신경회로망을 이용한 동적 시스템의 자기동조 제어기 설계

조현섭*, 오명관**
청운대학교 디지털방송공학과, 혜전대학 디지털서비스과
e-mail: chohs@chungwoon.ac.kr

Design of auto-tuning controller for Dynamic Systems using neural networks

Hyun-Seob Cho*, Myoung Kwan Oh**
*Dept of Digital Broadcast Engineering Chungwoon University
**Dept of Digital Service Hyejeon College

요 약

"Dynamic Neural Unit"(DNU) based upon the topology of a reverberating circuit in a neuronal pool of the central nervous system. In this thesis, we present a genetic DNU-control scheme for unknown nonlinear systems. Our methodis different from those using supervised learning algorithms, such as the backpropagation (BP) algorithm, that needs training information in each step. The contributions of this thesis are the new approach to constructing neural network architecture and its trainin.

1. Introduction

In practical, many control problems suffer from difficulty due to system nonlinearity, uncertainty, and dynamic property. The existing control methods are system specific; i.e., a control methodology designed for a type of nonlinear system may not be suitable for other types of nonlinear systems, while neural networks can be approximate arbitrary nonlinear functions. A neuro-control system, in general, performs a specific form of adaptive control, with the controller taking the form of multilayer neural network and the adaptable parameters being defined as the adjustable connection weights which could be better adapt to different plant and environmental conditions.

2. The Dynamic Neural Units (DNUs)

The dynamic neural unit (DNU), whose topology embodies delay elements, feedforward and feedback synaptic weights, and a nonlinear operator has been used successfully for controlling unknown linear and nonlinear dynamic systems by considering only the synaptic weights as the adjustable parameters of the network.

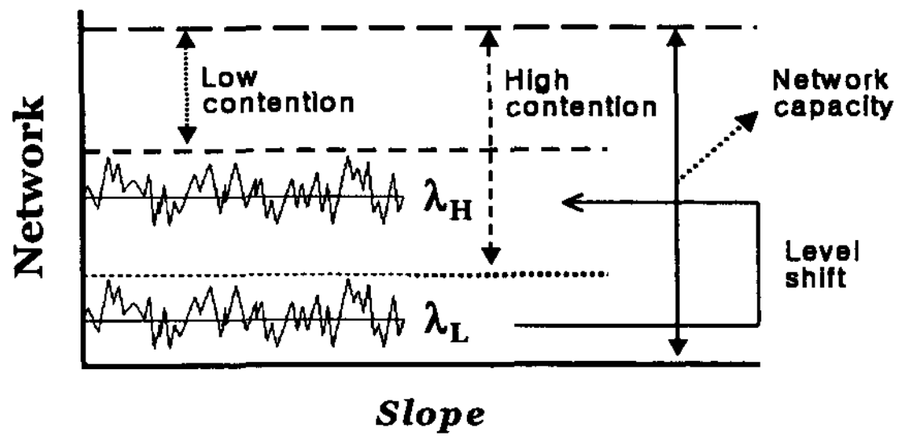

Fig.1, Symbolic representation of DNU.

The difference equation that describes the behavior of the second order dynamical structure is given in Eqn.(1) in which v 1 (k), x (k) ÅR1. Similarly, the pulse transfer function of this part can be given by Eqn.(2).

$$v_1(k) = -b_1 v_1(k-1) - b_2 v_1(k-2) + a_0 x(k) + a_1 x(k-1) + a_2 x(k-2) \quad (1)$$

$$T(z) = \frac{a_0 + a_1 z^{-1} + a_2 z^{-2}}{1 + b_1 z^{-1} + b_2 z^{-2}} \quad (2)$$

The output of the DNU can be evaluated as follows;

$$v(k) = g_s v_1(k) \quad (3)$$

$$u(k) = \Psi(v(k)) = \tanh(g_s v_1(k)) \quad (4)$$

The objective is based on the minimization of the instantaneous error evaluated at the output of the system. The cost function that is to be minimized is defined by Eqn.(6) in which E denotes the expectation operator. If w denotes the parameters of a single DNU, the update rule can be given by;

$$w_{k+1} = w_k - \mu \frac{\partial J}{\partial w} \quad (5) \quad ; \quad J = \tfrac{1}{2} E(e^2(k)) \quad (6)$$

$$e(k) = y_d(k) - u(k) \quad (7)$$

More compactly, the parameter update rule is given by Eqns (8) (9) (10) where i = 0,1,2 and j = 1,2 ;

$$\Delta a_i(k+1) = u_{a_i} E(e(k) g_s^2(k) \operatorname{sech}^2[v_1(k)] x(k-i)) \quad (8)$$

$$\Delta b_j(k+1) = u_{b_j} E(e(k) g_s^2(k) \operatorname{sech}^2[v_1(k)] v_1(k-j)) \quad (9)$$

$$\Delta g_s(k+1) = g_s(k) u_{g_s} E(e(k) \operatorname{sech}^2[v_1(k)] v_1(k)) \quad (10)$$

In the parameter update equations, the coefficients u_{a_i}, u_{b_j} and u_{g_s}, denote the step size for the corresponding parameter and are chosen to be constant throughout a simulation.

3. The Dynamic Neural Network (DNN) control model

In Fig.2 DNU layer includes the desired number of individual DNU blocks whose inputs areconnected together and whose outputs are added to form the control u(k). Depending on the magnitude of the output error, the algorithm updates the feedforward and feedback weights and the gain of the nonlinear activation functions of each dynamical neural unit in the DNU layer. The derivation of the algorithm is given in [1] in detail.

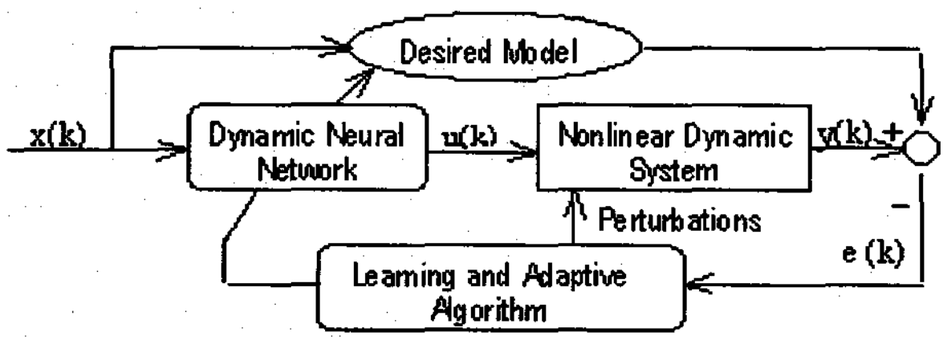

Figure 2. Control scheme for a nonlinear dynamic system using dynamic neural network

In our example, we adopt a three-stage dynamic neural network, an input stage, an intermediate stage, and an output stage is formed. In each stage there are two DNUs with a weight connect to next stage in each DNU. The DNN structure is showed in fig.3

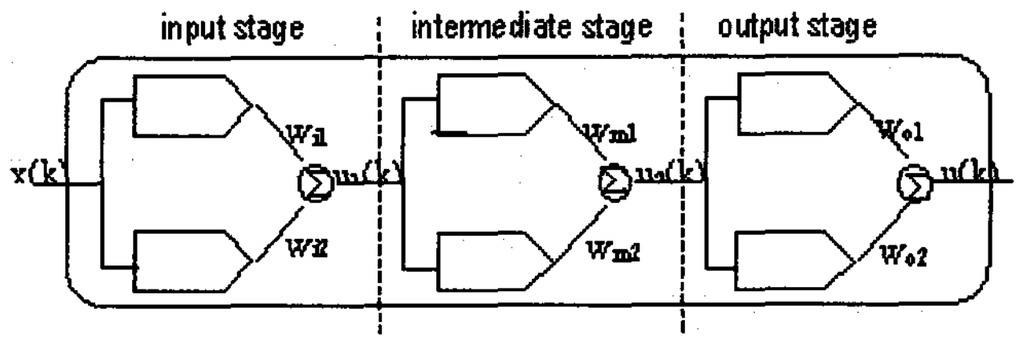

Fig 3. the Dynamic Neural Network structure

4. Applied Genetic Algorithm to adjust the parameters of the DNN

Genetic algorithms have been utilized for many control problems. P. Wang et al. presented a numerical example in which a pH (potential of hydrogen) neutralization process is regulated by a PID controller with its parameters optimized using simple GA[2]; A. Varek et al. employed GA to derive control rules[3] encoded as decision tables and to optimize the parameters of the induced rules. Some researchers also studied genetically optimized fuzzy logic control. In their study, GAs played a role of optimizing membership functions and fuzzy linguistic rule sets. W. Schiffmann et al. introduced GAs to optimize the BP algorithm for training multilayer neural networks.

5. Tuning of the DNN Parameters by GA

The steps optimizing parameters of DNN by GA is as follows:

Coding strategy of the DNN parameters: a total of 42 parameters (6DNUs*6variables+6weights) are needed to be tuned by GA. Each of the parameters is encoded into 8_bit string, resulted in a complete chromosome of 336bits.

Initial populations comprise a set of chromosome: in this experiments, the population consists of 200 chromosomes which are all randomized initially.

Optimization by GA: the process is shown in fig.4:

The terminal condition of the GA optimization process is to determine a bounded control input u(k) such that:

$$\lim_{k->\infty}[y_d(k) - y(k) = e(k)] = 0$$

(k: the steps of the GA optimization process.)

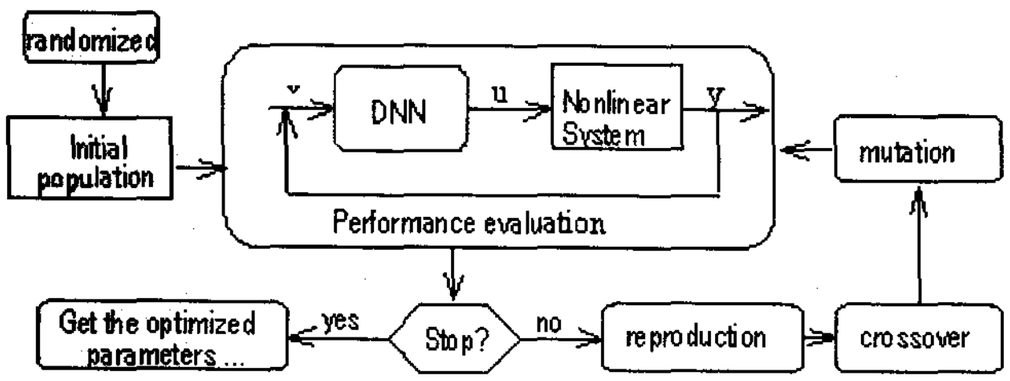

Fig.4 A functional block diagram showing the GA optimization process.

6. Computer Simulation Results

In our simulations, we applied this model to control of an unknown nonlinear plant. The plant to be controlled is an arbitrary unknown function:

$f[.]=[2+\cos\{7\times(y2(k-1)+y2(k-2))\}]+e-u(k)/1+u2(k-1)+u2(k-2);$

The input to the system is $x(k)=\sin(2\times k/250)$ in the interval [-1,1].

In this experiment, the input signal x(k) is considered the desired response for the unknown nonlinear plant to follow.

The simulation result of the typical DNN is shown as Fig.5,6 [1], and our simulation result of the same DNN using GA optimized algorithm is shown as Fig.7,8.9.

From these figures, we could see whether the typical DNN model or the DNN with GA controlled the plant response follows the desired command signal closely.

7. Conclusion

This paper has presented a Dynamic Neural Network (DNN) where all its parameters be simultaneously tuned by Genetic Algorithm (GA). By appropriate coding of the DNN parameters, it can achieve self-tuning properties from an initial random state. By using of dynamic crossover and mutation probability rates, the tuning process by GA was further improved. Though it not better than typical DNN controller, and can be argued that in this model, before GA can be used to optimize its parameters, initial encoding and setting are required, however, such procedures are somewhat relatively simpler, and could widely be used.

8. References

[1] A.Satyadas and K. Krishnakumar, "GA-optimized fuzzy Controller for Spacecraft Attitude Control", Proc. IEEE Int'l Conf. Fuzzy System, pp.1979-1984, 1994

[2] P.Wang, D.P.Kwok, "Optimal Design of PID Process Controllers Based on Genetic Algorithms", Proc. Of the 12th Triennial World Congress of the IFAC, Sydney, Australia, pp.193-197, 1993

[3] Alen Varsek, Tanja Urbancic, Bodgan Filipic, "Genetic Algorithms in Controller Design and Tunning", IEEE Trans. On Systems, Man, and Cybernetics, vol.23, no.5, pp.1330-1339, 1993

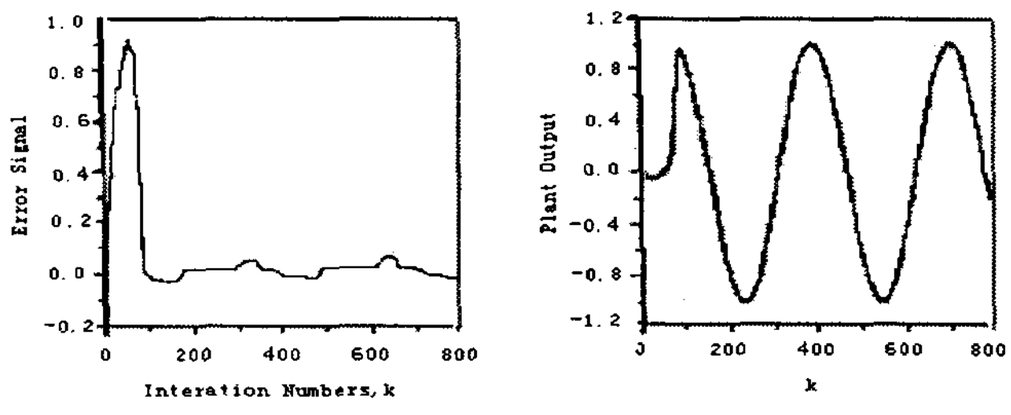

Fig.5 Error Signal of typical DNN
Fig.6 Plant Output of typical DNN

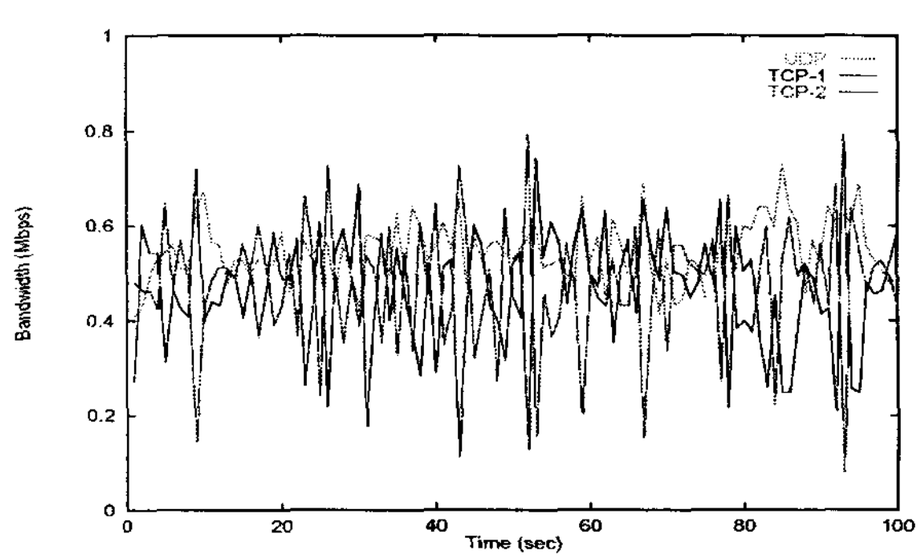

Fig.7 The Input and Nonlinear Signal

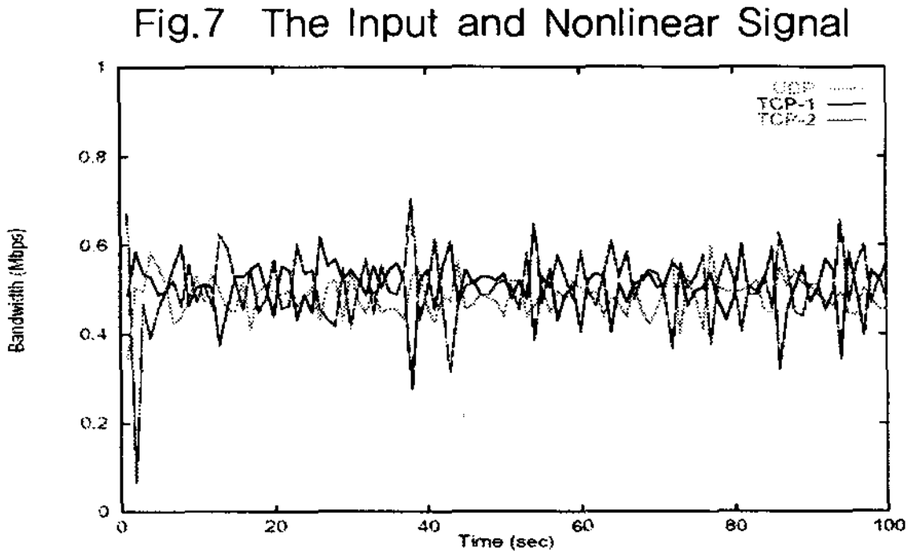

Fig.8 The DNN Output signal Using GA

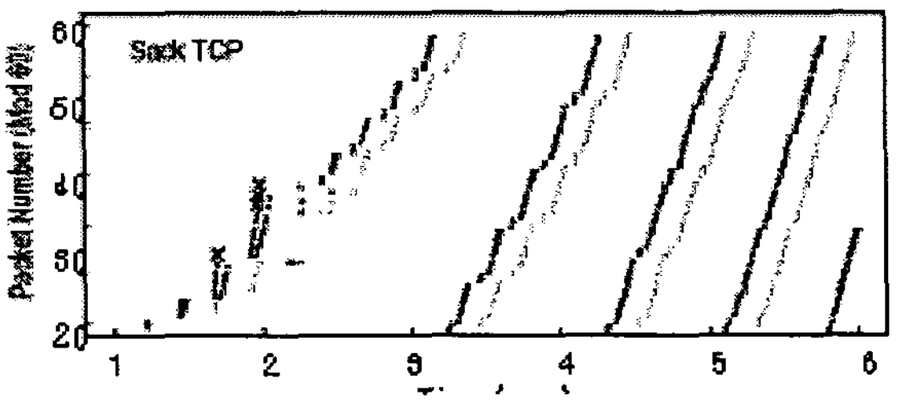

Fig.9 The Error Signal of DNN Using GA