

論文2001-38SC-2-3

다중 모델을 이용한 비선형 시스템의 예측제어에 관한 연구

(Nonlinear Predictive Control with Multiple Models)

申承哲*, 卞增男**

(Seung-Chul Shin and Zeungnam Bien)

요 약

본 논문에서는 신경회로망 기반의 다중 모델을 이용한 예측제어 방법에 관하여 기술한다. 플랜트의 특정한 파라미터 값들에 대해 다중의 모델을 구성하고, 이들 중 현재 시간에서 최적의 예측 값을 제공하는 모델을 스위칭 기법으로 선택한다. 선택된 모델의 예측 값을 기반으로 비선형 프로그래밍 방법으로 현재 시간에서의 제어 입력 값을 구하여 예측제어를 수행한다. 제안한 방법을 시간지연 값이 변하거나 매개변수 값이 변하는 시스템에 적용하여 그 유용성을 보이고, 부하가 변동하는 자기부상열차 시스템의 부상제어에 이용한 모의 실험 결과를 보인다.

Abstract

In the paper, we propose a predictive control scheme using multiple neural network-based prediction models. To construct the multiple models, we select several specific values of a parameter whose variation affects serious control performance in the plant. Among the multiple prediction models, we choose one that shows the best predictions for future outputs of the plant by a switching technique. Based on a nonlinear programming method, we calculate the current process input in the nonlinear predictive control system with multiple prediction models. The proposed control method is shown to be very effective when a parameter of the plant changes or the time delay, if it exists, varies. It is also shown that the proposed method is successfully applied for the control of suspension in a electro-magnetic levitation system.

I. 서 론

예측제어 방법들은 선형 모델(Linear model)을 기반으로 이론적으로 정립되어, 지금까지 산업현장에서 많이 응용되어져 오고 있다^[1~4]. 하지만, 선형 모델을 기반으로 한 예측제어 방법들에서는 대상 플랜트가 심한

비선형성을 보일 경우 선형 모델의 근사화 능력의 한계로 인해 만족할 만한 결과를 보여주지 못한다. 따라서, 이러한 비선형 시스템에서의 예측제어를 위하여 퍼지학습망이나 신경회로망과 같은 학습기능을 가진 네트워크를 이용하고 있다^[5~7].

신경회로망을 기반으로 하는 예측제어 방법들에 있어서의 공통점을 보면, 우선 대상 플랜트의 가용할 수 있는 입출력 데이터를 이용하여 신경회로망을 학습시킨다. 학습된 신경회로망으로부터 플랜트의 미래의 출력 값을 예측하여, 이를 현재 및 미래의 입력 값을 결정하는 데 사용하고 있다. 또한 매 순간 변화하는 플랜트의 내부상황이나 외부환경에 적응하기 위하여 실시간으로 학습을 수행하기도 한다. 기존의 선형 모델을 이용한 예측제어 방법들이 아주 심한 비선형성을 나타

* 正會員, 韓國電子通信研究院 精報通信源泉技術研究所
(Telecommunication Basic Research Laboratory, ETRI)

** 正會員, 韓國科學技術院 電子電算學科
(Department of Electrical Engineering & Computer Science, KAIST)

接受日字:2000年4月14日, 수정완료일:2001年2月28日

공정 시스템들에서는 입력 값이 어느 정도의 시간 지연 후에 그 영향이 출력 값에 나타난다^[10]. 또한 어떤 시스템의 경우에는 시간 지연 값이 샘플링 주기에 대하여 정수배가 되지 않을 수도 있으며 변하는 경우도 있다. 천연가스 공급기에는 많은 파이프라인들이 설치되어 있다. 이들 중 하나의 파이프라인에 설치되어 있는 제어 밸브를 개방하는 경우 가정 혹은 공장으로 공급되는 최종 파이프라인에서의 가스흐름은 시간이 지연되어 나타난다. 이때, 가스관 내의 잔여 가스량에 따라서 지연시간이 다르게 나타나므로 이와 같은 시스템에 대해서는 특정의 시간지연 값들에 대한 다중 모델을 구성할 수 있다.

그림 1에서는 본 논문에서 제안한 신경회로망 기반의 다중 모델과 스위칭을 이용한 예측제어 방법의 전체적인 구조를 보인다. 실제 제어에서는 그림 1에서처럼 구성된 다중 모델들 중에서 하나의 모델을 선택하여 그 모델의 출력 값을 이용하여 제어 입력 값을 구한다. 하나의 모델을 선택하기 위한 스위칭 동작은 다음과 같은 순서로 행해진다.

1. 각 모델로부터 설정한 시간 윈도우(Time-window) T_w 구간에 대한 출력 값 $\hat{y}_i(t)=[\hat{y}_i(t)\hat{y}_i(t-1)\dots\hat{y}_i(t-T_w+1)]^T, i=1,2,\dots,p$ 를 구한다.
2. 시간 구간에서의 실제 플랜트의 출력 값 $y(t)=[y(t)y(t-1)\dots y(t-T_w+1)]^T$ 와 과정 1에서 구한 각 모델에서의 출력 값 $\hat{y}_i(t)$ 의 차로부터 각 모델에서의 예측 오차 $e_i(t)$ 를 구한다.

$$e_i(t)=y(t)-\hat{y}_i(t), i=1,2,\dots,p \quad (2)$$

3. 과정 2에서 구한 각 모델에서의 예측 오차에 대하여 다음과 같은 RMS(Root Mean Square) 값을 계산한다.

$$RMS_i(t)=\sqrt{\frac{1}{T_w}e_i(t)^T e_i(t)}, i=1,2,\dots,p \quad (3)$$

4. 과정 3에서 구한 $RMS_i(t), i=1,2,\dots,p$ 중에서 최저 값을 나타내는 모델을 현재 시간 t 에서 플랜트의 최적 모델로 선택한다.
5. 매 시간 또는 일정시간 주기로 위의 과정 1-4를 반복하여 그 순간에서의 최적의 예측 모델을 선택한다.

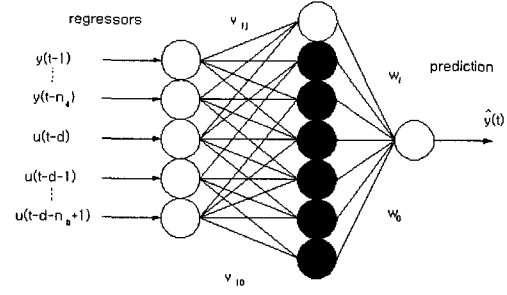


그림 2. 신경회로망의 구조 (흰색 원: 선형 노드, 검은색 원: 비선형 노드)

Fig. 2. Neural network structure (white circle: linear node, black circle: nonlinear node).

본 논문에서는 3층의 전방향 신경회로망(3-layered feed-forward neural network)을 이용하여 위에서 기술한 다중 모델을 구현한다. 그림 2에서처럼 구성된 신경회로망 모델에서는 선형 ARX 모델(Linear Auto-Regressive with eXogenous input model)을 포함하고 있음을 알 수 있다. 즉, 은닉층에서 하나의 선형 뉴런(Linear neuron)을 사용함으로써 이를 구현하고, 나머지의 비선형 뉴런(Nonlinear neuron)들과 이들에 연결된 웨이트(Weight)들을 이용하여 주어진 시스템의 비선형성을 근사화한다. 신경회로망의 출력 값은 다음 식 (4)와 같이 된다.

$$\begin{aligned} \hat{y}(t) = & -a_1y(t-1) - a_2y(t-2) - \dots - a_{n_a}y(t-n_a) \\ & + b_1u(t-d) + b_2u(t-d-1) + \dots + b_{n_b}u(t-d-n_b+1) \\ & + \sum_i w_i \sigma \left(\sum_j v_{ij} z_j \right) \end{aligned} \quad (4)$$

여기서 z_j 는 신경회로망의 입력으로 들어가는 벡터

$$z=[\hat{y}(t-1)\dots\hat{y}(t-n_a+1)u(t-d)\dots u(t-d-n_b+1)]^T \quad (5)$$

의 j 번째 값을 나타내고, d 는 시간지연 요소이고, n_a 와 n_b 는 각각 출력과 입력에서의 최대 지연요소(Maximum lag)를 나타낸다. w_i 와 v_{ij} 는 출력단과 입력단에서의 웨이트를 표시하고, σ 는 ± 1 사이의 값을 갖는 시그모이드(Sigmoid) 함수를 나타낸다. 여기서 선형 모델에 대한 웨이트 값은 $W_{Lim}=[-a_1\dots-a_{n_a}b_1\dots b_{n_b}]^T$ 와 같이 표현할 수 있다.

위와 같은 신경회로망 기반의 모델을 구성할 때 파라미터인 n_a 와 n_b , 그리고 시간 지연을 나타내는 d 값을 원래 시스템에서의 값(여기서는 이 값들을 n_a' , n_b' , d' 으로 표기한다)과 정확히 일치되도록 설정하면 가장 좋은 예측 값을 나타낸다^[11, 12]. 즉, 이와 같이 설정을 하면 신경회로망의 크기를 줄일 수 있을 뿐만 아니라 매우 정확한 예측 값을 구할 수 있다. 그러나, 이러한 설정에 있어서의 문제점은 시간지연 요소가 변화하는 시스템의 경우에는 시간지연에 대한 보상을 해줄 수가 없다는 것이다. 즉, 주어진 시스템의 시간지연의 변화 범위가 $1 \leq d' \leq d + n_b$ 를 만족할 경우에는 실시간 적응(On-line adaptation) 알고리즘을 구현하면 어느 정도 좋은 결과를 보이지만, $d' > d + n_b$ 의 경우에는 실시간으로 학습을 수행하더라도 시스템의 출력을 잘 예측할 수 없게 되어 좋지 않은 결과를 보이게 된다.

그림 3에서는 아래의 시스템^[8, 11, 12]에서 시간지연 요소 d' 이 변하는 경우 위의 사항들을 고려하여 구성된 모델에서의 예측 오차를 관찰한 것이다.

$$y(t) = \frac{y(t-1)y(t-2)y(t-3)u(t-d'-1)(y(t-2)-1)+u(t-d')}{1+y(t-2)^2+y(t-3)^2}$$

여기서 d' 은 초기에 1로 설정되어 있었으나, 매 100 샘플링 시간마다 1에서 2, 3, 7, 1로 변하는 값으로 설정된다. 그림 3(a)에서는 시스템의 정확한 매개변수 값 ($n_a' = 3, n_b' = 2, d' = 1$)으로 신경회로망을 구성한 경우 즉, $n_a = n_a', n_b = n_b', d = d'$ 으로 설정 하여 $z = [y(t-1)y(t-2)y(t-3)u(t-1)u(t-2)]^T$ 와 같이 구성하고, 매 100샘플마다 시간지연이 바뀌는 상황에 있어서의 임의의 입력 값에 대한 예측 오차를 보인 것이다. 그림 3(b)에서는 시스템의 시간지연의 변화 범위 ($1 \leq d' \leq 7$ 를 고려하여 신경회로망의 입력 벡터를 $z = [y(t-1)y(t-2)y(t-3)u(t-1)\dots u(t-6)]^T$ 로 설정한 경우 ($n_b = 5$)에 있어서의 예측 오차를 나타낸 것이다. 여기에서 보면, 시스템의 시간지연과 신경회로망 모델에서의 시간지연 요소가 정확하게 일치하는 경우 ($d = d'$)인 $0 \leq t < 100$ 과 $400 \leq t \leq 500$ 구간에서는 신경회로망 모델에서 예측 오차가 거의 0에 가까운 결과를 보이고, 시간지연이 다른 경우 ($d \neq d'$)인 $100 \leq t < 400$ 구간에서는 예측 오차가 매우 커지는 것을 볼 수 있다. 그림 3(c)에서는 (b)의 상황에서 실시간으로 웨이트 값을 갱신하는 경우에 대한 결과를 나타낸 것이다. 여기에서

보면, 시스템의 시간지연의 변화 범위가 모델에서의 $d + n_b$ 값보다 작은 경우에 대해서는 실시간 적응에 의해 예측 오차가 줄어드는 것을 볼 수 있다. 그러나, 시간지연의 크기가 이 범위를 벗어나는 경우 ($d' = 7$)인 경우에 대해서는 큰 예측 오차를 보임을 알 수 있다. 표 1에서는 각 시간지연 요소 구간에서의 예측 오차의 자승의 합(SSE, Summation of Squared Error)을 나타낸다.

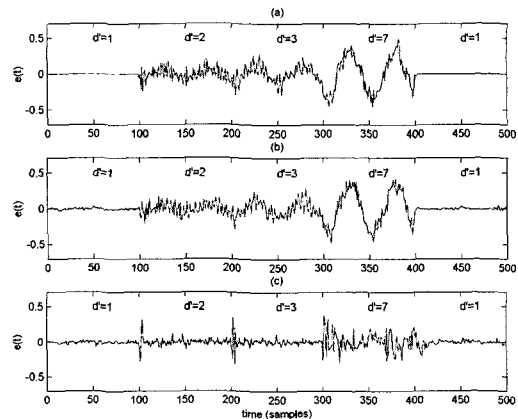


그림 3. 시간지연 요소의 변화에 따른 학습오차 비교 (a) 정확한 n_b 값으로 신경회로망을 구성한 경우, (b) 가변 시간지연을 고려하여 n_b 값을 크게 설정한 경우, (c) (b)의 경우에 실시간 적응 알고리즘을 고려한 경우

Fig. 3. Comparison of training error according to the time-delay variation (a) set n_b as n_b' in the neural network structure, (b) set $n_b > n_b'$, (c) an on-line adaptation algorithm is applied to the case of (b).

표 1. 시스템의 시간지연 요소 변화에 따른 예측 오차 제곱의 합

Table 1. Summation of SSE with respect to the time-delay variation.

SSE	$d'=1$	$d'=2$	$d'=3$	$d'=7$	$d'=1$
(a)	0.0002	0.7261	1.2140	5.9907	0.0008
(b)	0.0113	0.6991	1.1951	6.2189	0.0121
(c)	0.0297	0.3246	0.4228	1.6143	0.1866

이와 같은 플랜트의 시간지연 요소의 변화를 신경회로망 기반의 제어방법에서 처리하기 위하여 Narendra와 Mukhopadhyay^[8]는 매 시간지연 요소에 대한 모델을 구성하고 스위칭으로 이들 중 한 모델을 선택하도록

록 하는 방법을 제안하였다. 그러나, 이 방법은 모델의 개수가 시간지연 요소에 따라 증가하게 되어 신경회로망의 크기가 매우 크게 되는 단점이 있다. 따라서, 본 논문에서는 그림 4에서와 같이 이웃하는 모델 사이의 시간지연 n (정수)을 1이 아니라, $m \geq 1$ 으로 설정하여 신경회로망의 크기를 줄이는 한편 예측 오차도 학습을 통해 줄이도록 한다. 예를 들어, $d=d'$ 스텝의 가변 시간지연을 갖는 어떤 시스템에 대하여 다중 모델을 구성하는 것을 고려해 보자. 그림 4(a)와 같은 방법으로는 총 $[(n_a + n_b + 1) \times h + (h + (h + 1)) \times d]$ 개의 웨이트들을 포함하는 신경회로망이 된다. 반면, 그림 4(b)에서는 총 $[(n_a + n_b + 1) \times h + (h + 1)] \times (\frac{d}{m} + 1)$ 개의 웨이트가 필요함을 알 수 있다. 여기서는 $1 \leq m \leq n_b$ 를 가정한다. h 는 신경회로망에서의 은닉층(Hidden layer)의 개수를 나타낸다. 만약, $d=10, m=3, n_a=3, n_b=5$ 이고, $h=5$ 인 경우에서 보면, 그림 4(a)와 (b)에서 소요되는 총 웨이트의 수는 각각 510개와 약 220개가 됨을 알 수 있다.

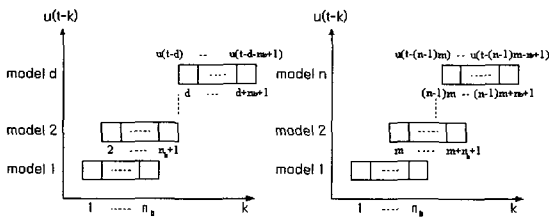


그림 4. 시간지연 요소의 변화에 따른 신경회로망의 입력 벡터값 결정 (a) Narendra와 Mukhopadhyay의 다중 모델, (b) 제안한 방법
 Fig. 4. Input vector determination methods in the neural network in the presence of time-delay variations (a) Narendra and Mukhopadhyay's method, (b) proposed method.

다른 하나의 방법으로 주어진 시스템의 가변할 수 있는 최대 시간지연 요소를 포함하도록 신경회로망의 입력 노드(Input node)를 구성할 수 있다. 이렇게 구성한 경우에 대해서는 앞의 예에서 그림 3(a)와 (b), (c)를 비교함으로써 그 영향을 파악할 수 있다. 표 1의 $d'=1$ 인 경우에서 보면, 시스템의 가변 시간지연을 고려하여 모델을 구성한 경우(그림 3(b)와 (c), $d=d', n_a = n_a', n_b > n_b'$)에는 실제로 시스템의 출력과는 상관

없는 입력항들의 영향으로 예측오차 제곱 값들의 합이 정확하게 시간지연과 입출력항의 개수를 고려하여 신경회로망을 구성한 경우(그림 3 (a), $d=d', n_a = n_a', n_b = n_b'$)보다는 상대적으로 매우 크게 나타남을 알 수 있다. 따라서, 이와 같이 예측 모델을 구성한 경우에는 실제 시스템의 출력에 관계하지 않는 입력항들이 모델의 구성에 포함되어짐으로써 시스템의 출력 값을 예측하는 데 오차가 생기는 것을 볼 수 있다.

다중 모델들은 사용 가능한 입출력 데이터들을 이용하여 각각의 매개변수 값에서의 시스템 특성을 파악하도록 학습되어진다. 본 논문에서의 신경회로망의 학습은 다음과 같은 식 (6)을 최소화하는 웨이트 값을 구하도록 하는 방향으로 이루어진다.

$$V(\theta, Z^P) = \frac{1}{2P} \sum_{t=1}^P (y(t) - \hat{y}(t))^T (y(t) - \hat{y}(t)) \quad (6)$$

여기서 Z^P 는 학습에 이용한 P 개의 입출력 데이터 쌍을 나타내고, θ 는 신경회로망의 모든 웨이트를 벡터의 형태로 표시한 것을 나타낸다. 식 (6)에서 보면, $V(\theta, Z^P)$ 값이 신경회로망 내의 비선형 뉴런들에 의해 θ 에 대해 비선형적으로 연결되어 있음을 알 수 있다. 따라서, 이 식으로부터 직접적으로 최적의 θ 값을 유도하는 것은 매우 어렵기 때문에 반복적인 방법(Iterative method)으로 해를 구해 나간다. 즉, 시스템에서의 웨이트 값은 식 (7)과 같이 결정된다.

$$\theta^{(i+1)} = \theta^{(i)} + \mu^{(i)} f^{(i)} \quad (7)$$

여기서 $f^{(i)}$ 는 i 스텝에서의 방향 벡터를 표시하고, $\mu^{(i)}$ 는 스텝의 크기를 나타낸다. 방향 벡터를 결정하는 방법들로는 여러 가지가 있으나, 본 논문에서는 Gauss-Newton 기반 Levenberg-Marquardt 방법^[13]을 채택하여 식 (6)의 비용함수를 최소화하는 방향으로 웨이트 값들이 결정되도록 한다.

III. 비선형 예측제어

본 장에서는 신경회로망 기반의 다중 모델에 대한 비선형 예측제어 방법에 대하여 기술한다.

예측제어 방법들에 있어서의 목표는 식 (8)과 같이

주어지는 가격함수를 최소화하는 미래의 입력벡터 $U(t)=[u(t)u(t+1)\dots u(t+N_u-1)]^T$ 를 구하는 것이다.

$$J(U) = \sum_{n=N_1}^{N_2} [r(t+n) - \hat{y}(t+n)]^2 + \rho \sum_{n=1}^{N_u} [\Delta u(t+n-1)]^2 \quad (8)$$

이때 다음 식 (9)의 제약 조건을 만족하도록 설계된다.

$$\Delta u(t+n) = 0, \quad n \geq N_u \quad (9)$$

여기서 $r(t+n)$ 값은 현재의 시간 t 에서 n 번째 스텝 이후의 기준 입력 값을 나타내고, $\hat{y}(t+n)$ 은 앞 장에서 말한 바와 같이 선택된 신경회로망의 출력 값으로서 n 스텝 이후의 시스템의 출력에 대한 예측 값을 나타내며 식 (10)과 같이 표현된다.

$$\hat{y}(t+n) = W_{Lin}^T \cdot z^n + \sum_i w_i \sigma \left(\sum_j v_{ij} z_j^n \right) \quad (10)$$

여기서

$$z^n = [\hat{y}(t+n-1) \dots \hat{y}(t+n-n_a+1) u(t+n-d) \dots u(t+n-d-n_b+1)]^T \quad (11)$$

로 구성된다. N_1 은 최소 예측구간(Minimum prediction horizon)을 나타내고, N_2 는 최대 예측구간(Maximum prediction horizon)을 표시하며, N_u 는 고려하고자 하는 미래 입력의 범위(Control horizon)를 나타낸다. ρ 는 가중치(Control weighting factor)로서 미래의 제어 입력 값의 변화량의 합과 미래의 예측 오차의 합 사이의 비중을 결정하기 위하여 사용된다. 일반적으로 시스템의 시간지연 요소가 정확하게 알려져 있지 않은 경우에는 모든 가능성을 고려하기 위하여 N_1 을 1로 설정한다. 또한, 시스템의 입력항 최대 시간지연 요소(n_b')가 얼마인지를 정확히 알 수 없는 경우에는 N_2 를 n_b' 의 값보다 충분히 크게 되도록 설정한다^[14].

식 (10)과 (11)을 보면, n 스텝에서의 예측 값 $\hat{y}(t+n)$ 를 구하는 데 그 이전에 예측된 값, 즉 $\hat{y}(t+n-1), \hat{y}(t+n-2), \dots$ 와 같은 값들이 이용되고 있음을 알 수 있다. 이와 같은 예측모델의 구성은 이미 잘 알려져 있으며^[15, 16], 현재의 입력 및 미래의 입력벡

터 $U(t)$ 와 미래의 출력 값들이 신경회로망 내부의 비선형 뉴런함수에 의해 (8)과 (10)에서처럼 매우 복잡한 비선형 관계가 있음을 알 수 있다. 이러한 비선형 관계 식부터 닫힌 형태(Closed-form)의 미래 입력벡터를 구하는 것은 거의 불가능하다. 따라서, 아래의 식 (12)에서와 같은 최적의 입력벡터 U^* 를 비선형 프로그래밍 방법(Nonlinear programming method)^[6, 7]이나 유전자 알고리즘(Genetic algorithm)^[12, 16] 등을 이용하여 구해 오고 있다.

$$U^* = \arg \min_U J(U) \quad (12)$$

본 논문에서는 비선형 프로그래밍 방법들 중에서도 비교적 수행 속도가 빠른 BFGS^[17] 방법으로 현재 및 미래의 입력 값들을 구한다.

IV. 모의 실험 결과

본 장에서는 우선 스프링-질량 동역학 시스템의 시간지연에 따른 제어 결과를 보인다. 그리고, 시스템의 매개변수 변화에 따른 기존의 하나의 모델을 이용한 예측제어 방법과 제안한 다중의 예측 모델을 이용하는 방법에서의 제어 결과를 비교한다. 또한 제안한 방법을 시간적으로 부하의 변동이 있는 자기부상열차의 부상 제어에 적용한 결과를 상태환(State feedback) 제어 시스템의 결과와 비교한다.

1. 시간지연 요소의 변화에 대한 비선형 예측제어 대상으로 한 시스템의 동역학식은 다음과 같다.

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= -a \cdot x_1 - x_2 - x_1^3 + u, \\ y &= x_1. \end{aligned}$$

여기서 $a=1,0$ 으로 설정되고, 입력 u 가 $u(t-d \cdot T_s)$ 의 형태로 샘플링 주기 $T_s=0.2$ 초의 d' 정수배로 시간이 지연되어 시스템으로 인가된다. 시간 지연은 매 10초마다 $d'=1,2,5,4,2$ 와 같이 변한다.

이와 같은 시스템에 대하여 $n_a=2, n_b=4, m=3$ 으로 설정하여 두 개의 신경회로망 모델을 구성한다. 두 모델의 입력 벡터는

$$\begin{aligned} z_1 &= [y(t-1)y(t-2)u(t-1)u(t-2)u(t-3)u(t-4)]^T, \\ z_2 &= [y(t-1)y(t-2)u(t-3)u(t-4)u(t-5)u(t-6)]^T \end{aligned} \quad (13)$$

와 같이 구성한다. 구성한 두 모델에 대하여 예측제어기에서의 파라미터 값들은 $N_1=1, N_2=7, \rho=0.03$ 으로 설정하고, 스위칭을 위한 시간 윈도우는 $T_w=10$ 으로 하였다. 여기서 T_w 의 값을 지나치게 작게 설정하면 스위칭이 자주 일어나게 되고, 너무 크게 설정하면 상대적으로 스위칭이 적게 일어나 시스템의 변화에 빨리 적응해 나갈 수 없게 된다. 따라서 T_w 의 값은 시스템의 특성을 분석하여 이에 따라 적절히 설정해야 하며 두 영향에 대해 Trade-off가 이루어져야 한다.

그림 5에서는 하나의 신경회로망 모델에서 입력벡터를 $z=[y(t-1)y(t-2)u(t-1)u(t-2)]^T$ 와 같이 설정하여 제한한 결과를 보인다. 여기서 점선은 기준 입력을 나타내고, 실선은 시스템의 출력을 나타낸다. 이 결과에서 보면 시간 지연 d' 이 2이하인 경우에는 좋은 결과를 보여주나, 2보다 큰 경우($d' > n_b=2$)에 대해서는 좋은 결과를 주지 못함을 알 수 있다. 이 때의 전체 모의 실험 구간에 대한 RMSE(Root Mean Square Error) 값은 0.2916이었다. 그림 6에서는 식 (13)에서와 같이 입력 벡터를 설정한 두 개의 신경회로망 모델과 스위칭을 이용한 경우의 모의 실험 결과를 보인다. 그림 6에서 보이는 것과 같이 시스템의 출력(실선)이 시간지연의 변화에도 불구하고 기준 입력(점선) $r(t)$ 를 전체 모의 실험 구간에서 RMSE=0.1163의 값으로 충실히 따라가는 것을 볼 수 있다. 그리고 다중 모델을 이용한 경우 시스템의 시간지연 요소의 변화가 1에서 4까지의 범위 내에 있을 때에는 신뢰할 만한 제어출력을 보인 반면 이보다 큰 변화에 대해서는 좋지 못한 결과를 보였다. 그림 6(b)에서는 이 때의 오차 값(실선)과 스위칭 동작에 의해 선택된 모델의 번호가 점선 위에 숫자로 표시되어져 있다. 시간 윈도우의 크기를 $T_w=10$ 으로 설정하였으므로 스위칭이 비교적 자주 일어나는 것을 볼 수 있다.

2. 파라미터 변화에 따른 스위칭

이 경우에는 4.1절에 주어진 동역학식에서 매개변수 a 의 값이 처음 15초 동안에는 1.0 값을 유지하다가 이후 매 15초마다 -2.0, 1.0, -3.0, 1.0의 값으로 변하는 경

우에 대하여 스위칭 모드를 도입하여 제한한 결과를 보인다. 다중 모델은 $a=1.0, -2.0, -3.0$ 에 대하여 각각 모델 1, 2, 3으로 구성하였고, 각 모델의 입력 벡터는 $z_i=[y(t-1)y(t-2)u(t-1)u(t-2)u(t-3)u(t-4)]^T$ 와 $i=1,2,3$ 같다. 예측제어기의 파라미터 값들은 4.1절에서와 동일하게 설정한다.

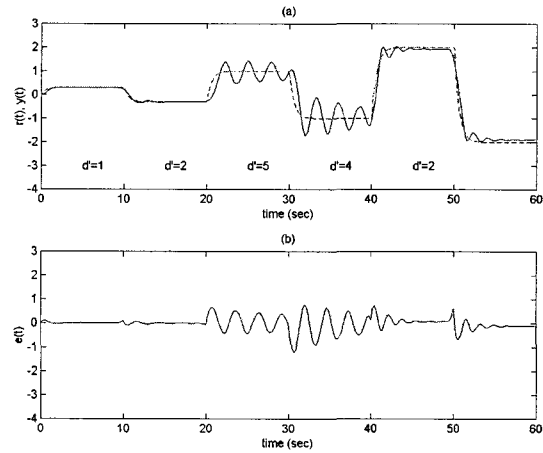


그림 5. 하나의 예측모델을 이용한 경우의 시간지연 변화에 따른 비선형 예측제어의 결과

Fig. 5. Nonlinear predictive control results when only one model is used in the presence of time-delay variations.

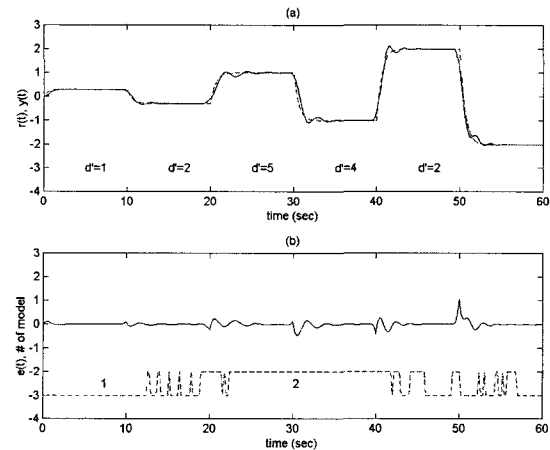


그림 6. 다중 모델을 이용한 경우의 시간지연 변화에 따른 비선형 예측제어의 결과

Fig. 6. Nonlinear predictive control results when multiple models are used in the presence of time-delay variations.

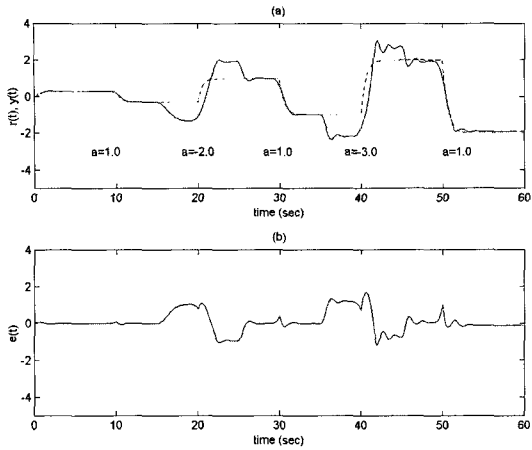


그림 7. 파라미터 변화에 대해 하나의 모델을 학습없이 이용하여 제어한 경우

Fig. 7. Nonlinear predictive control results by using one model with no online learning scheme in the presence of parameter variations.

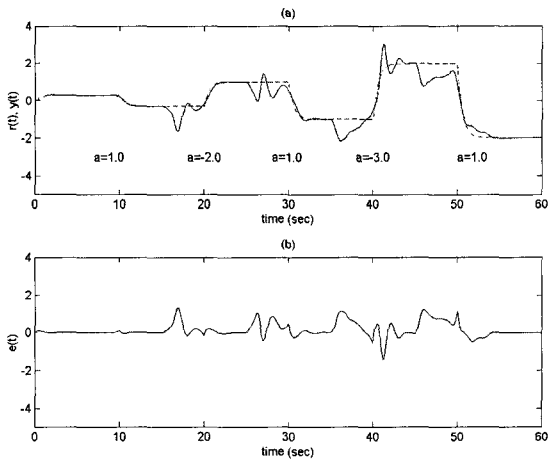


그림 8. 파라미터 변화에 대해 하나의 모델을 학습시키면서 제어한 경우

Fig. 8. Nonlinear predictive control results with an online learning scheme in the presence of parameter variations.

그림 7에서는 하나의 모델 ($a=1.0$)을 이용하여 온라인 학습을 시키지 않고 예측제어를 수행한 결과를 보인다. 그림 7(a)에서 보면 플랜트의 출력(실선)이 파라미터 값이 변하지 않는 동안에는 설정 값(점선)을 잘 따라가지만, 파라미터 값이 변하는 경우에는 좋지 않은 결과가 나타남을 알 수 있다. 그리고 같은 그림에서 매 시간 구간에서의 파라미터 값을 표시하였다. 그림 7(b)에서는 이 때의 오차 값을 나타내고, 전체 구간에서의 RMSE 값은 0.5472이다.

그림 8에서는 위의 모델에 대하여 온라인 학습을 수행한 경우에 대한 결과를 보인다. 이 경우에 있어서도 파라미터 값이 변하는 순간에서 입력 및 출력에서 큰 흔들림이 발생하는 것을 볼 수 있으며 전체 구간에서의 RMSE 값은 0.4033이다.

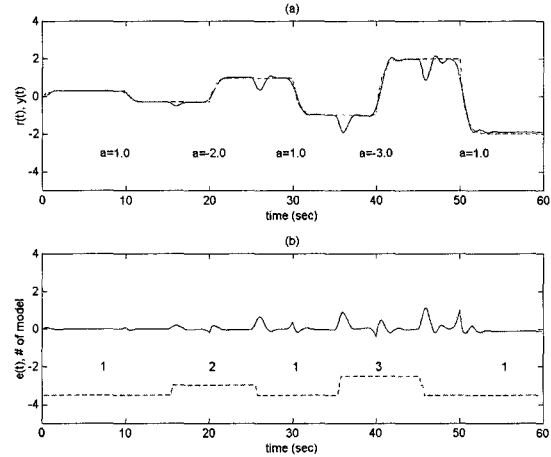


그림 9. 파라미터 변화에 대해 세 모델을 스위칭하며 제어한 경우

Fig. 9. Nonlinear predictive control results by using 3 models in the presence of parameter variations.

그림 9(a)에서는 위에서 구성한 3개의 신경회로망 모델들을 기반으로 스위칭을 통하여 제어한 결과를 보인다. 여기서 실선은 플랜트의 출력 $y(t)$ 를 나타내고, 점선은 기준 입력 $r(t)$ 를 나타낸다. 그림 9(b)에서는 이 때의 오차 값(실선)과 스위칭에 의해 선택된 모델의 번호를 점선 위에 보이고 있으며, 전체 구간에서의 RMSE 값이 0.2154로서 앞의 두 경우보다 오차가 줄어들음을 확인할 수 있다. 이 모의 실험에서 구성한 3개의 모델을 이용한 경우에는 플랜트의 매개변수 a 값이 대략 $-5 \leq a < 10$ 의 범위 내에서 변화할 때에 대하여 신뢰할 만한 결과를 보여 주었다.

다음에서는 제안한 방법을 자기 부상열차의 부상제어에 적용한다. 대상으로 한 자기부상열차는 단축형 (Single-axis type)으로 다음과 같은 동역학식으로 표현된다.

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= -\frac{\mu_0 N^2 A}{4m} \left(\frac{i}{x_1} \right)^2 + g \\ y &= x_1. \end{aligned}$$

여기서 x_1 은 수직방향의 air-gap이고, x_2 는 수직방향의 속도, i 는 자기전류 (Magnetic current)를 나타낸다. 대상으로 한 부상열차의 사양¹⁸⁾은 다음과 같다. m 은 부상열차의 총 무게(부하가 없는 경우 $m = m_0 = 300Kg$)이고, $N = 600$ 은 철심에 감긴 코일의 수를 나타낸다. $A = 0.05m^2$ 는 철심의 단면적을 나타내고, $\mu_0 = 4\pi \times 10^7 H/m$ 는 공기 중에서의 투자율이다. $R = 1\Omega$ 은 코일의 저항 값이고, $g = 9.8m/s^2$ 는 중력가속도를 나타낸다. 부상제어를 위한 시스템의 입력은 부상을 위하여 인가한 전류 i 이고, 출력 y 는 고정된 트랙과 열차 사이의 간격(air-gap)이다. 그림 10에서는 부상시스템의 간략한 구조를 보인다.

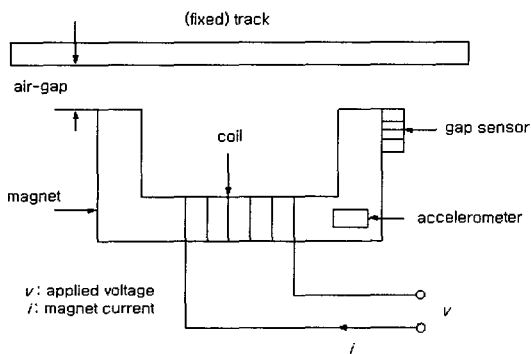


그림 10. 자기부상 시스템의 구조
Fig. 10. Electromagnetic suspension system.

자기부상 시스템은 불안정한 개루프(Open-loop) 특성을 보이기 때문에 입력에 임의의 신호를 인가하여 신경회로망의 학습을 위한 데이터를 구하는 것은 매우 힘들다. 따라서, 본 논문에서는 상폐환 제어를 설계하여 우선 안정한 폐루프 회로를 구성한 후 인가하는 기준 신호(Reference signal)를 가변시킬 때 나타나는 자기부상열차의 입출력 데이터를 이용하여 2장에서 기술한 방법으로 신경회로망을 학습시킨다. 여기서의 샘플링 간격은 0.002초이다.

그림 11에서는 기존의 상폐환 제어를 이용한 결과를 보인다. 그림 11(a)에서 보면, 부상열차의 무게 변화가 없는 경우에는 제어된 air-gap(실선)이 설정한 air-gap(점선)을 잘 따라가는 결과를 보여주나, 1초 이후에 부상시스템의 무게가 300 Kg에서 900 Kg으로 변하면 큰 오프셋(Offset) 현상이 나타나는 것을 볼 수 있

다. 이때 인가된 자기전류는 그림 11(b)에 나타나 있다. 제안한 방법에서는 300 Kg과 900 Kg의 무게에 대하여 각각의 모델을 구성하고 스위칭 기법을 이용한다. 파라미터 값은 $T_w = 10$, $N_1 = 1, N_2 = 10, \rho = 0.01$ 와 같고, 이 때의 결과를 그림 12(a)에 보인다. 1초에서 부상열차의 부하의 무게가 변화하지만 부상 결과(실선)는 설정한 air-gap(점선)을 아주 잘 따라가고 있음을 볼 수 있다. 매순간 스위칭에 의해 선택된 모델의 번호를 그림 12(a)의 윗 부분에 일점쇄선과 숫자로 표시하였다. 그림 12(b)에서는 이때의 자기전류의 변화를 나타낸다.

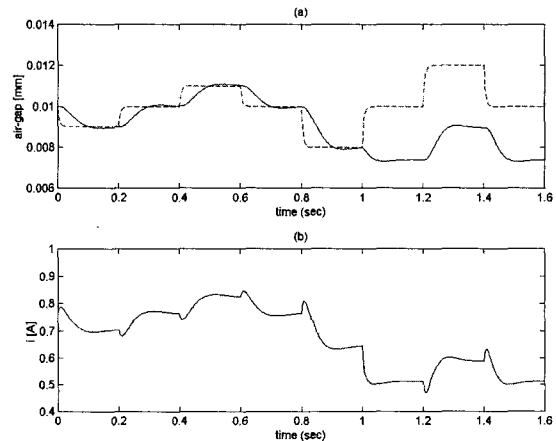


그림 11. 자기부상 시스템의 상폐환 제어 결과
Fig. 11. State feedback control results in a MAGLEV system.

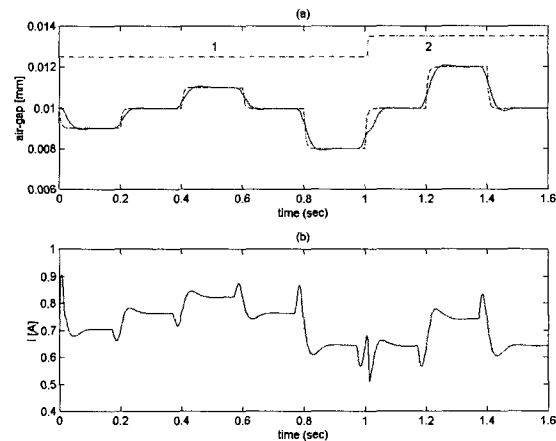


그림 12. 제안한 방법으로 자기부상 시스템을 제어한 결과
Fig. 12. MAGLEV system control results by using multiple models.

V. 결 론

본 논문에서는 하나의 예측 모델을 이용하는 신경회로망 기반의 비선형 예측제어 방법들에 있어서의 시간 지연 요소의 가변과 플랜트의 매개변수 변화에 대한 문제점을 해결하고자 다중의 모델을 구성하고 이들을 스위칭 기법으로 선택하는 방법을 제안하였다. 이러한 방법을 이용한 경우 주어진 시스템의 시간지연 요소나 매개변수의 변화에 대해서 신뢰할 만한 제어 결과를 보임을 확인하였다. 그러나, 제안한 방법에서는 다중 모델의 구성과 스위칭 기법의 구현이 필요하므로 기존의 방법들보다는 더 복잡한 구조를 가지고 있는 단점이 있고, 스위칭에 따른 안정성 해석이 필요하다. 또한, 스위칭할 때의 시간 윈도우의 크기를 적절히 설정하는 것이 필요하다. 다양한 모의 실험의 경험에 비추어 보면, 일반적으로 시스템의 변화가 급격히 변화하는 경우에는 이 값을 작게 하고 천천히 변화하는 플랜트의 경우에는 이 값을 크게 하는 것이 제어성능을 향상시키는 것을 알 수 있었다.

참 고 문 헌

[1] E.F. Camacho, F.R. Rubio, and F.M. Hughes, "Self-tuning control of a solar plant with a distributed collector field", *IEEE Control Systems*, pp. 72-78, April 1992.

[2] D.W. Clarke, "Application of generalized predictive control to industrial processes", *IEEE Control Systems*, pp. 49-55, April 1988.

[3] G. Geng and G.M. Geary, "Application of a neural network-based RLS algorithm in the generalized predictive control of a nonlinear air-handling plant", *IEEE Transactions on Control Systems Technology*, vol. 5, pp. 439-445, 1997.

[4] M. Fischer, M. Schmidt, and K. Kavsek-Biasizzo, "Nonlinear predictive control based on the extraction of step response models from Takagi-Sugeno fuzzy systems", *Proceedings of ACC*, pp. 2878-2882, June 1992.

[5] S.C. Ahn, Y.H. Kim, and W.H. Kwon, "A fuzzy generalized predictive control using affine fuzzy predictors for nonlinear systems", *Journal of Intelligent and Fuzzy Systems*, vol. 6, pp. 185-207, 1998.

[6] A. Draeger, S. Engell, and H. Ranke, "Model predictive control using neural networks", *IEEE Control Systems*, vol. 15, pp. 61-66, 1995.

[7] M. Norgaard, P.H. Sorensen, and N.K. Poulsen, "Intelligent predictive control of nonlinear processes using neural networks", *Proceedings of IEEE Int. Symp. Intelligent Control*, pp. 301-306, 1996.

[8] M.M. Gupta and N.K. Sinha, *Intelligent control systems: theory and applications*, IEEE Press, 1996.

[9] P.K. Sinha, *Electromagnetic Suspension: dynamics and control*, Peter Peregrinus Ltd., London, 1987.

[10] E.F. Camacho and C. Bordons, *Model predictive control in the process industry*, Springer-Verlag, 1995.

[11] X. He and H. Asada, "A new method for identifying orders of I/O models for nonlinear dynamical systems", *Proceedings of ACC*, pp. 2520-2523, 1993.

[12] S.C. Shin and Z. Bien, "Constrained GA-based predictive control", *Proceedings of IEEE Fall Conf.*, vol. 22, pp. 732-735, Nov. 1999.

[13] L.O. Ljung, *System identification: theory for the user*, Prentice Hall, 1987.

[14] D.W. Clarke, C. Mohtadi, and P.S. Tuffs, "Generalized predictive control: Part I, and Part II", *Automatica*, vol. 23, pp. 137-160, 1987.

[15] J. Saint-Donat, N. Bhat, and T.J. McAvoy, "Neural net based model predictive control", *Int. J. Control*, vol. 54, pp. 1453-1468, 1991.

[16] S.C. Shin and S.B. Park, "GA-based predictive control for nonlinear processes", *Electronics Letters*, vol. 34, pp. 1980-1981, 1998.

[17] J.E. Dennis and J.J. More, "Quasi-Newton methods, motivation and theory", *SIAM*

Review, vol. 19, pp. 46-89, 1977.

- [18] S.H. Lee, H.K. Sung, J.T. Lim, and Z. Bien,
"On self-tuning control of electromagnetic
levitation systems", *Control Engineering
Practice*, vol. 8, pp. 749-756, 2000.

저 자 소 개

申承哲(正會員) 第 32卷 B編 第 8號 參照
현재 한국전자통신연구원 선임연구원

下增男(正會員) 第 37卷 第 11號 參照
현재 한국과학기술원 전자전산공학과 교수