine the relationship between the feedback gain Kp and the parameter η determining the neural network learning speed. Figures 7, 8 and 9 respectively show the tracks of (η=0.001, Kp=0.01), (η=0.0001, Kp=0.1) and (η=0.00001, Kp=1). As shown in these figures, fig.7 is almost same to fig.8. This fact shows that the feedback gain Kp has same capability to that of the parameter η for the convergence of the neural network weights. However, the track of fig.9 is different from those of figs.7 and 8. That is, the feedback gain Kp has the above capability in some restricted region.

4. Conclusion

This paper applied the tracking method of the neural network weight change to the adaptive type neural network feedforward feedback controller. The simulation results confirmed its usefulness. They also clarified that the feedback gain has same capability to that of the parameter determining the neural network learning speed in some restricted region.

Acknowledgment

The author wishes to express his thanks to Mr.Takamitsu Nakazaki, graduated student, Ibaraki University, for his programming and simulation.

References

[1] Takayuki Yamada, "Remarks on tracking method of neural network weight change for learning type neural network direct controller", Proceedings of AROB 10th '05 (The Tenth International Symposium on Artificial Life and Robotics 2005), pp.624-627(2005)

[2] Takayuki Yamada, "Remarks on tracking method of neural network weight change for adaptive type neural network direct controller", Proceedings of AROB 11th '06 (The Eleventh International Symposium on Artificial Life and Robotics 2006), GS19-3 (2006)

[3]K.S.Narendra and K.Parthictsarathy, "Identification and Control of Dynamics System Using Neural Networks", IEEE Transaction on Neural Networks, Vol.1, No.1, pp.4-27, 1990

[4] Takayuki Yamada and Tetsuro Yabuta, "Adaptive Type Feedforward Feedback Controller Using Neural Networks", Transactions of the Society of Instrument and Control Engineers, Vol.30, No.10, pp.1234-1241(1994)(in Japanese)

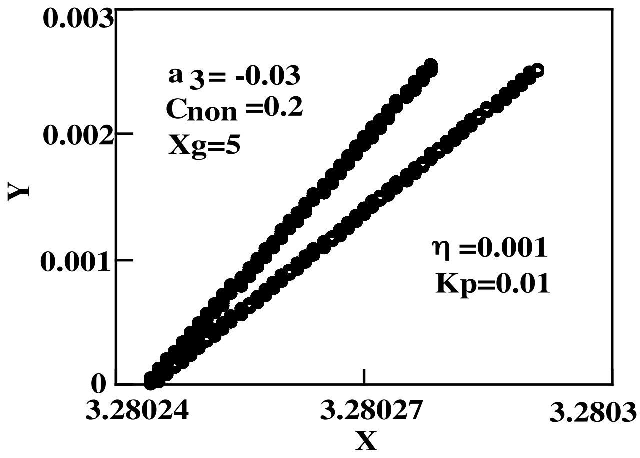

Fig.7 Track (η=0.001, Kp=0.01).

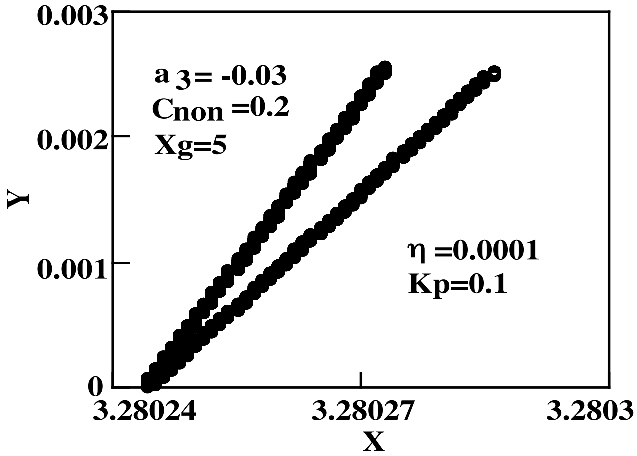

Fig.8 Track (η=0.0001, Kp=0.1).

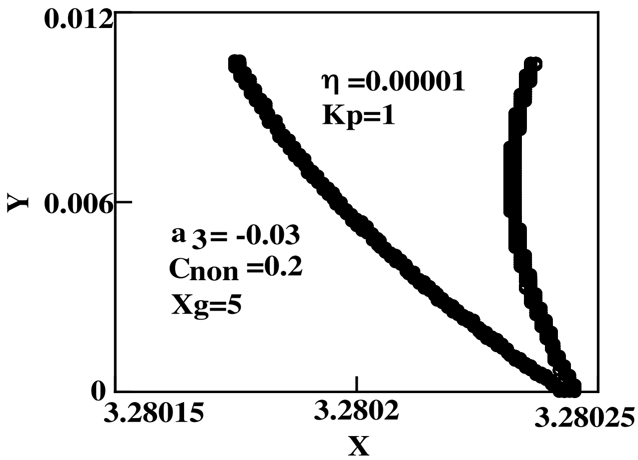

Fig.9 Track (η=0.00001, Kp=1).