

Geneo-tic Algorithms를 이용한 비선형 시스템 제어

조현섭*

*청운대학교 디지털방송공학과

e-mail:chohs@chungwoon.ac.kr

Unknown Nonlinear Systems Control Using Genetic Algorithms

Hyun-Seob Cho*

*Dept of Digital Broadcast Engineering Chungwoon University

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 method is 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 training.

1. Introduction

Control refers to a task which is to apply appropriate inputs to a plant so that the plant performs in a desirable way. 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.

The "dynamic neural Units"(DNUs) which have been proposed by professors M.M.Gupta and D.H.Rao with a different architecture to model a new artificial neuron that enables both the synaptic and somatic adaptations to control the nonlinear

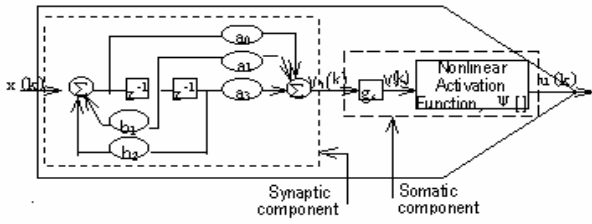
system.

In this paper, we present a dynamic neural Network(DNN) that created by DNUs to control the nonlinear system. and give an assume of using Genetic algorithms(GA) to optimize the parameters of the network weights.

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.

The topology of a single dynamical neural unit consists delay elements, feedforward and feedback synaptic weights and a nonlinear somatic operator. The architecture of the DNU model is illustrated in Fig 1.[1]



[Fig.1] Basic structure of DNU, which consists of synaptic and somatic components.

The difference equation that describes the behavior of the second order dynamical structure is given in Eqn.(1) in which $v_1(k)$, $x(k)$ are the input and output of the DNU. Similarly, the pulse transfer function of this part can be given by Eqn.(2).

$$v_1(k) = -b_1v_1(k-1) - b_2v_1(k-2) + a_0x(k) + a_1x(k-1) + a_2x(k-2) \quad (1)$$

$$T(z) = \frac{a_0 + a_1z^{-1} + a_2z^{-2}}{1 + b_1z^{-1} + b_2z^{-2}} \quad (2)$$

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;

$$w_{k+1} = w_k - \mu \frac{\partial J}{\partial w} \quad (5)$$

$$J = \frac{1}{2} E(e^2(k)) \quad (6)$$

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

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

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

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

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

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

DNU layer includes the desired number of individual DNU blocks whose inputs are connected

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.

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[4][5][6]. 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 [2].

4.1 Genetic Algorithm

Genetic Algorithm (GA) is a random technique that imitates natural evolution with Darwinian survival of the fittest approach. GAs perform on the coding of the parameters and not on the exact parameters, therefore, it does not depend on the continuity of the parameter nor the existence of derivatives of the functions as needed in some conventional optimization algorithms. The coding method allows GAs to handle multi-parameters or multi-model type of optimization problems easily, which is rather difficult or impossible to be treated by classical optimization methods[3].

4.2 Tuning of the DNN Parameters by GA:

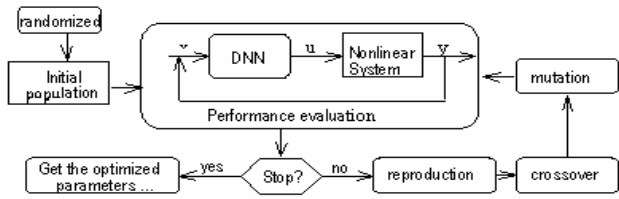
The steps optimizing parameters of DNN by GA is as follows:

1 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.

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

1 Optimization by GA: the process is shown in

fig.2:



[Fig. 2] A functional block diagram showing the GA optimization process.

The terminal condition of the GA optimization process is to determine a bounded control input $u(k)$ such that: $\lim_{k \rightarrow \infty} [y_d(k) - y(k) = e(k)] = 0$ (k : the steps of the GA optimization process.)

5. 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: $f[J] = [2 + \cos(7 - (y_2(k-1) + y_2(k-2)))] + e^{-u(k)/1 + u_2(k-1) + u_2(k-2)}$; The input to the system is $x(k) = \sin(2 - k/250)$ 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.3,4 [1], and our simulation result of the same DNN using GA optimized algorithm is shown as Fig.5,6

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.

6. 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.

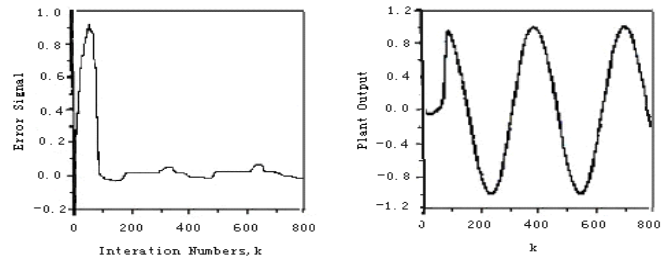
7. References

[1]. M.M.Gupta, D.H.Rao, "Dynamic Neural Units with Applications to the Control of Unknown Linear System", in *Journal of Intelligent and Fuzzy Systems*, pp.73-92, vol.1, no.1

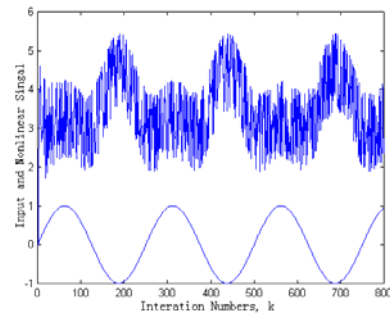
[2] P.Wang, D.P.Kwok, "Optimal Design of PID Process Controllers Based on Genetic Algorithms", *Proc. Of the 12th Triennial World Congress of the IFAC*, Sydney, Australia, pp.193-197, 2005

[3] W.Schiffmann, M.Joost, R.Werner, "Synthesis and Performance Analysis of Multilayer Neural Network Architectures", *Technical Report 16/1992*, University of Koblenz, 2006

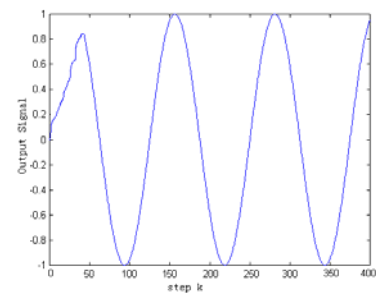
Ungar, L. H., "A Bioreactor Benchmark for Adaptive-Network Based Process Control," in W.T. Miller, R. S. Sutton, P. J. Werbos, *Neural Networks for Control*, MIT Press, 2007



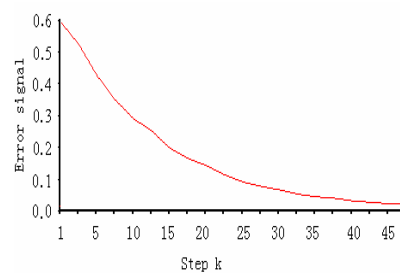
[Fig. 3] Error Signal of typical DNN



[Fig. 4] Plant Output of typical DNN



[Fig. 5] The Input and Nonlinear Signal



[Fig. 6] The Error Signal of DNN Using GA