GMDH 알고리즘을 이용한 모델링 및 제어에 관한 연구

A Study on the Modelling and Control Using GMDH Algorithm

최 종 헌*, 홍 연 찬*
Jong Hun Choi*, Yeon Chan Hong*

※이 연구는 1994년도 전반기 한국과학재단 해외 Post-Doc. 연수 지원에 의한 결과임.(접수번호 : 94-08-33-1)

요 약

신경 회로망의 출현으로 비선형 시스템 모델링에 대한 관심이 다시 고조되고 있다. 따라서 본 논문에서는 미지의 비선형 시스템을 동적으로 인식하기 위해 GMDH(Group Method of Data Handling) 알고리즘을 사용한 DPNN(Dynamic Polynomial Neural Network)을 제안한다. GMDH를 사용한 동적 시스템의 인식은 일련의 입/출력 데이타를 인가하여 필요한 계수들의 집합을 동적으로 산출함으로써 회로망을 훈련시킨다.

또한 DPNN을 이용하여 비선형 시스템을 제어하기 위해, MRAC(Model Reference Adaptive Control)를 설계한다. 결과에서는 컴퓨터 시뮬레이션을 통해 DPNN을 사용한 모델링과 제어가 잘 수행됨을 알 수 있었다.

ABSTRACT

With the emergence of neural network, there is a revived interest in identification of nonlinear systems. So in this paper, to identify unknown nonlinear systems dynamically we propose DPNN(Dynamic Polynomial Neural Network) using GMDH (Group Method of Data Handling) algorithm. The dynamic system identification using GMDH consists of applying a set of input/output data to train the network by dynamically computing the necessary coefficient sets.

Then, MRAC(Model Reference Adaptive Control) is designed to control nonlinear systems using DPNN. In the result, we can see that the modelling and control using DPNN work well by computer simulation.

I. 서 론

A. G. Ivakhnenko에 의하여 고안된 GMDH(Group Method of Data Handling) 알고리즘은 어떻게 하면 다항식 범함수(polynomial functional) 형태를 측정된 데이타에 적용할 수 있을까 하는 문제의 해결책으로 대두되었으며, 바로 PNN(Polynomial Neural Network)의 일반적인 알고리즘이다[1, 2]. GMDH 알고리즘은 패턴인식, 신호처리, 그리고 시스템 모델링 등과 같은 여러분야에서 사용되고 있다. 이 GMDH 알고리즘은

*인천대학교 전자공학과

극소수의 원시적이며 단순한 형태의 방정식으로부터 출발하여, 이전의 세대들로부터 합성되어진 복잡하고 다양한 자손들(complex off-springs)로 구성되는 새로운 세대들(new generations)을 생성시켜 나간다. 이러한 새로운 자손들은 그 다음 세대로 넘어가기 전에 적자 생존의 원칙(the survival-of-the-fittest principles)이 적용되며, 이때 살아 남은 것들을 합성하여 다음 세대를 만들어 가는 것이다. 그리고 이러한 합성 과정을 진행하는 동안, 모델링 대상이 되는 시스템을 가장 잘 표현 할 수 있다고 판단되어지는 세대에서 그 합성을 멈추게 되는 것이다.

한편, 과거 30여년간의 다이나믹 시스템들에 대한 제어기 설계에 있어서의 주요한 발전은 선형 대수학, 복소 변수 이론(complex variable theory), 그리고 상미분 방정식(ordinary differencial equation)과 같은 선형 시스템을 대상으로 하여 진행되어 왔다. 또한 이러한 이론들은 선형 시스템에 대하여 안정성(stability), 강인성(robustness), 그리고 바람직한 다이나믹 응답 특성들을 동시에 만족시킬 수 있었다. 그러나 비선형 시스템에 있어서는 선형 시스템에서 사용된 기존의 수학적 기법들로는 제어가 불가능했으며, 비선형 시스템에 대한 수학적 표현식 또한 매우 복잡하여 제어가 용이하지 않았다. 결국 기존의 방법 이외의 또 다른 수학적 기법들에 대한 필요성을 느끼게 되어 대두된 것 중에 하나가 바로 신경 회로망이다. 특히, 다층 퍼셉트론(multilayer perceptron)을 기초로한 신경 회로망 알고리즘이 이러한 비선형 시스템 인식 및 제어에 사용되고 있다.

본 논문의 주 목적을 다음과 같이 크게 두가지로 요약할 수 있다. 첫째, PNN 분야에서 이용되고 있는 GMDH 알고리즘을 사용하여 미지의 비선형 다이나믹 시스템(unknown nonlinear dynamical system)을 모델링하는 것이다. 둘째, 모델링된 시스템을 기준모델 적응제어에 적용하여 제어 목적을 달성하고자 하는 것이다. 즉, DPNN(Dynamic PNN)을 구현함으로써 어떠한 외란에서도 시스템의 구조를 다이나믹하게 모델링하고자 한다. 이를 위해서, 다층 신경 회로망(mutilayer neural network)과 회귀 신경 회로망(recurrent neural network)의 개념을 도입하고자 한다[3]. 그리하여 기존의 GMDH 알고리즘에서 입력 신호들이 다르게 들어왔을 때 다시 학습시켜야 하는 단점들을 보완하였다. 그리고 이 DPNN은 일반적인 신경 회로망이 가진 가장 큰 장점 중의 하나인 구조적, 산술적 단순함보다도 더욱 단순하다. 결국에는, 비선형 시스템 설계시, 매우 훌륭한 응답 특성을 갖는 시스템 인식 및 제어를 보다 쉽게 이룰 수 있다.

Ⅱ. 비선형 시스템의 모델링

1. 비선형 시스템의 설정

일반적으로, 차분 방정식(difference eqation)에 의한 비선형 다이나믹 시스템의 상태 방정식은 다음과 같다.

$$\begin{aligned} x(k+1) &= \Phi[x(k), u(k)] \\ y(k) &= \Psi[x(k)] \end{aligned} \tag{1}$$

이때, $x(k)$, $y(k)$, 그리고 $u(k)$는 각각 상태, 출력, 그리고 입력을, $\Phi[\cdot]$와 $\Psi[\cdot]$는 비선형 함수를 나타낸다. 이와 같은 비선형 차분 방정식으로 표현된 비선형 이산시간 시스템을 여러가지로 표현할 수 있다. 본 논문에서는 다음 식과 같은 형태의 n차 비선형 시스템을 설정하였다[1].

$$y(k+1) = f[y(k), y(k-1), \cdots, y(k-n+1); u(k), \cdots, u(k-m+1)] \tag{2}$$

이때 $f[\cdot]$는 비선형 함수이다.

여러 분야에서 이러한 비선형 시스템에 대한 모델링에 많은 관심을 보이고 있으며, 또한 신경 회로망의 출현으로 인하여 비선형 시스템 모델링에 대한 관심이 더욱 고조되고 있다. 특히 다층 퍼셉트론을 기초로한 신경 회로망 알고리즘이 이러한 비선형 시스템 모델링을 목적으로 사용되고 있다. 따라서 본 논문에서는 전방향 다층 신경 회로망인 GMDH를 소개하고, 이를 응용하여 파라미터를 다이나믹하게 결정할 수 있는 DPNN을 구현하고자 하는 것이다.

2. GMDH의 도입[4]

GMDH는 선택된 다항식 함수에 의해 일련의 데이타들을 모델링하는 신경 회로망이다. 이 신경 회로망은 입력 노드(input node)들의 집합으로 이루어져 있

으며, 이 입력 노드들은 단 하나의 출력을 얻기 위하여 중간층(hidden processing layer)들을 거치게 된다. 각 층에서의 노드들은 바로 전단계 층의 노드들의 출력들로부터 두개의 입력값만을 취하여 노드의 출력을 만들어낸다. 이때 발생되는 모든 층의 각각의 노드들의 출력은 다음 식과 같이 2차 방정식(quadratic equation)의 형태를 취하게 된다.

$$z_{i,j}=a_j z_{(i-1),r}^2+b_j z_{(i-1),s}^2+c_j z_{(i-1),r}+d_j z_{(i-1),s}+e_j z_{(i-1),r}\,z_{(i-1),s}+f_j \quad (3)$$

이때, a_j, b_j, c_j, d_j, e_j, f_j는 노드의 연결강도(connection weight)이고, i와 j는 각각 층과 노드의 인덱스(index)이다. $z_{i,j}$는 i번째 층의 j번째 노드의 출력이다.

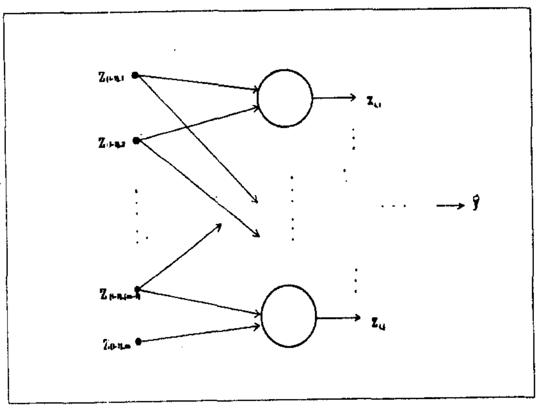

그림 1. 스태틱 PNN의 구성도

다시 말해서, GMDH의 원리는 $(i+1)$층으로 들어오는 m_i개의 입력들의 조합의 수($C_2^{m_i}$)에 의해서 다음 층의 노드 수가 결정되고 난 후, 이 노드들의 출력이 식(3)에 의해 계산되며, 이로인한 출력의 적합성 여부를 어떠한 기준에 의하여 판별한다는 것이다. 이처럼 각각의 층에서의 적합치 못한 노드들을 판별해내기 위해서는, 먼저 원하는 최종 출력(desired final output)과 실제 노드 출력들 사이의 평균제곱 에러(mean squared error)들을 계산하는 것이다. 이 평균제곱 에러의 크기를 구하여 큰 에러를 가진 노드는 없애고 에러가 작은 노드들만을 가지고 앞서 설명한 과정들을 원하는 출력을 얻을 때까지 반복하게 되는 것이다. 그리고 이 알고리즘은 일반적으로 사용되고 있는 다층 퍼셉트론 구조(예를들어, 역전파)와는 달리 중간층의 수와 노드들의 수가 미리 정해지는 것이 아니다. 모든 중간층의 노드의 수는 다음 식과 같다.

$$m_{i+1}=\frac{m_i(m_{i-1})}{2} \quad (4)$$

이때, m_i는 i번째 층에서 살아남은 노드의 갯수이다. 이제까지 설명한 스태틱(static) PNN의 구성도를 그림 1에 나타내었다.

3. DPNN의 구현

일반적으로 다이나믹 신경 회로망을 구현하는 방법에는 여러가지가 제시되고 있는데, 본 논문에서는 스태틱 PNN을 이용하여 DPNN을 구현하는 방법을 제안하려 한다.

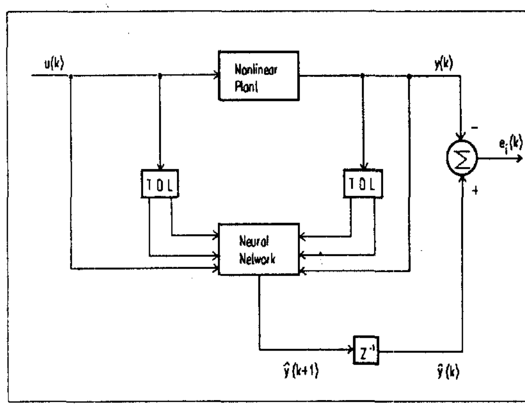

그림 2. 신경 회로망에의한 비선형 시스템의 모델링

앞서 나타낸 비선형 시스템 식 (2)에 대한 모델링의 도식적인 표현을 그림 2에 보였다(이때, TDL은 tapped delay line 이다.). 즉 과거의 입력값과 귀환된 시스템의 출력이 신경 회로망의 입력으로 들어가는 직렬-병렬(series-parallel) 모델을 선택했다. 그러므로 신경 회로망에 대한 표현식은 다음과 같다.

$$\hat{y}(k+1)=N[y(k),\, y(k-1),\cdots,\, y(k-n+1);u(k),\, u(k-1),\cdots,u(k-m+1)] \quad (5)$$

이러한 모델은 모델 내에 귀환 루프가 없기 때문에, 스태틱 PNN의 구조를 사용하여 시스템의 파라미터들을 다이나믹하게 추종할 수가 있다. 이 파라미터들은 다음과 같이 $y(k)$와 $\hat{y}(k)$ 사이의 에러를 최소화 시키는 방향으로 추종된다.

$$\lim_{k \to \infty} |y(k) - \hat{y}(k)| \le \varepsilon_i$$

이러한 DPNN의 구현 방법을 다음과 같이 3단계로 나누어 설명할 수 있다.

[단계 1] 변수벡터의 구성:

식 (3)에 의하여 k 순간에서 i층의 j번째 노드의 출력 $z_{i,j}(k)$를 만들어 내며 다음 식과 같이 간단하게 표현할 수 있다.

$$z_{i,j}(k) = X_j \theta_j \tag{6}$$

이때,

$$X_j = [z_{(i-1),r}(k)^2 \; z_{(i-1),s}(k)^2 \; z_{(i-1),r}(k) \; z_{(i-1),s}(k) \; z_{(i-1),r}(k) \; z_{(i-1),s}(k) \; 1]$$

$$\theta_j = [a_j \; b_j \; c_j \; d_j \; e_j \; f_j]^T$$

여기서 x_j는 차수가 1×6인 데이타 벡터로서 두 스칼라 입력 $z_{(i-1),r}(k)$, $z_{(i-1),s}(k)$를 합성한 것이다. 첫번째 층의 노드의 입력들은 $y(k), y(k-1), \cdots, y(k-n+1), u(k), u(k-1), \cdots, u(k-m+1)$이다. 식 (6)은 θ_j와 함께 2차 방정식을 만들어낸다. 이때 $z_{i,j}(k)$를 얻기 위해서는, 먼저 식(6)을 풀어 계수 벡터(coefficient vector) 즉 연결 강도 θ_j를 얻어야 한다. 각 노드의 출력 $z_{i,j}(k)$가 최종적으로 시스템의 출력 $y(k)$가 되도록 하는 것이 이 알고리즘의 목적이므로 그 일반식은 다음과 같다.

$$\theta_j = (X_j^T X_j)^{-1} X_j^T y(k) \tag{7}$$

[단계 2] 영향이 적은 변수의 제거

앞서 설명했듯이, 단계 1에서 구해진 모든 노드들의 출력들이 다음층으로 전달되는 것은 아니다. 즉 적자 생존의 원칙을 적용해야한다. 일반적인 스태틱 GMDH에서는 출력과 새로이 생성된 변수를 훈련 부분과 검사 부분으로 나누어야 한다. 그래서 출력과 생성된 변수의 훈련부분 과정에서 구해진 평균제곱에러에 의해 필요없는 노드들을 제거한다. 이를 구하는 방법들에는 여러가지가 제시되고 있다[5]. 그러나 본 논문에서는 비선형 시스템의 파라미터를 다이나믹하게 구하고자 하므로 k 순간때 들어온 데이타들에 의하여 그러한 훈련과 검사과정 모두를 수행하였다. 본 논문에서는 GMDH에서 쓰이는 평가함수의 기본적인 형태인 정규화 기준(Regularity criterion)을 다이나믹 시스템에 적합하도록 식(8)과 같이 재정의하였다.

$$r_{i,j}(k)^2 = \frac{[y(k) - z_{i,j}(k)]^2}{[y(k) + \delta(k)]^2} \tag{8}$$

이때

$$\delta(k) = \begin{cases} |y(k-1)| \times \alpha & |y(k-1)| > 0 \\ \delta(k-1) & |y(k-1)| = 0 \end{cases},$$

여기서 $\alpha \ll 1$이다. $\delta(k)$는 어떤 순간 k에서 $y(k)$가 0이 되었을때 $r_{i,j}(k)^2$이 무한대의 값이되는 것을 방지하기 위하여 택한 매우 작은 값이다.

한편 $r_{i,j}(k)$에 의하여 이 값의 증가순으로 i층의 모든 $z_{i,j}(k)$를 다시 정렬시킨다. 그리고 미리 정해진 임의의 기준값 R_{target} 보다 큰 값을 갖는 노드들을 제거한다. 그리고 제거되지 않은 변수 $z_{i,j}(k)$들을 $(i+1)$층의 입력으로 사용한다.

[단계 3]: 최적성의 시험

단계에서는 훈련을 계속할지를 판별하기 위하여, 단계 2에서 계산된 $r_{i,j}(k)$들 중에서 가장 작은 값 $R_{i,j}(k) = \min[r_{i,j}(k)]$를 전 단계의 $R_{(i-1),j}(k)$와 비교하는 것이다. $R_{i,j}(k) < R_{(i-1),j}(k)$이면 단계 1, 2를 반복하고, $R_{i,j}(k) \ge R_{(i-1),j}(k)$이면 $R_{(i-1),j}(k)$에 해당하는 $z_{(i-1),j}(k)$를 최종 출력 $\hat{y}(k)$로 보고 모든 훈련을 끝내게 되는 것이다. 그러면 이 $z_{(i-1),j}(k)$는 Ivakhnenko 다항식이 된다.

Ⅲ. DPNN을 이용한 제어기의 설계

1950년대에 항공기의 자동 조정 장치에 적용하기

위하여 연구하기 시작했던 적응 제어의 개념과 그 이론들은, 1970년대의 혁신적인 컴퓨터 기술의 발달로 인하여 실제 시스템에 적용하기 시작하였다. 이러한 적응 제어는 제어기 설계 방법에 따라, 크게 MRAC (Model Reference Adaptive Control)와 STR(Self-Tuning Regulator)로 분류한다. 특히 기준 입력 신호에 대해서 시스템의 출력이 기준모델의 출력을 추종하도록 제어하는 방식인 MRAC는 간접 MRAC와 직접 MRAC로 분류한다. 본 논문에서는 간접 MRAC를 앞서 구현했던 DPNN을 이용하여 설계하려 한다. 즉, DPNN을 이용하여 시스템을 추정하고 이를 참값이라는 가정하에 DPNN을 이용하여 실제 시스템의 출력이 원하는 출력인 기준모델의 출력을 추종하도록 제어기의 파라미터를 조정하는 것이다.

다시 말해서, 이러한 간접 적응 제어기 설계는 다음과 같은 과정에 의해 이루어진다. 제어기의 구조를 결정하고 시스템의 출력 $y(k)$와 원하는 출력인 기준모델 $y_m(k)$ 사이의 에러를 최소화하기 위하여 시스템을 추정한 후, 제어기의 파라미터를 조정하는 것이다. 즉, 플랜트의 입출력이 $\{u(k),\ y(k)\}$이고 안정한 기준모델의 입출력이 $\{r(k),\ y_m(k)\}$ 일때, 다음과 같은 식을 만족하도록 제어 입력 $u(k)$를 결정하는 것이다.

$$\lim_{k\to\infty} |y(k)-y_m(k)| \le \varepsilon_c \tag{9}$$

그리고 본 논문에서의 제어기 설계에 있어서 비선형 시스템을 식 (10)과 같이 설정하였다. 식 (10)에 대한 간접 기준모델 적응 제어는 그림 3과 같다.

$$y(k+1)=f[y(k),\cdots,y(k-n+1)]+\sum_{i=0}^{m-1}\beta_i u(k-i) \tag{10}$$

이러한 비선형 시스템에 대한 DPNN 모델링의 결과는 식 (11)과 같이 표현되며, 이때 $y(k)$, $y(k-1),\cdots$, $y(k-n+1)$은 DPNN에 들어가는 입력이다.

$$\hat{y}(k+1)=N[y(k),\cdots,y(k-n+1)]+\sum_{i=0}^{m-1}\hat{\beta}_i u(k-i) \tag{11}$$

여기서 $\hat{\beta}_i$은 적응 알고리즘(adaptive algorithm)에 의해 추정한 β_i의 추정치이다. DPNN 모델링에 의해 식 (10)과 식 (11)이 같아지면, 식 (9)를 만족시키는 제어기의 입력을 다음과 같이 표현할 수 있다.

$$u(k)=-N[y(k),\cdots,y(k-n+1)]+y_m(k+1)+r(k) \tag{12}$$

이때 $r(k)$는 기준 입력이고 $y_m(k+1)$은 $y(k),\cdots$, $y(k-n+1)$들로 나타낼 수 있다.

그러나 지금까지 설명한 제어기 설계에 있어서 만족스러운 결과를 얻기 위해서는 시스템의 특성에 관한 몇가지 가정이 필요하다. 예를들면, 모든 경우에 있어서 시스템은 인가된 입력에 대해 한정된 출력을 가져야 한다. 또한 모든 시스템은 최소 위상(minimum phase)을 가져야 한다.

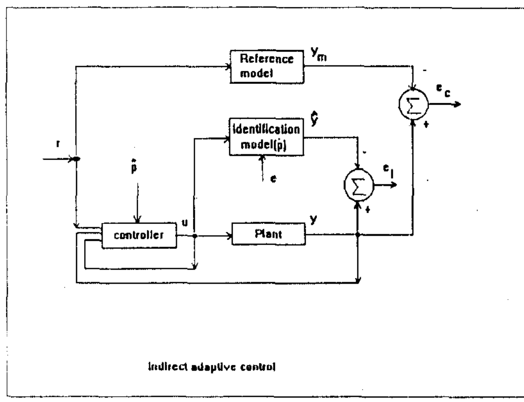

그림 3. 신경 회로망을 이용한 간접 기준모델 적응 제어기.

Ⅳ. 시뮬레이션 결과

(예제 1) DPNN을 이용한 비선형 시스템의 모델링 시뮬레이션 결과

앞서 설명했던 DPNN 알고리즘을 시뮬레이션으로 입증하기 위하여 다음과 같은 비선형 시스템을 설정했다.

$$y(k+1)=$$

$$\frac{y(k)\,y(k-1)y(k-2)\,u(k-1)(y(k-2)-1)+u(k)}{1+y(k-2)^2+y(k-1)^2}+d(k)$$

이때 $d(k)$는 외란이고, 입력 $u(k)=\sin(2\pi k/250)$이며, $R_{target}=10^{-3}$이다. 그리고 $\alpha=10^{-3}$, $\delta(0)=R_{target}^{-1}$으로

설정하였다.

그림 4의 시뮬레이션 결과에서 알수 있듯이, 본 논문에서 제안된 DPNN은 외란을 제외하고는 에러 없이 모델링을 훌륭하게 수행한다.

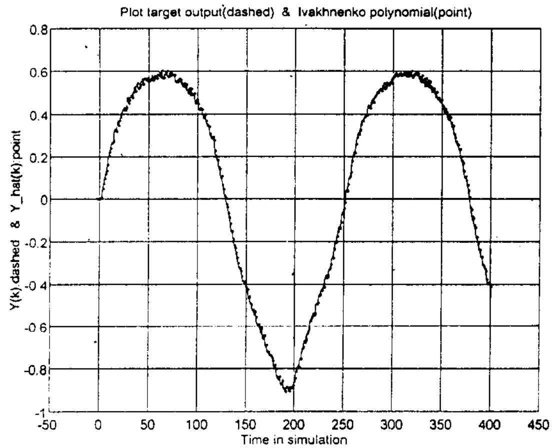

그림 4. DPNN의 모델링 결과

(예제 2) DPNN을 이용한 간접 기준모델 제어기의 시뮬레이션 결과

본 논문에서 구현한 DPNN을 이용한 간접 기준모델 제어기의 시뮬레이션을 위하여 다음과 같은 3차 차분 방정식으로 시스템을 정의하였다.

$$y(k+1)=f[y(k),\ y(k-1),\ y(k-2)]$$
$$+\beta_0\, u(k)+\beta_1\, u(k-1)+d(k)$$

이때 비선형 함수 $f[\cdot]$, 기준모델, 기준 입력, 그리고 두 파라미터 β_0, β_1을 각각 다음과 같이 정의하였다.

$$f[y(k),\ y(k-1),\ y(k-2)]=$$

$$\frac{5\,y(k)\,y(k-1)}{1+y(k)^2+y(k-1)^2+y(k-2)^2}$$

$$y_m(k+1)=0.32\,y_m(k)+0.64\,y_m(k-1)-0.5\,y_m(k-2)+r(k)$$
$$r(k)=\sin(2\pi k/25)$$
$$\beta_0=1,\ \beta_1=0.8$$

그러면 제어기 입력은 다음과 같이 구해진다.

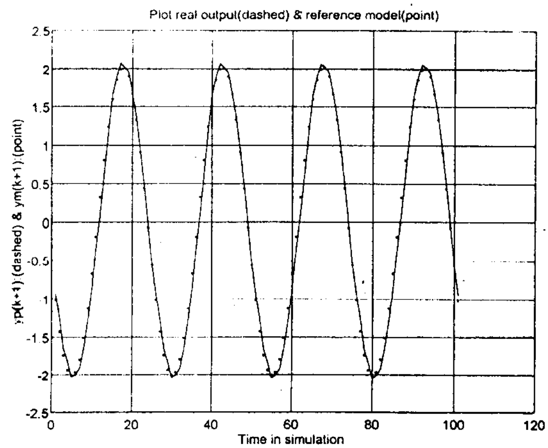

그림 5. 간접 기준모델 제어기의 시뮬레이션 결과

$$u(k)=1/\hat{\beta}_0\ [-N[\,y(k),\ y(k-1),\ y(k-2)]$$
$$-\hat{\beta}_1\, u(k-1)+0.32\,y(k)+0.64\,y(k-1)-0.5\,y(k-2)+r(k)$$

이때, $\hat{\beta}_0$과 $\hat{\beta}_1$은 적응 알고리즘에 의해 추정해야할 파라미터들이다.

이상의 DPNN을 이용한 간접 기준모델 제어의 시뮬레이션 결과를 그림 5에 나타내었다. 시스템의 출력 $y(k)$가 기준 모델의 출력 $y_m(k)$를 에러없이 추종하고 있다는 것을 알 수 있다.

V. 결 론

PNN에서 사용되고 있는 GMDH 알고리즘을 이용하여 비선형 시스템에 대해 알고 있는 정보가 아주 적은 경우에 단지 입/출력값들과 시스템의 차수만으로 비선형 시스템의 모델링을 보다 쉽게 구현하고자 했다.

이를 위해서 GMDH라고 하는 스태틱 알고리즘(static algorithm)을 재구성하여 시스템을 다이나믹하게 추정하는 DPNN을 구현하여 시뮬레이션한 결과, 매우 훌륭한 응답 특성을 갖는 것으로 확인되었다. 또한 이러한 DPNN을 이용한 간접 기준모델 제어기의 시뮬레이션에서는 시스템의 출력이 기준모델의 출력을 매우 잘 추종하는 것으로 확인되었다.

그러나 본 논문에서는 시스템을 모델링하고 제어

하기 위하여 몇 가지 가정이 필요하였고 안정성이 이 가정으로부터 보장되므로, 앞으로 해결해야 할 과제로 남긴다.

참 고 문 헌

1. A. G. Ivakhnenko, "The group method of data handling-A rival of stochastic approximation," *Soviet Automatic control*, pp. 43-55, 1968.
2. A. G. Ivakhnenko, "Polynomial Theory of Complex Systems," *IEEE Trans. Systems, Man & Cyber*., vol. SMC-12, pp. 368-378, Oct. 1971.
3. K. S. Narendra and K. Parthasarathy, "Identification and Control of Dynamical Systems Using Neural Networks," *IEEE Trans. Neural Networks*, vol. 1, no. 1, pp. 329-351, Mar. 1990.
4. S. J. Farlow, *Self-Organizing Methods in Modelling :GMDH type algorithms*. New York : Marcel Decker Inc., 1984.
5. Madan M. Gupta and Dandina H. Rao, *Neuron-Control Systems : The Theory And Application*. IEEE Press, 1993.
6. D. C. Montgomery and E. A. Peck, *Introduction to Linear Regression Analysis* Wiley Inter. Science, 1992.
7. 최종헌, 홍연찬, "DPNN을 이용한 비선형 시스템의 모델링에 관한 연구," '95대 한전자공학회 신호 및 시스템 연구회, 전력전자연구회 추계 합동 학술대회, 1995. 10. 21.
8. K. J. Astrom, B. Wittenmark, *Adaptive Control*, Addison Wesley, 1989.

홍 연 찬(Yeon Chan Hong) 정회원
1979년~1983년 : 서울대학교 전자공학과 졸업(공학사)
1983년~1985년 : 서울대학교 대학원 전자공학과 졸업 (공학석사)
1985년~1989년 : 서울대학교 대학원 전자공학과 졸업 (공학박사)
1985년~1987년 : 엘지전기(주) 연구소 연구원
1987년1990년 : 엘지정보통신(주) 연구소 선임연구원
1990년~1992년 : 순천향대학교 전자공학과 전임강사
1994년~1995년 : Georgia Institute of Technology 전기공학과 Post-Doctor
1992년~현재 : 인천대학교 전자공학과 부교수
※주관심분야 : GMDH 알고리즘, 적응제어, 로보틱스, 신경망 제어, 강인제어

최 종 헌(Jong Hun Choi) 정회원
1994년~1996년 : 인천대학교 대학원 전자공학과 졸업 (공학석사)
1996년~현재 : 엘지정밀(주) 연구소 연구원
※주관심분야 : GMDH 알고리즘, 신경망 제어, ITS