

신경회로망을 이용한 이산치 혼돈 시스템의 모델 예측 제어

°김세민*, 최윤호**, 박진배*, 주영훈***

*연세대 전기·컴퓨터공학과, **경기대 전자공학과, ***군산대 제어계측공학과

Model Predictive Control of Discrete-Time Chaotic Systems Using Neural Network

*Se-Min Kim, **Yoon-Ho Choi, *Jin-Bae Park, ***Young-Hoon Joo

*Dept. of Electrical & Computer Engineering, Yonsei Univ.

**Dept. of Electronic Engineering, Kyonggi Univ.

*** Dept. of Control & Instrumentation Engineering, Kunsan National Univ.

Abstract – In this paper, we present model predictive control scheme based on neural network to control discrete-time chaotic systems. We use a feedforward neural network as nonlinear prediction model. The training algorithm used is an adaptive backpropagation algorithm that tunes the connection weights. And control signal is obtained by using gradient descent (GD), some kind of LMS method. We identify that the system identification results through model prediction control have a great effect on control performance. Finally, simulation results show that the proposed control algorithm performs much better than the conventional controller.

I. 서론

비선형 동역학 시스템에서 발생하는 혼돈 현상은 시스템의 상태 변수들간의 비선형적인 관계로 인해 비주기적인 동작을 나타내지만, 내부적으로 결정론적 규칙을 가지고 있는 현상으로 정의되며, 초기 조건에 매우 민감한 특성을 가진다. 따라서 정확한 비선형 시스템 모델을 알고 있다 하더라도 시스템의 출력을 예측하기 어려우며, 이것이 혼돈 시스템이 갖은 본질적인 특징이라 할 수 있다[1].

본 논문에서는 혼돈 시스템의 지능 제어를 위해 신경 회로망을 이용한 모델 예측 제어기 설계 방법을 제안하고자 한다. 한편 제어 대상으로는 이산 시간 혼돈 시스템인 Henon 과 Lozi Map 을 선정하여, 제안된 모델 예측 제어 방법이 혼돈 시스템의 동정과 제어에 효과적으로 사용될 수 있음을 보이고자 한다.

2. 혼돈 시스템의 모델 예측 제어

2.1 신경회로망 모델

혼돈 비선형 시스템을 동정하기 위해 사용된 모델은 NARMA 모델로 일반적으로 비선형 차분 방정식으로 기술할 수 있다[2, 3].

$$\hat{y}(t+1) = f(y(t), y(t-1), \dots, y(t-n+1); u(t), u(t-1), \dots, u(t-m+1)) \quad (1)$$

여기서, f 는 신경회로망에 의해 동정 되어야 할 비선형 함수이다. 본 논문에서는 신경회로망 모델의 활성화 함수로 $\tanh(X)$ 을 사용하며, 출력층은 선형성을 유지하기 위해 사용하지 않았다.

그림 1은 출력 오차 e_c 가 제어기를 조절하기 위해 직접 사용되므로 혼돈 시스템의 직접 적응 제어를 나타낸다[4].

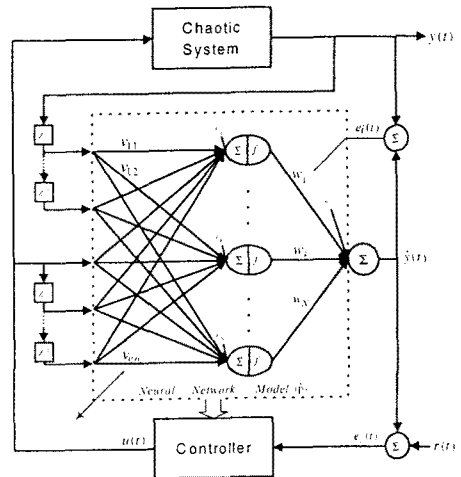


그림 1 혼돈 시스템의 직접 적응 제어
Fig. 1 Direct Adaptive Control of Chaotic Systems

2.2 학습 알고리즘과 제어 알고리즘

신경회로망을 통해 혼돈 시스템을 동정하고 \hat{y} 이 r 에 접근하도록 즉, 오차 $e_c = r - \hat{y}$ 가 0으로 접근하도록 적응 제어기를 구성한다. 시스템 동정을 수행하기 위한 신경회로망 학습 알고리즘은 오차 역전파로 측정값 y 와 추정값 \hat{y} 의 차로써 정의된 에너지 함수 $E(t) = 0.5(y(t) - \hat{y}(t))^2$ 을 최소화 하는 방향으로 연결강도를 갱신한다. 한편 제어 신호 $u(t)$ 을 구하는 과정은 다음과 같다. $u(t)$ 는 $J(=0.5(r(t+1) - \hat{y}(t+1)))$ 가 최소가 되도록 선택되어야 한다. 즉, $\partial J(t+1)/\partial u(t)$ 가 0에 접근할 때 최적의 제어 신호를 얻는다. 그리고 그림 1을 통해 다음과 같은 동정 모델의 출력을 나타낼 수 있다.

$$\hat{y}(t+1) = \sum_{j=1}^N w_j \tanh\left[\sum_{i=1}^{m+n} v_{ij}x_i + b_j\right] + b_o \quad (2)$$

여기서,

$$X = [y(t), y(t-1), \dots, y(t-n+1), u(t), u(t-1), \dots, u(t-m+1)]$$

N : 은닉층의 개수

$m+n$: 입력층의 개수

경사하강법(GD)을 적용하여 제어 신호 $u(t)$ 에 대해 갱신된 다음과 같은 식들을 얻을 수 있다.

$$u(t+1) = u(t) - \eta \frac{\partial J}{\partial u(t)} \quad (3)$$

$$\frac{\partial J}{\partial u(t)} = -e_c(t+1) \frac{\partial \hat{y}(t+1)}{\partial u(t)} \quad (4)$$

$$\frac{\partial \hat{y}(t+1)}{\partial u(t)} = \sum_{j=1}^N w_j \sec h^2 \left[\sum_{i=1}^{m+n} v_{ij} x_i + b_j \right] v_{i(n+1)j} \quad (5)$$

식 (3)-(5)을 이용하면 다음과 같은 식을 최종적으로 얻게 된다.

$$u(t+1) = u(t) + \eta e_c(t+1) \sum_{j=1}^N w_j \sec h^2 \left[\sum_{i=1}^{m+n} v_{ij} x_i + b_j \right] v_{i(n+1)j} \quad (6)$$

결과적으로 혼돈 비선형 시스템의 직접 적응 제어 방법을 단계별로 서술하면 다음과 같다[5].

1. 신경회로망 모델의 동정 입력 벡터 X 에 대해 출력 $\hat{y}(t+1)$ 을 구한다.
2. 오차 역전과 학습 알고리즘을 이용하여 연결강도를 변화시키고, 제어 오차 $e_c(t+1)$ 을 구한다.
3. 제어 오차 신호와 신경회로망의 동정 파라미터를 이용하여 새로운 제어 입력 신호를 얻는다.
4. 제어 입력 신호에 대한 혼돈 시스템 출력을 구한다.
5. 단계 1에서 단계 4까지 여러 번 반복하고, 혼돈 시스템의 출력과 목표값의 평균자승오차로서 제어 성능을 알아본다.

2.3 모델 예측 제어 알고리즘

2.2절에서 기술한 제어 알고리즘은 제어 신호로서 현재값 외에 미래값도 설계하는 일반 예측 제어 이론을 바탕으로 개선될 수 있다. 일반적으로 모델 예측 제어는 다음과 같은 4 단계로 구성된다[6, 7].

1. 각 샘플링 시간마다, 추정값, $\hat{y}(t+k)$ 가 예측 지표범위(prediction horizon) $k=1, \dots, N$ 만큼 예측된다.
2. 예측 지표범위만큼 기준 경로(reference sequence)가 정의된다.
3. 제어 오차로 이루어진 성능 함수(performance functional)가 최소가 되도록 예측된 제어 신호 벡터, $U(t+k)$ 가 계산된다. 그리고 제어 신호 벡터의 첫번째 성분이 플랜트에 인가된다.
4. 예측된 출력과 측정된 출력과의 오차를 신경회로망 모델 동정에 사용한다.

결과적으로 기준신호와 시스템 출력의 미래값이 제어 신호를 얻는데 필요하다. 신경회로망 모델이 근사적으로 제어 대상인 플랜트를 구현하기 때문에 그것은 시스템 출력의 미래값을 예측하는데 사용될 수 있다.

이러한 목적에 따라 N 을 예측 지표범위라고 하면, 기준 신호의 미래값과 신경회로망 출력의 미래값, 그리고 예측된 오차 벡터는 다음과 같이 된다.

$$\begin{aligned} R_{Ntr} &= [r(t+1), r(t+2), \dots, r(t+N)] \\ \hat{Y}_{Ntr} &= [\hat{y}(t+1), \hat{y}(t+2), \dots, \hat{y}(t+N)] \\ E_{Ntr} &= [e(t+1), e(t+2), \dots, e(t+N)] \end{aligned} \quad (7)$$

여기서, 오차는 $e(t+i) = r(t+i) - \hat{y}(t+i)$ 이고, 제어 입력 신호 벡터는 다음과 같이 정의 한다.

$$U_{Ntr} = [u(t), u(t+1), \dots, u(t+N)] \quad (8)$$

또한 성능 함수는 다음과 같다.

$$J = \frac{1}{2} [E_{Ntr}^T E_{Ntr}] \quad (9)$$

제어의 목적은 J 가 최소가 되도록 경사하강법에 의해 최적의 U_{Ntr} 을 구하는 것이다.

$$U_{Ntr+1} = U_{Ntr} - \eta \frac{\partial J}{\partial U_{Ntr}} \quad (10)$$

여기서,

$$\frac{\partial J}{\partial U_{Ntr}} = -E_{Ntr} \frac{\partial \hat{Y}_{Ntr}}{\partial U_{Ntr}} \quad (11)$$

$$\frac{\partial \hat{Y}_{Ntr}}{\partial U_{Ntr}} = \begin{bmatrix} \frac{\partial \hat{y}(t+1)}{\partial u(t)} & 0 & \dots & 0 \\ \frac{\partial \hat{y}(t+2)}{\partial u(t)} & \frac{\partial \hat{y}(t+2)}{\partial u(t+1)} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \hat{y}(t+N)}{\partial u(t)} & \frac{\partial \hat{y}(t+N)}{\partial u(t+1)} & \dots & \frac{\partial \hat{y}(t+N)}{\partial u(t+N-1)} \end{bmatrix}_{N \times N} = G_{Ntr} \quad (12)$$

식 (12)의 자코비안 (Jacobian)행렬 각 요소는 각각의 제어 신호에 대한 출력 신호의 편미분으로 얻어질 수 있다.

본 논문에서는 빠른 응답을 갖는 실시간 시스템에 적용 가능한 순환(Recursive) 계산 방법을 사용하여 자코비안 행렬을 계산하였으며, $N=3$ 일 때의 자코비안 행렬 계산 과정을 부록에 서술하였다.

그러면, 위에서 얻은 행렬을 이용한 새로운 제어 입력 신호는 식 (13)과 같이 유도 된다.

$$U_{Ntr+1} = U_{Ntr} + \eta E_{Ntr} G_{Ntr} \quad (13)$$

3. 모의 실험

본 논문의 모의실험에서는 대표적인 이산치 혼돈 비선형 시스템인 Henon과 Lozi Map을 제어대상으로 선정하였다. 또한 모의실험은 총 20000번의 학습과정을 통해 이루어졌으며, 신경회로망 모델의 은닉층의 개수 N 은 10으로 하였고, 입력수는 두개로 설정하였다.

1. Henon Map

Henon Map은 스트레인지 어트랙터 (strange attractors)의 미세구조를 조명하기 위해 고안된 것으로 제어 가능한 상태 방정식은 아래와 같다[8].

$$x_{n+1} = y_n + 1 - ax_n^2, \quad y_{n+1} = bx_n + a \quad (14)$$

여기서, a 와 b 는 조절 가능한 파라미터로서 각각 1.3과 0.3이고, 식 (14)의 기준신호로는 $r=[0-1]$ 을 선택하였다.

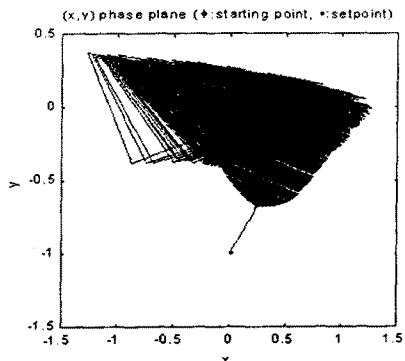


그림 2 (x,y)의 위상평면
Fig. 2 Phase Plane of Henon Map

그림 2은 약 5000번의 학습 횟수를 거쳐서야 수렴하는 모습을 보여주고 있다. 그리고 그림 3은 제어 입력 신호로서 -0.9956으로 일정한 상수값에 수렴하고 있음을 알 수 있다.

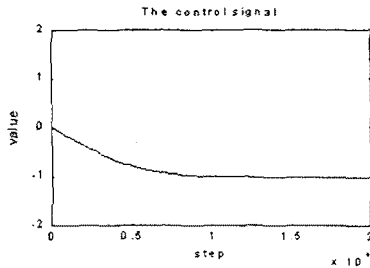


그림 3 제어 입력
Fig. 3 Control signal (Henon)

2. Lozi Map

Lozi Map은 아래와 같이 Henon Map과 유사한 형태를 갖는다.[8]

$$x_{n+1} = 1 + y_n - a|x_n|, \quad y_{n+1} = bx_n + u \quad (15)$$

여기서, a, b 는 $-1 < b < 1$ 인 실수 파라미터이고 그 값은 각각 1.4와 0.1로 선택하였다. 기준신호는 Henon Map과 마찬가지로 $r=[0-1]$ 로 선택하였다. 그림 4에서와 같이 Lozi Map에서는 비교적 빠른 수렴을 보여 약 700번째의 학습에서 이루어짐을 알 수 있다.

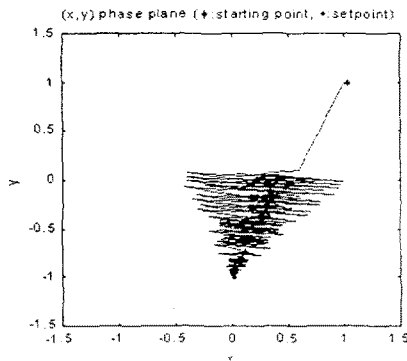


그림 4 (x,y)의 위상평면
Fig. 4 Phase Plane of Lozi Map

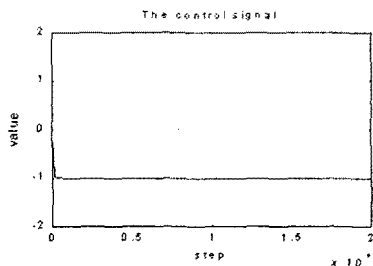


그림 5 제어 입력
Fig. 5 Control signal (Lozi)

4. 결론

본 논문에서는 신경회로망을 이용한 모델 예측 제어가 오차 역전과 학습 알고리즘과 관련되었을 때, 예측 제어를 적용하지 않았던 기존 방법보다 시스템 동정에 있어서 보다 향상된 성능을 보여주고 있다. 이산치 혼돈 시스템인 Henon과 Lozi Map에 대한 동정 오차의 MSE(mean square error)는 각각 $3.9209e-010$ 와 $2.5891e-009$ 이고, 또한 제어 성능의 우수성을 평가하기 위해 제어 오차를 측정해 본 결과, 각각 $8.0224e-005$,

$1.8170e-008$ 으로 나타났다. 결과적으로 본 논문의 성과는 주로 선형 시스템에 이용했던 예측 제어방법을 비선형 시스템 모델인 신경회로망에 적용하여 확실한 시스템 모델링이 가능하다는 결론을 얻을 수 있었고, 그것은 제어 성능에 있어서 만족할 만한 결과를 도출해 내었다.

[Appendix]

예측 지표범위 $N=3$ 일 때,

$$(1,1)\text{성분} : \frac{\partial \hat{y}(t+1)}{\partial u(t)} = \frac{df(x)}{du(t)} = g_1$$

$$(1,2)\text{성분} : \frac{\partial \hat{y}(t+1)}{\partial u(t+1)} = 0$$

$$(1,3)\text{성분} : \frac{\partial \hat{y}(t+1)}{\partial u(t+2)} = 0$$

$$(2,1)\text{성분} : \frac{\partial \hat{y}(t+2)}{\partial u(t)} = \frac{\partial f(x)}{\partial u(t)} + \frac{\partial f(x)}{\partial \hat{y}(t+1)} \frac{\partial \hat{y}(t+1)}{\partial u(t)} = g_2$$

$$(2,2)\text{성분} : \frac{\partial \hat{y}(t+2)}{\partial u(t+1)} = \frac{\partial f(x)}{\partial u(t+1)} = g_1$$

$$(2,3)\text{성분} : \frac{\partial \hat{y}(t+2)}{\partial u(t+2)} = 0$$

$$(3,1)\text{성분} :$$

$$\frac{\partial \hat{y}(t+3)}{\partial u(t)} = \frac{\partial f(x)}{\partial u(t)} + \frac{\partial f(x)}{\partial \hat{y}(t+1)} \frac{\partial \hat{y}(t+1)}{\partial u(t)} + \frac{\partial f(x)}{\partial \hat{y}(t+2)} \frac{\partial \hat{y}(t+2)}{\partial u(t)} = g_3$$

$$(3,2)\text{성분} : \frac{\partial \hat{y}(t+3)}{\partial u(t+1)} = \frac{\partial f(x)}{\partial u(t+1)} + \frac{\partial f(x)}{\partial \hat{y}(t+2)} \frac{\partial \hat{y}(t+2)}{\partial u(t+1)} = g_2$$

$$(3,3)\text{성분} : \frac{\partial \hat{y}(t+3)}{\partial u(t+2)} = \frac{\partial f(x)}{\partial u(t+2)} = g_1$$

여기서, 일정한 규칙을 보이는 자코비안 행렬은 다음과 같다.

$$G = \begin{bmatrix} g_1 & 0 & 0 \\ g_2 & g_1 & 0 \\ g_3 & g_2 & g_1 \end{bmatrix}_{(3 \times 3)} \quad (16)$$

본 연구는 정보통신부의 '정보통신분야 우수학교 지원사업'의 연구비 지원으로 수행되었음.

[Reference]

- [1] G. Chen X. Dong, "From Chaos to Order-Perspectives and Methodologies in Controlling Chaos Nonlinear Dynamical Systems." *Int. Jour. of Bifurcation and Chaos*, Vol. 3, No. 6, pp. 1363-1409, 1993.
- [2] K. S. Narendra and S. Mukhopadhyay, "Adaptive Control Using Neural Networks and Approximate Models." *IEEE Trans. Neural Networks*, Vol. 8, pp. 475-485, 1997.
- [3] W. T. Miller, R. S. Sutton and P. J. Werbos, "Neural Networks for Control." Chap. 5, pp. 126-140.
- [4] K. S. Narendra and A. M. Annaswamy, "Stable Adaptive Systems." Chap. 1, pp. 13-28.
- [5] J. R. Noriega and H. Wang, "A Direct Adaptive Neural-Network Control for Unknown Nonlinear Systems and Its Application." *IEEE Trans. Neural Networks*, Vol. 9, pp. 27-34, 1998.
- [6] A. Draeger, S. Engell and H. Ranke, "Model Prediction Control Using Neural Network." *IEEE Control Systems*, October 1995.
- [7] D. W. Clarke, C. Mohtadi, and P. S. Tufts, "Generalized Predictive Control: Part I, the basic algorithm," *Automatica*, Vol. 23, No. 2, pp. 137-148, 1987.
- [8] S. H. Strogatz, "Nonlinear Dynamics and Chaos," Chap. pp. 429-453.
- [9] K. S. Park, J. B. Park, Y. H. Choi, G. Chen, "Generalized Predictive Control of Discrete-time Chaotic System," *Int. Jour. of Bifurcation and Chaos*, Vol. 8, No. 7, 1998.
- [10] S. M. Kim, Y. H. Choi and J. B. Park, "A Study on Adaptive Control of Chaotic Systems using RBFN," *Proc. of the 13th KACC*, pp. 365-368, 1998.