

비선형 시스템 제어를 위한 동적 신경망의 최적화

° 유동완 · 이진하 · 이영석 · 서보혁
* 경북대학교 전기공학과 ** 영진전문대 전기공학과

Optimization of Dynamic Neural Networks for Nonlinear System control

° Dong-Wan Ryoo · Jin-Ha Lee · Young-Seog Lee ** · Bo-Hyeok Seo

*Dept. of Electrical Eng. Kyungpook Nat. Univ. **Dept. Electrical Eng. Yeunghin Junior Coll.

Abstract - This paper presents an optimization algorithm for a stable Dynamic Neural Network (DNN) using genetic algorithm. Optimized DNN is applied to a problem of controlling nonlinear dynamical systems. DNN is dynamic mapping and is better suited for dynamical systems than static forward neural network. The real time implementation is very important, and thus the neuro controller also needs to be designed such that it converges with a relatively small number of training cycles. SDNN has considerably fewer weights than DNN. The object of proposed algorithm is to the number of self dynamic neuron node and the gradient of activation functions are simultaneously optimized by genetic algorithms. To guarantee convergence, an analytic method based on the Lyapunov function is used to find a stable learning for the SDNN. The ability and effectiveness of identifying and controlling a nonlinear dynamic system using the proposed optimized SDNN considering stability is demonstrated by case studies.

1. 서론

신경망을 이용한 적응 제어기법은 기존의 제어기법에 비해 시스템 파라미터를 추정하는 데 복잡한 수식이 필요 없으며 비선형성이 강한 시스템에 대해서도 좋은 제어 성능을 보인다. 신경망을 이용한 비선형 동적 시스템의 식별과 제어가 폭 넓게 연구되고 있다. 기존의 모델 기준 적응 제어와 유사한 구조를 갖는 정적 신경망을 이용하여 비선형 시스템에 대해 제어방법을 제시하였고, 비선형 시스템의 자기동조 제어를 위한 정적 전방향 신경망의 학습방법을 제안하였다[1-3]. 이러한 비선형 시스템의 제어기 설계에 이용되어 온 신경망은 정적 전방향 신경망구조와, 동적 신경망 구조로 분류될 수 있다. 정적 전방향 신경망은 동적 사상을 표현하지 않는 정적 사상인 반면에 동적 신경망은 동적 사상이다. 이 중에 동적 시스템을 제어하기에 적합한 신경망은 내부적으로 동적인 특성을 갖는 동적 신경망이다 [4-5]. 그러나 동적 신경망은 연결강도의 수가 많기 때문에 제어를 수행하기 위해서는 무엇보다도 학습속도가 빠른 신경망 구조가 필요하고, 신경망제어기의 노드 수와 활성화 함수의 기울기는 시스템의 출력에 직접적인 영향을 미치므로 신중을 기해야한다. 그러나 기존의 신경망으로 구성된 제어기는 체계적인 방법이 제시되어 있지 않아 노드 수와 활성화 함수의 기울기를 결정하는데 어려움이 있다.

따라서 본 논문에서는 안정성을 고려한 연결강도수가 적은 자기 동적 신경망을 사용하여 유전알고리즘으로 노드수와 활성화함수를 최적화 시키므로써 비선형 시스템을 제어할 수 있는 알고리즘을 제안한다. 유전알고리즘은 복잡한 최적화 문제를 해결하려는 도구로서 목적함수 외 탐색공간

에 대한 사전 지식이 필요하지 않고 매우 크고 복잡한 공간일지라도 전역해쪽으로 수렴해 가는 특성을 가지고 있다 [6,7]. 제안된 기법을 사용한 최적화 된 자기 동적 신경망으로 비선형 동적 시스템의 식별과 제어에 관한 사례연구를 통하여 알고리즘의 효용성을 확인하였다.

2. 동적 신경망의 구조

정적 전방향 신경망은 단순히 연결강도에 가해진 입력의 합을 활성화 함수를 통해 출력을 나타낸다. 반면에 동적 신경망은 상태 변수의 방정식으로 출력을 표현할 수 있다. 동적 신경망의 수학적인 표현은 식 2.1로 표현된다.

$$O^g(k) = \sum_{i=1}^g W_i^g f(S_i(k)) = \sum_{i=1}^g W_i^g f \left(\sum_{j=1}^g W_{ij}^g I_j(k) + \sum_{j=1}^g W_{ij}^H X_{jg}(k-1) \right) + \sum_{j=1}^g W_{ij}^H X_{jg}(k-2) \quad (2.1)$$

여기서 $X_{ij}(k-1)$ 은 $k-1$ 시작에서 g 번째 신경의 상태이며, $I_j(k)$ 는 신경으로 들어오는 외부입력, $S_i(k)$ 는 외부 입력의 합을 나타낸다. $O^g(k)$ 는 동적 신경망의 출력, W^I, W^H, W^O 는 입력층, 중간층, 출력층의 연결강도행렬을 나타낸다. $f(\cdot)$ 는 신경의 활성화 함수이다. 자기 동적 신경망은 동적 신경망에서 변화된 구조를 가진다. 즉 자기 동적으로 구성된 신경망으로 구성되며 연결강도의 수가 적다. 자기 동적 신경망을 수학적인 표현식으로 나타내면 식 2.2와 같다.

$$O(k) = \sum_{j=1}^D W_j^O f(S_j(k)) = \sum_{j=1}^D W_j^O f \left(\sum_{i=1}^D W_{ij}^D I_i(k) + W_j^D X_j(k-2) \right) = \phi_j^D(I_i(t)), \quad i=0,1,2,\dots,k \quad (2.2)$$

여기서 $X_j(k)$ 는 j 번째 동적 신경의 출력이며, $S_j(k)$ 는 j 번째 동적 신경으로 들어오는 외부입력의 합이며, $O(k)$ 는 신경망의 출력이며, W^I, W^D, W^O 는 입력층에서 중간층, 중간층에서 출력층, 출력층에서 출력층의 연결강도행렬이다. $\phi_j^D(\cdot)$ 은 비선형 함수이고 l 과 k 는 0 또는 양의 실수이다. 식 2.2도 이전의 입력을 포함하고 있는 비선형 동적 사상을 나타낸다.

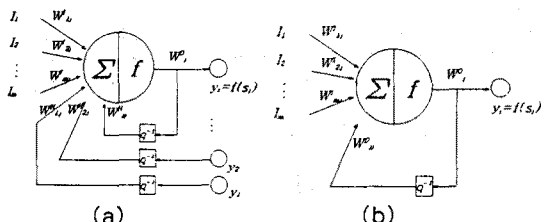


그림 1. 동적 신경기(DNU)와 자기 동적 신경기(SDNU)의 구조

Fig. 1. Structures of Dynamic neural unit and Self Dynamic neural unit.

3. 자기 동적 신경망의 최적화와 제어 알고리즘

3.1 유전 알고리즘을 이용한 동적 신경망의 최적화

유전 알고리즘은 자연 세계의 진화 과정을 컴퓨터상에서 시뮬레이션 하므로써 복잡한 실제세계의 문제를 해결하고자 하는 계산 모델이다. 특히 적응적 탐색 학습과 최적화를 통한 공학적인 문제의 해결에 많이 응용된 바 있다. 이러한 유전 알고리즘을 이용하여 본 논문에서 동적신경망의 중간층 유닛의 개수와 활성화 함수의 기울기를 최적화 한다. 자연도태(natural selection)의 유전적인 메카니즘에 기초한 유전 알고리즘은 세대(generation)를 거듭함에 따라 어떤 최적화하는 해에 수렴하는, 또는 도달하는 탐색 알고리즘이다. 탐색 공간이 크거나 분석적으로 해를 찾을 수 없는 문제에 대해 유용한 알고리즘이다.

유전자 알고리즘을 이용하여 최적해를 구하는 과정은 다음과 같다.

- 1) 부호화(Coding) : 문제에 따라 다를 수 있겠지만, 보통 2진수를 이용하여 랜덤하게 해집단을 생성시킨다.
- 2) 평가(Evaluation) : 주어진 목적함수에 대한 각 해의 적합도(fitness)를 평가한다.
- 3) 복제(Reproduction) : 적합도에 따라 염색체들을 랜덤하게 복제한다. (자연생태계의 적자생존의 원리를 이용한 roulette wheel selection 방법을 사용)
- 4) 교배(Crossover) : 두 부모 염색체를 조합하여 새로운 정보를 가진 자식 염색체를 생성한다.
- 5) 돌연변이(Mutation) : 염색체의 일부분을 랜덤하게 변화시켜 다양한 해를 구할 수 있다.

자기 동적 신경망으로 구성된 식별기를 수식으로 표현하면 식 3.1-3.5과 같이 나타낼 수 있다.

$$\hat{y}(k) = \sum_j W_j^{j0}(k) X_j^{j0}(k) + \sum_j W_j^{j1}(k) X_j^{j1}(k) U(k) \quad (3.1)$$

$$X_j^{j0}(k) = f(S_j^{j0}(k)) \quad (3.2)$$

$$S_j^{j0}(k) = W_j^{j0} X_j^{j0}(k-1) + \sum W_{ij}^{j0} I_i(k) \quad (3.3)$$

$$X_j^{j1}(k) = f(S_j^{j1}(k)) \quad (3.4)$$

$$S_j^{j1}(k) = W_j^{j1} X_j^{j1}(k-1) + \sum W_{ij}^{j1} I_i(k) \quad (3.5)$$

여기서 $X_j^{j0}(k)$ 와 $X_j^{j1}(k)$ 는 추정함수 $\hat{f}(\cdot)$ 와 $\hat{g}(\cdot)$ 의 j 번째 동적 신경의 출력이며, $I_i(k)$ 는 신경으로 들어오는 i 번째 외부입력이며, $S_j^{j0}(k)$ 와 $S_j^{j1}(k)$ 는 각각의 추정함수 $\hat{f}(\cdot)$ 와 $\hat{g}(\cdot)$ 의 j 번째 동적 신경으로 들어오는 외부 입력의 합이며, $\hat{y}(k)$ 는 식별기의 출력이며, $U(k)$ 는 시스템 입력이며, W_j^{j0}, W_j^{j1} 와 W_{ij}^{j0}, W_{ij}^{j1} 는 추정함수 $\hat{f}(\cdot)$ 와 $\hat{g}(\cdot)$ 의 입력층에서 중간층, 중간층에서 중간층, 중간층에서 출력층의 연결강도행렬이다. 비선형 함수의 자기 동적 신경망의 모형에 유전알고리즘의 적용을 그림 2에 나타내었다. 그림 2에서 와 같이 연결강도수를 줄인 자기동적신경망을 이용하여 시스템을 f 와 g 로 분리 추정하며 유전알고리즘으로 동적신경망의 유닛의 개수와 활성화 함수의 기울기를 최적화 한다.

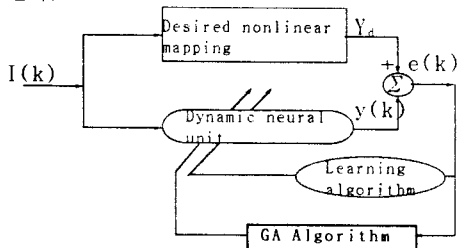


그림 2. 유전 알고리즘을 이용한 동적 신경망의 최적화
Fig. 2. Optimization of dynamic neural unit using GA algorithm.

3.2 동적 신경망의 제어 알고리즘

자기 동적 신경망을 사용하여 비선형 시스템을 식별하고 제어할 수 있다. 자기 동적 신경망으로 구성된 식별기는 원하는 시스템의 출력을 추정하여 신경망 제어기의 입력으로 사용되어지고, 신경망으로 구성된 제어기는 시스템을 제어한다. 신경망으로 구성된 식별기로부터 제공된 정보를 가지고 시스템의 제어입력을 구해낸다. 임의의 비선형 동적 시스템을 이산 시스템으로 나타내면 식 3.6과 같이 표현할 수 있다[3.5.11].

$$y(k+1) = f[y(k), y(k-1), \dots, y(k-n), u(k-1), u(k-2), \dots, u(k-m)] + g[y(k), y(k-1), \dots, y(k-n), u(k-1), u(k-2), \dots, u(k-m)] \quad (3.6)$$

여기서 $y(k)$ 는 시스템 출력이며 $u(k)$ 는 시스템 입력이고 $f(\cdot)$ 와 $g(\cdot)$ 는 비선형 함수이다. 오차 함수는 실제 시스템의 출력과 식별기 출력의 함수로 표현된다.

$$E = \frac{1}{2} (y(k) - \hat{y}(k))^2 \quad (3.7)$$

추정되는 함수의 각 연결강도 오차 기울기는 식 3.8과 같이 나타낼 수 있다.

$$\frac{\partial E}{\partial W} = -e(k) \frac{\partial \hat{y}(k)}{\partial W} \quad (3.8)$$

여기서 $e(k) = y(k) - \hat{y}(k)$ 이다. 신경망의 연결강도의 학습 법칙은 식 3.9와 같다.

$$W(k+1) = W(k) + [\eta] \left(-\frac{\partial E}{\partial W} \right) \quad (3.9)$$

여기서 $[\eta]$ 는 추정되는 함수의 각 연결강도 학습률이다. 본 논문에서 사용된 자기 동적 신경망 $\hat{f}(I, W)$ 의 입력층, 중간층, 출력층의 연결강도에 대한 출력의 기울기는 식 3.10-3.12과 같이 나타낼 수 있다.

$$\frac{\partial \hat{y}(k)}{\partial W_j^{j0}} = X_j^{j0}(k) \quad (3.10)$$

$$\frac{\partial \hat{y}(k)}{\partial W_j^{j1}} = W_j^{j1} L_j^{j1}(k) \quad (3.11)$$

$$\frac{\partial \hat{y}(k)}{\partial W_{ij}^{j1}} = W_j^{j1} M_{ij}^{j1}(k) \quad (3.12)$$

여기서 $L_j^{j1}(k) = \partial X_j^{j1}(k) / \partial W_j^{j1}$, $M_{ij}^{j1}(k) = \partial X_j^{j1}(k) / \partial W_{ij}^{j1}$ 이다. 식 3.10에서 식 3.12에 표현된 자기 동적 신경망 $\hat{f}(I, W)$ 의 입력층에서 중간층, 중간층에서 중간층, 중간층에서 출력층의 연결강도에 대한 출력의 기울기를 사용하여 일반적인 역전파 법으로 학습시킬 수 있다. 그리고 신경망 $\hat{g}(I, W)$ 의 입력층, 중간층, 출력층의 연결강도에 대한 출력의 기울기는 식 3.13에서 식 3.15과 같다.

$$\frac{\partial \hat{y}(k)}{\partial W_j^{j0}} = X_j^{j0}(k) U(k) \quad (3.13)$$

$$\frac{\partial \hat{y}(k)}{\partial W_j^{j1}} = W_j^{j1} U(k) L_j^{j1}(k) \quad (3.14)$$

$$\frac{\partial \hat{y}(k)}{\partial W_{ij}^{j1}} = W_j^{j1} U(k) M_{ij}^{j1}(k) \quad (3.15)$$

여기서 $L_j^{j1}(k) = \partial X_j^{j1}(k) / \partial W_j^{j1}$, $M_{ij}^{j1}(k) = \partial X_j^{j1}(k) / \partial W_{ij}^{j1}$ 이다. 그림 3은 자기 동적 신경망으로 구성된 제어기의 블록선도를 나타낸 것이다. 유전알고리즘은 식별기와 제어기의 최적화 시킨다. 자기 동적 신경망으로 구성된 식별기의 입력은 이전 시스템의 출력, 그리고 이전의 제어신호로 구성되며, 식별기 출력과 시스템 출력사이의 오차를 최소화하므로써 식별된 정보를 제어기에 제공한다. 자기 동적 신경망으로 구성된 제어기의 입력은 기준입력, 이전 시스템의 출력, 그리고 이전의 제어신호로 구성되며, 신경망으로 구성된 제어기의 출력이 시스템의 제어신호가 된다.

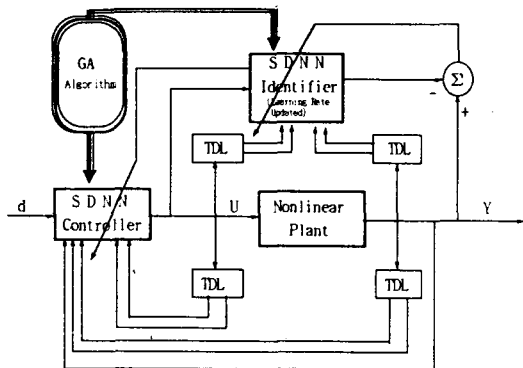


그림 3 자기 동적 신경망으로 구성된 제어기의 블록선도
Fig. 3 A block diagram of SDNN based control system.

학습 제어 알고리즘에서 작은 값의 학습률은 수렴성은 보장할지라도 학습속도가 매우 느리게 된다. 반면에 학습률이 너무 크면 제어 알고리즘이 불안정하게 된다. 즉 안정성을 보장하지 못한다. 안정성을 고려한 학습률을 Lyapunov 함수를 사용하여 유도한다. Lyapunov 후보 함수를 식 3.16과 같이 나타낸다.

$$V(k) = \frac{1}{2} e^2(k) \quad (3.16)$$

Lyapunov 함수의 변화량은 식 3.17과 같이 표현될 수 있다.

$$\Delta V(k) = V(k+1) - V(k) = \frac{1}{2} [e^2(k+1) - e^2(k)] \quad (3.17)$$

η 를 연결강도의 학습률이라 하고 P_{MAX} 를 연결강도에 대한 출력의 변화율의 최대값으로 식 3.18과 같이 정의한다.

$$P_{MAX} = \max_k \|F(k)\| \quad (3.18)$$

여기서 $F(k) = \frac{\partial y(k)}{\partial W}$ 이고 $\|\cdot\|$ 는 Euclidean norm이다.

η 가 식 3.19를 만족하면 학습 알고리즘의 수렴성을 보장한다(8).

$$0 < \eta < \frac{2}{P_{MAX}^2} \quad (3.19)$$

본 논문에서 사용된 안정성을 고려한 $f(\cdot)$ 와 $g(\cdot)$ 의 학습률을 사용하였다.

4. 모의실험

사례연구를 통하여 유전알고리즘으로 최적화된 안정한 학습을 고려한 자기동적신경망을 사용하여 비선형 시스템의 식별, 제어능력의 유용성을 알아보기 위해서 모의실험을 실시하였다. 첫째로, 다음과 같은 비선형 시스템 모형(3)[5]에 적용하여 시스템을 식별제어하는 경우를 다루어 본다.

$$y_p(k+1) = a \sin[2y_p(k)] + b u(k) \quad (4.1)$$

$$a=0.8, \quad b=1.2$$

$$u(k) = 0.5 [\sin(0.5r_k) + \cos(2r_k) + \sin^2(3r_k) + \cos^2(4r_k) \cos(5r_k) + \sin(20r_k)], \quad r_k = \pi k/40 \quad (4.2)$$

식 4.1-4.2로부터 입·출력 데이터를 얻어 시스템을 식별하고 15회 반복 학습한 후 학습된 초기치를 가지고 플랜트 파라미터가 변화할 때, 시스템의 제어를 한다. 그림 4는 자기 동적신경망을 유전알고리즘으로 유닛 개수와 활성화 함수의 기울기를 최적화시킬 때 최대 적합도의 변화를 나타내었다. 교배확률(P_c)은 0.9, 돌연변이 확률 P_m 은 0.3으로 하였다. 최적화 된 신경망을 가지고 식 5.1로부터 200개의 입·출력쌍의 데이터를 15회 반복 학습한 후 결과물 그림 5에 나타내었다. 유전알고리즘으로 최적화 시킨 동적신경망 오차의 수렴특성이 우수함을 그림 6으로부터 알 수 있다. 플랜트의 파라미터가 70번째부터 변화하는 경우에 제어 성능을 그림 7에 나타내었다.

표 1. 사례연구 1에 대한 15회 학습한 후 적합도와 오차의 실효치

Table 1. Fitness and mean square error for case study after 15 training cycles.

Fitness, MSE (before Optimize)	Fitness	270.27
	MSE	0.0037
Fitness, MSE (after Optimize)	Fitness	2618.126
	MSE	0.00038

표 2. 사례연구 2에 대한 15회 학습한 후 적합도와 오차의 실효치

Table 2. Fitness and mean square error for case study 2 after 15 training cycles.

Fitness, MSE (before Optimize)	Fitness	75.7576
	MSE	0.0132
Fitness, MSE (after Optimize)	Fitness	82.6446
	MSE	0.0121

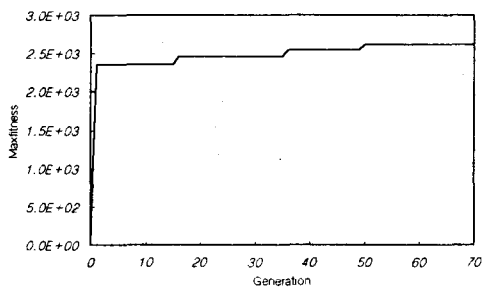


그림 4. SDNN의 유전알고리즘 적합도
Fig. 4. Fitness of genetic algorithm for SDNN.

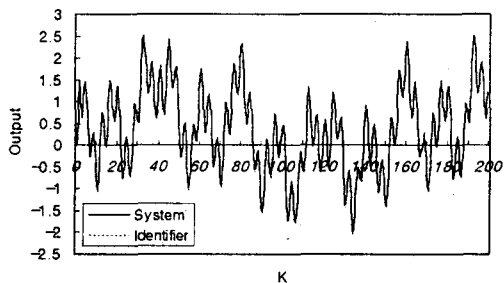


그림 5. 200쌍을 15회 학습 후 시스템 출력(y_p)과 식별기(\hat{y}) 출력

Fig. 5. Output of the system and identifier after 15 training cycles using 200 pairs.

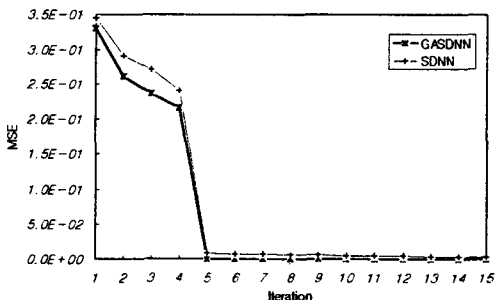


그림 6. 최적화된 SDNN 오차의 실효치 비교
Fig. 6. Mean squared error comparison with Optimized SDNN

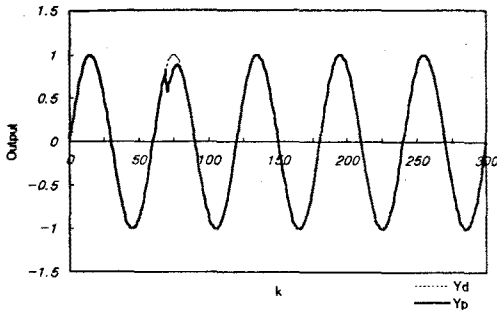


그림 7. 시스템 파라미터 a와 b가 70번째 샘플링순간에 a=0.8 에서 a=0.5, b=1.2에서 b=0.8로 변했을 때 의 제어성능 (15회 학습 후)

Fig. 7. Control Performance when system parameter a and b are varied 0.8 to 0.6 and 1.2 to 0.8 at 70-th sampling time (after 15 cycles).

둘째로, 식 4.3-4.4와 같은 비선형 시스템 모형 [14] 으로부터 입·출력 데이터를 얻어 시스템을 식별하고 15회 반복 학습한 후 학습된 초기치를 가지고 식 4.4와 같은 플랜트 변화를 식별하여 제어한다.

$$y_p(k+1) = 0.35 \left\{ \frac{y_p(k) y_p(k-1) + [y_p(k) + 2.5]}{1 + y_p(k)^2 + y_p^2(k-1)} + u(k) \right\} \quad (4.3)$$

$$u(k) = \left[\begin{array}{c} \sin(0.5r_k) + \cos(2r_k) + \sin^2(3r_k) + \cos^3(4r_k) \\ \cos(5r_k) + \sin(20r_k) \end{array} \right] - 0.75, \quad r_k = \pi k / 40 \quad (4.4)$$

$$y_p(k+1) = 0.35 \left\{ \frac{y_p(k) + [y_p(k-1) + 2.5]}{1 + y_p^2(k-1)} + 1.2 u(k) \right\} \quad (4.5)$$

15회 학습한 후 동적 신경망으로 학습한 결과와 비교한 결과를 표 2에 나타내었다. 플랜트가 100번째부터 식 5.5와 같이 변화하는 경우에 대해 학습한 결과를 그림 10에 나타내었다.

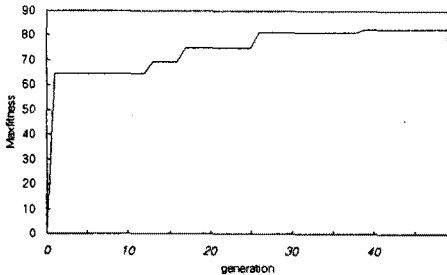


그림 8. SDNN의 유전알고리즘 적합도
Fig. 8. Fitness of genetic algorithm for SDNN.

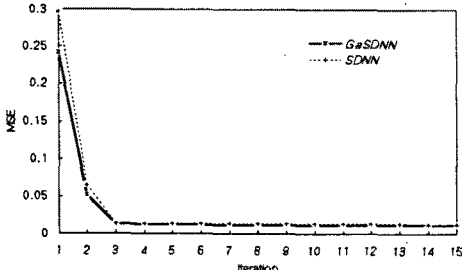


그림 9. 최적화된 SDNN 오차의 실효치 비교
Fig. 9. Mean squared error comparison with Optimized SDNN

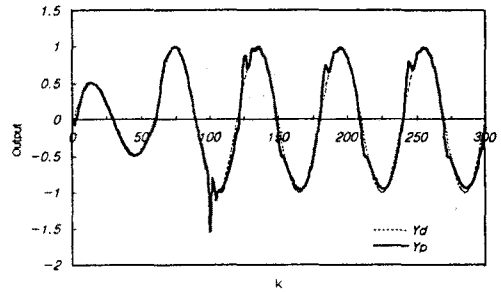


그림 10. 100 번째 샘플링 순간에 시스템이 변화했을 때의 제어 성능 (15회 학습 후)

Fig. 10. Control performance when system are varied at 100-th sampling time (after 15 cycles).

5. 결 론

동적 신경망 제어기의 노드 수와 활성화 함수의 기울기는 시스템의 출력에 직접적인 영향을 미치므로 신중을 기해야 한다. 그러나 기존의 신경망으로 구성된 제어기는 체계적인 방법이 제시되어 있지 않아 노드수와 활성화 함수의 기울기를 결정하는데 어려움이 있다. 따라서 본 논문에서는 안정성을 고려한 연결강도수가 적은 자기 동적 신경망을 사용하여 유전 알고리즘으로 노드수와 활성화 함수를 최적화 시키므로 비선형 시스템을 제어할 수 있는 알고리즘을 제안하였다. 유전 알고리즘은 탐색 공간이 크거나 분석적으로 해를 찾을 수 없는 문제에 대해 유용한 알고리즘이다. 제안된 알고리즘을 사용하여 최적화된 자기 귀환 신경망으로 비선형 시스템의 식별과 제어를 위한 모의실험을 통해 학습 알고리즘의 안정성과 효용성을 확인하였다.

[참 고 문 헌]

- [1] K. S. Narrendra and K. Pathasarathy, "Gradient Methods for the Optimization of Dynamical Systems Containing Neural Networks," *IEEE Trans. on Neural Networks*, vol. 2, pp. 252-262, Mar. 1991
- [2] K. S. Narrendra and K. Pathasarathy, "Identification and control of dynamical systems using neural networks," *IEEE Trans. on Neural Networks* vol 1, pp. 4-27, 1990.
- [3] C. F. Chen, "Back-propagation neural networks for nonlinear self-tuning adaptive control," *IEEE Contr. Syst. Mag.* vol. 10, no 3, pp. 44-48, Apr. 1990.
- [4] George A. Rovithakis and A. Christodoulou, "Adaptive Control of Unknown Plants Using Dynamical Neural Networks," *IEEE Trans on SMC*, Vol. 24 No. 3, pp. 400-411, March, 1994.
- [5] A. Karakasoglu, S. I. Suharsanan and M. K. Sundareshan, "Identification and Decentralized Adaptive Control Using Dynamical Neural Networks with Application to Robotic Manipulators," *IEEE Trans on Neural Networks*, vol. 4, no 6, pp. 919-930, 1993.
- [6] Goldberg, D. E., *Genetic algorithms in search, optimization and machine learning*, Addison Wesley, 1989
- [7] ichalewicz., *Genetic Algorithms + Data Structures = Evolution Programs*, Springer-Verlag, 1994
- [8] M. M Polycarpou and P. A. Ioannou, "Learning convergence analysis of neural-type structured networks," *IEEE Trans. Neural Networks*, vol. 3, no. 1, pp. 39-50, Jan. 1992.
- [9] Chao-chee Ku and Kwang Y. Lee, "System Identification and Control Using Diagonal Recurrent Neural Networks," *Proc. ACC* vol. 1, pp. 545-549, July 1992.
- [10] P. S. Sastry, G. Santharam and K. P. Unnikrishnan, "Memory Neuron Networks for Identification and Control of Dynamical System," *IEEE Trans. Neural Networks*, vol. 5, no. 2, pp. 306-319, Mar. 1994.
- [11] K. S. Narendra, J. Balarkrishnan and M. K. Ciliz, "Adaptation and Learning Multiple Models, Switching, and Tuning," *IEEE Contr. Syst. Mag.* vol. 15, no 3, pp. 37-51, June, 1995.