'89 KACC 1989. 10. 27~28

Neural Network for Servo Control System

Hideki Hashimoto*, °Junichi Endo**, and Fumio Harashima*

*Institute of Industrial Science, University of Tokyo, **Hitachi, Ltd.
7-22-1, Roppongi, Minato-ku, Tokyo 106, JAPAN
(TEL) 03-402-6231 (FAX) 03-423-1484

Abstract: In this paper, the inverse model of a servo system is realized in a PDP-type neural network. The neural network learns the mapping between the input and output of the servo system. Some simulation results show the effectiveness of this inverse model obtained here.

1. Introduction

Many methods are proposed to construct an inverse model of the plant and apply it to adaptive and iterative control system[1]. However, the structure of the plant must be known in advance to apply those methods, and that most of the constructed inverse model represent only a part of the original plant.

Authors of this paper have proposed a method[2] based on neural network to construct an inverse model of the dynamical plant which structure is unknown. In this paper, a method is presented to construct an inverse model of the servo system by using PDP type neural network. It is confirmed that the neural network is able to learn the relationship between inputs and outputs of the system.

The servo control system with a neural network as a feedforward model can be completed by introducing a feedback loop to compensate the error due to the inaccuracy of the inverse model.

2. Control Strategy

Figure 1 shows a typical inverse system when a perfect inverse model of the plant is obtained. The plant dynamics is represented by G(s) and the inverse system is represented by $G^{-1}(s)$ if there exists an inverse model. The transfer function of the combined control system becomes 1, and then the output of the system can be controlled freely. However, if the inverse model (whose transfer function is represented by $G^{-1*}(s)$) is obtained with some error, it is required to introduce a feedback mechanism to compensate the error due to the inaccuracy of the inverse model as shown in Fig.2. In Fig.3, supposing that $G^{-1*}(s)$ can be divided into $G^{-1}(s)$ and $\Delta G^{-1}(s)$, it is reasonable to compensate $\Delta G^{-1}(s)$ by the feedback mechanism. Since the error due to $\Delta G^{-1}(s)$ is not large compared with that of the system without $G^{-1*}(s)$, it is easy to design the feedback control system.

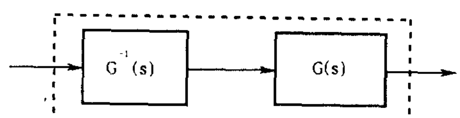

Fig.1 Complete inverse system and control systm

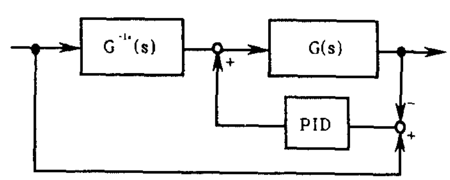

Fig.2 Compensation for incompletion of inverse model

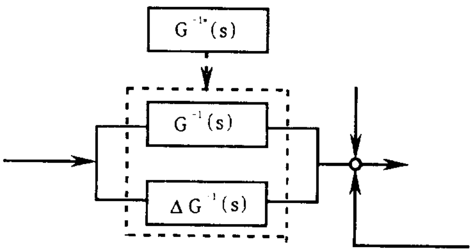

Fig.3 Incomplete inverse model

3. Representation of the plant dynamics using Neural Network

The neural network used in this paper is a hierarchical network with three layers as shown in Fig.4. The learning algorithm in this network is "Back-Propagation"[3]. This type of network can express only the static relationship between inputs and outputs, and cannot directly represent dynamical relationship. When the plant can be represented as the ARMA model, it is possible to represent the plant dynamics by inputting some time series data of the inputs and outputs of the plant to the neural network in parallel[4].

The problem of this method is that representable dynamics is dependent on the number of the available time series data, and that the more the number of the available data increases, the more the number of the link extremely increases. When the system structure is perfectly unknown, the available value is only input time series u and output time series y, so only ARMA model is used. On the other hand, in case of the system where a part of the structure is known in advance, the state vector is available, it is better to use it than time series data. For example, when all of the elements of the state vector x[k] in equation (1) are available, x[k+1] is determined by x[k] and u[k]. Therefore the values of x[k] and x[k-1] are sufficient for calculating u[k-1].

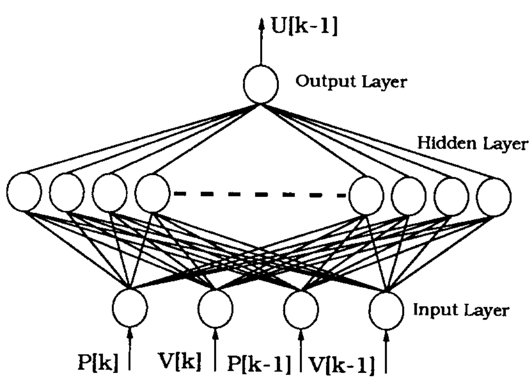

Fig.4 Parallel distributed processing (PDP)

$$x[k+1] = F\,x[k] + G\,u[k] \quad (1)$$
$$y[k] = H\,x[k] \quad (2)$$

In this paper, inverse system of the servo system is constructed on the assumption that both position and velocity are measurable. The inverse system is constructed by using the neural network as shown in Fig.5.

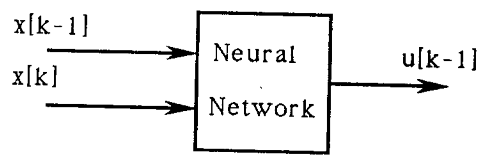

Fig.5 Case that stare variables are available

4. Simulation Study

Simulation results are shown in this section. The servo system is a first order system with an integrator as shown in Fig.6. The system is discretized by a sampling period T (=1 msec), and the parameter a is 100 in the simulation.

The procedure of the simulation consists of three steps as follows.

<step 1>
Give the appropriate inputs sequence u to the system, and save the state vector x at that time into the memory.

<step 2>
The neural network learns the relation between inputs x(k-1), x(k) and output u(k-1).

<step 3>
Construct the neural network control system as shown in Fig.7.

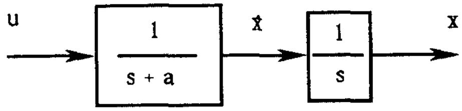

Fig.6 Servo system

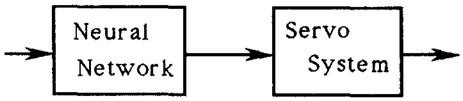

Fig.7 Feedforward control system

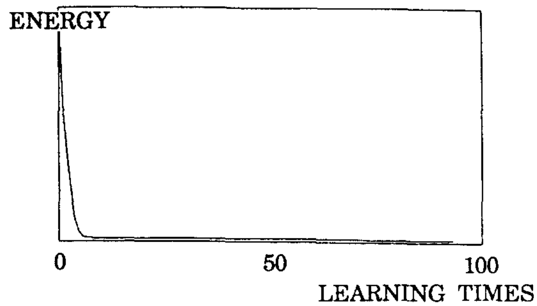

Fig.9 Behavior of energy

[1] Learning

Figure 8 shows the learning pattern. Fig.8(a), (b) and (c) show velocity, position and input pattern, respectively. A behavior of error energy in this learning is shown in Fig.9. Error energy is smaller enough after 20 times learnings. However, error doesn't decrease any more. In this simulation the learning is stopped at 100 times learning.

[2] Time Responses to reference

After learning, the feedforward control system shown in Fig.7 is used to investigate the capability of the neural network. The references are shown in Fig.10 where (a) is velocity, (b) is position and (c) is an ideal input pattern.

Responses to these references are shown in Fig.11. The input, velocity and position time responses have some errors, but the responses reflect the behavior of the references.

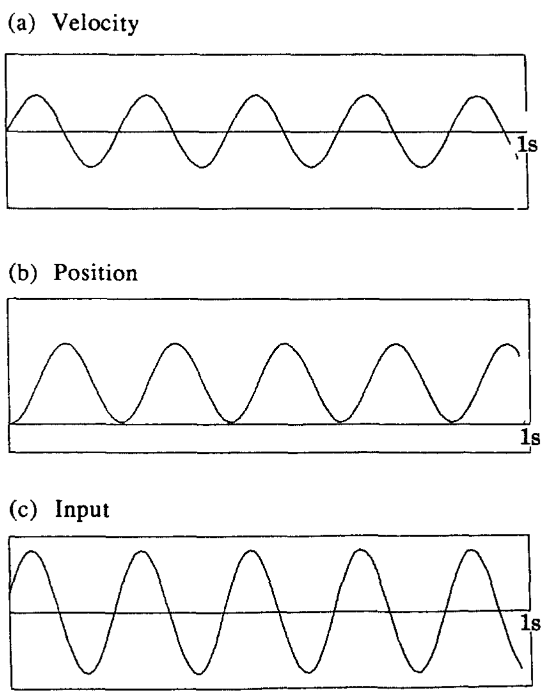

Fig.8 Learning pattern

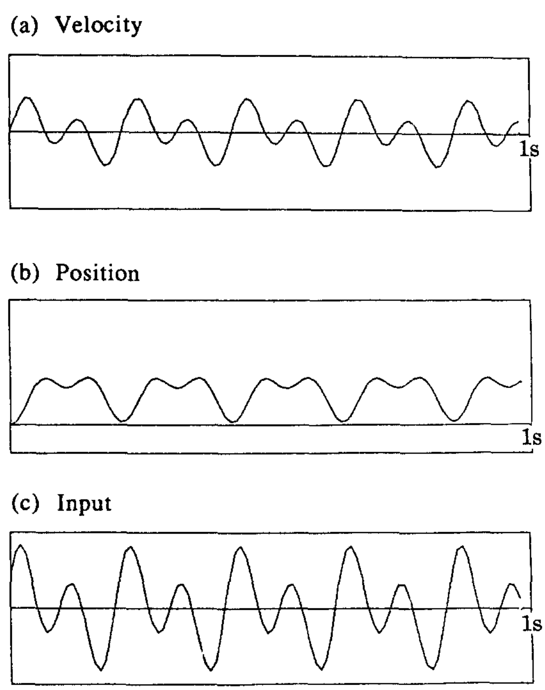

Fig.10 Reference

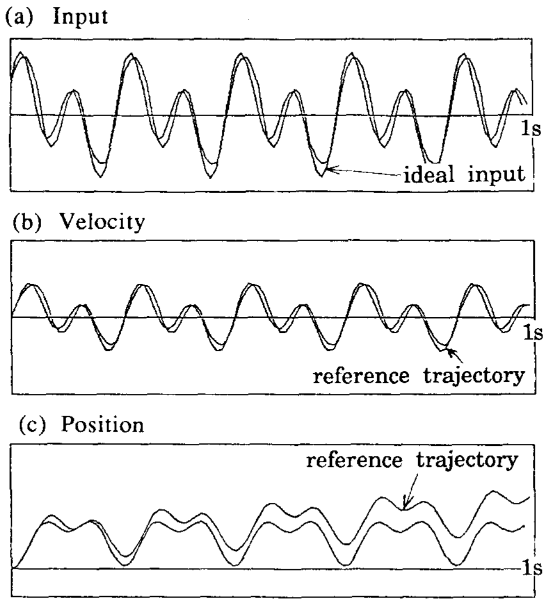

Fig.11 Response to reference

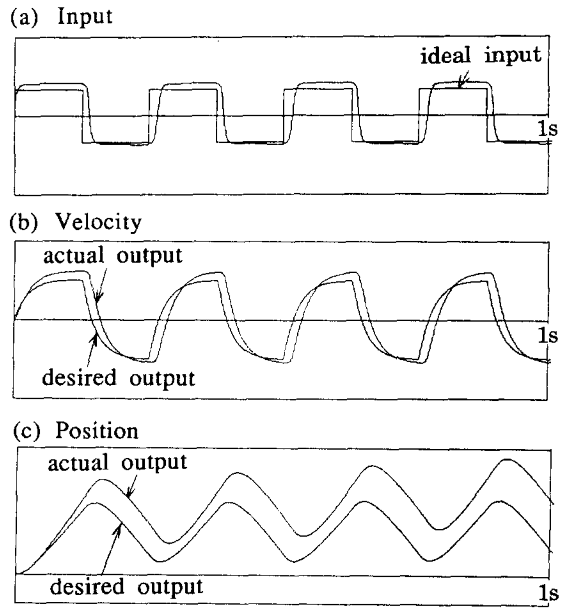

Fig.13 Reference and response

Another simulation result is shown in Fig.12 and Fig.13. Here square wave is used as learning pattern as shown in Fig.12. Figure 13 also shows that the responses reflect the behavior of the references though there are some errors.

From those simulation results, the network has also higher degree of mode (frequency), though the neural network learns only one reference.

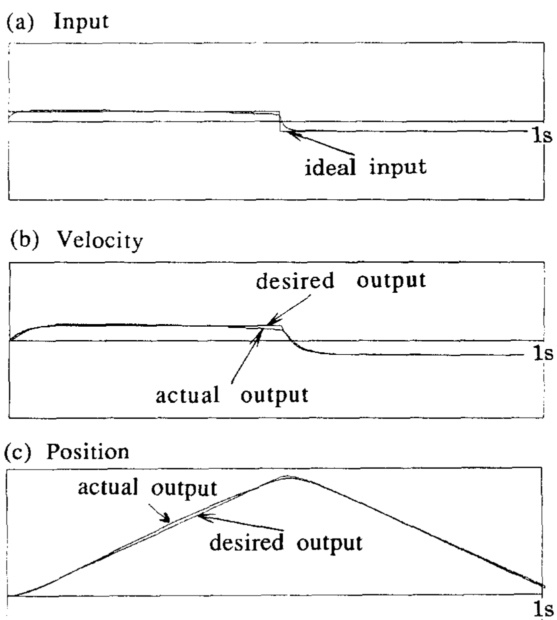

Fig.12 Learning pattern and response

5. Conclusion

It is confirmed by the simulation results that the neural network which learned only low frequency can express higher ones. It is the special feature that the error in low frequency is stressed while. A usual feedback can subdue the error in low frequency and an inverse model constructed by an typical iterative learning cannot express in high frequency. Therefore it is possible to get a good response by using the neural network in high frequency and a feedback loop in low frequency.

6. References

[1] S.Kawamura, F.Miyazaki, and S.Arimoto, "System Theoretic Study on Learning Control Method," Transactions of the Society of Instrument and Control Engineers, Vol.21, No.5, pp.445-450, 1985

[2] H.Hashimoto, H.Noda, J.Endo, and F.Harashima, "Application of Neural Network to Servo System," Proc. of the 28th SICE Annual Conference, pp.393-394, 1989

[3] D.E.Rumelhart, J.L.McClelland, the PDP Research Group, "Parallel Distributed Processing," Bradford Books, Vol.1&2, MIT Press, 1986

[4] K.Shibata, M.Inaba, and H.Inoue, "Learning of robot's movement by neural network," Proc. of the sixth RSJ Annual Conference, pp.141-142, 1988