

論文94-31B-10-16

자체반복구조를 갖는 다층신경망에 관한 연구

(A Study on a Recurrent Multilayer Feedforward Neural Network)

李志弘

(Ji Hong Lee)

要約

동적인 시스템을 내부적으로 동적인 특성이 있는 리커런트 신경망을 학습시켜 구현, 또는 근사적으로 흉내낼 수 있음을 보인다. 또한 보간형 신경망과 대표형 신경망의 장점을 모두 갖는 리커런트 신경망의 특성을 이용하여 XOR 문제를 다루고 성능을 개선하기위해 구조를 변형한 새로운 형태의 리커런트 신경망을 제안한다. 새로운 신경망은 내부에 고차항이 저절로 발생하게 함에 착안하였으며 이 제안된신경망으로 XOR 문제를 다루어 기존의 신경망보다 적은 학습으로 우수한 결과를 낼 수 있음을 보이고 잡음이 많 이 섞인 숫자 패턴의 인식에도 성공적으로 응용될 수 있음을 보인다.

Abstract

A method of applying a recurrent backpropagation network to identifying or modelling a dynamic system is proposed. After the recurrent backpropagation network having both the characteristics of interpolative network and associative network is applied to XOR problem, a new model of recurrent backpropagation network is proposed and compared with the original recurrent backpropagation network by applying them to XOR problem. Based on the observation that a function can be approximated with polynomials to arbitrary accuracy, the new model is developed so that it may generate higher-order terms in the internal states. Moreover, it is shown that the new network is successfully applied to recognizing noisy patterns of numbers.

1. 서론

*正會員, 忠南大學校 메카트로닉스 工學科
(Chungnam Nat'l Univ., College of Eng.
Dept. of Mechatronics)

※이 논문은 1993년도 학술진흥재단의 신진연구
과제 연구비에 의하여 연구되었음
接受日字 : 1994年 4月 19日

오늘날 널리 응용되고있는 인공신경망을 크게 두가
지로 분류하면 보간형 (Interpolative associative
memory) 과 대표형 (Heteroassociative or Autoas-
sociative memory) 으로 구분된다. 전자의 예로는 역
전달 회로망 (Backpropagation network) 을 들 수 있

고 후자의 예로는 홉필드 회로망(Hopfield network)을 들 수 있는데 전자의 경우는 주어진 입력형태에 대응하는 출력이 주어질때 이러한 대응을 기억하여 기억된 형태와 유사한 입력이 들어오면 그 기억된 입력에 해당하는 출력과 유사한 출력을 내는 동작을 하는 반면 후자의 경우는 어떤 입력이 들어오면 기억된 형태 중 이 입력과 가장 가까운 형태에 해당하는 출력이 나오도록 되어있다. 그러므로 보간형의 경우는 전체 입력공간내에서 드문드문 학습을 하고도 전체를 지배하는 일반률(Generalization)을 갖추게 되는 장점이 있고 대표형의 경우에는 입력형태의 일부가 상하였거나 불확실한 경우에도 기억된 어떤 형태를 재생하는(Reconstruction) 기능을 가질 수가 있게된다.

보간형의 대표격인 오차역전달 신경망(Backpropagation network)은 간단한 구성과 잘 정리된 학습이론 때문에 가히 폭발적이라 할만큼 여러 분야에서 활발히 연구, 응용되고있다. 본 논문에서도 이 오차역전달 신경망에 초점을 맞추어 내용을 기술하고자 한다. 오차역전달 신경망은 입출력 쌍이 가시적으로 주어져있는 경우에는 거의 모든 경우에 응용 가능한데 그 이유는 그 대응이 이 다층신경망에 의해 구현되도록 내부의 연결강도 및 임계치가 계속 학습되기 때문이다. 그런데 입력의 일부가 손상되었거나 입력이 불분명한 경우, 심한 비선형 대응등의 경우에는 안정점을 기억하는 대표형만큼의 성능을 내기가 어렵다. 왜냐하면 보간형이라는 말의 뜻대로 학습되지 않은 입력에 대해서는 주변의 학습된 입력으로부터 형태의 가까움의 정도에 따라 조합하여 출력이 결정되기 때문이다. 또한 입출력 쌍을 제공한, 즉 신경회로망으로 구현하고자하는 그 시스템이 동적인 특성(현재의 출력이 현재의 입력뿐 아니라 과거의 입력이나 시스템 내부의 상태에 영향을 받는 특성)이 있으면 고전적인 오차역전달 신경망으로는 한계에 부딪치게된다. 특히 이러한 문제는 time series나 동적인 제어대상시스템을 다층신경망으로 구현하려하면 바로 발생하게 된다. 왜냐하면 고전적인 다층신경망은 정해진 입력에 정해진 출력만을 대응시키는 정적인 대응만을 구현할 수 있기 때문이다. 이러한 문제를 극복하기 위한 가장 손쉬운 방법은 디지털 시스템인 경우에 입력 또는 히든 또는 출력유닛의 출력 값을 적당한 갯수만큼 지연(Time delay)시키는 방법이 있을 수 있다.^[8,10,11] 그런데 이 방법은 대상 시스템에 대한 비교적 자세한 정보, 예를 들어 지연 차수등을 필요로하고 디지털 이외의 경우에는 적용에 문제를 야기시킬 수 있다.

대표형 신경망의 경우는 초기 입력이 주어지면 시간이 지나 어떤 안정점에 수렴하게 되고 이 안정점이

정보를 갖고있게 되는데 주어진 초기상태에서 안정점에 이르는 다이내믹스를 조절할 수 있는 방법이 없어서 동적시스템의 다이내믹스를 근사하게 구현하는데 이용되지 못하였다.

이러한 문제를 극복하기 위한 하나의 시도로써 보간형과 대표형의 장점을 취하려는 의도로 개발된 리커런트 오차역전달 신경망(Recurrent Backpropagation Neural Network:RBPN, 앞으로 간단히 리커런트 신경망으로 부르기로한다)에 대한 연구는 최근 들어 활기를 띠어 1993년 San Fransisco에서 개최된 IEEE International Conference on Neural Network에서는 독립된 Session을 구성하게 되었다. 이 네트워크의 기본 착상은 다층 신경망에 피이드 백 경로를 두어 안정점을 기억하되 그 안정점을 역전달 학습 법칙에 의해 원하는 출력으로 대응되도록하여 대표형 신경망의 장점을 갖도록하며 연결강도의 학습을 통하여 초기상태에서 변화하는 내부상태의 시간에 따르는 변화를 조절할 수 있도록하여 신경망의 다이내믹스를 원하는대로 조절함이 가능하게 되었다. 아직 시작 단계에 불과하다고 할 수 있지만 BPTT(Backpropagation ThroughTime)을 다룬 여러 연구^[10,18,20]에서 출발하여 리커런트 네트워크 자체의 메카니즘을 정립하려는 시도^[1,7]와 리커런트 신경망의 성능을 개선하려는 노력이 이루어지고 있으며^[2, 17,7,9] 기본틀 안에서의 학습속도 개선^[12,4] 또는 전혀 다른 방법에 의한 학습 속도의 개선 등의 연구^[21,5,14, 15]가 활발히 진행되고 있음을 알 수 있다. 여기서 이상 정리된 연구 결과로부터 응용 가능 분야를 정리하면 첫째로 기존의 Hopfield 네트워크와 Backpropagation 네트워크를 조합한 형태로서 패턴 인식에 적용될 수 있다. 즉, 여러 패턴을 저장할 뿐만 아니라, 그 저장된 형상을 지정한 패턴으로 대응시키도록 할 수가 있겠다. 예를 들어 XOR 문제라든지 일부가 손상된 숫자등의 인식등에 기존의 회로망보다 일관된 방법으로 적용될수 있겠다. 둘째로 함수와 함수의 대응이다.^[19,16] 즉, 직선이 입력되면 포물선이 출력되고 삼각파가 입력되면 코사인 함수가 출력되는등 함수 공간에서 각 원소에 대해 함수 원소로 대응을 구현할 수 있다. 이는 오차 역전달 회로가 하나의 함수를 구현하기 위해 입력 값을 순차적으로 증가시키며 그에 해당하는 출력 값을 대응시키는 방법에 비해 확실히 확장된 능력을 갖고 있는 셈이 된다. 셋째는 함수를 구별해 내고 시간함수의 다음값을 예측하는 일이다.^[3,3] 예를 들어 입력으로 가능한 함수가 포물선, 삼각파, 구형파라면 어느 일정한 시간동안 이들 중 하나의 신호가 연속적으로 입력될때 어느 입력이 들어왔

었는지를 최종 시각에 구별해내는 종류의 응용이다.

이러한 리커런트 신경망에 대한 연구는 아직 초기단계이나 본 논문에서는 제어문제에 흔히 발생하는 시스템 파악(system identification)부분을 이 리커런트 신경망을 이용하여 해결하는데 대상 시스템이 동적인 경우에도 제대로 모델링하는 방법을 소개한다. 그리고 이 리커런트 신경망을 비선형대응 문제에 적용하였을 때 성능을 향상시키기 위하여 구조를 변형한 새로운 형태의 리커런트 신경망을 제안하고 그 특성 및 응용예를 기술한다. 앞으로 2장에서는 동적시스템에의 응용에 초점을 맞추어 리커런트 신경망의 신호 전달에 관련된 수식을 간단히 정리하고 동적시스템의 근사에 적용한 예를 소개하며 3장에서는 비선형 대응에의 적용에 초점을 맞추어 리커런트 신경망의 XOR 문제에 적용된 결과와 성능을 개선하기 위해 원래 리커런트 신경망의 변형된 형태(Modified Recurrent Neural Network: MRBPN)를 제시하고 성능을 비교하며, 이 RBPN을 숫자 인식에 적용한 예를 기술한다. 그리고 4장에서는 결론을 맺도록 하겠다.

II. 리커런트 신경망에의한 동적시스템 학습

본 장에서는 리커런트 신경망의 신호전달 방법과 학습방법에 대해 기술하고 이를 2차 동적시스템에 적용한 예를 기술한다.

1. 리커런트 신경망

리커런트 네트워크의 제안자는 Pineda^[13]로 볼 수 있으나 이산시간 형태로 신호전달 및 학습 법칙을 기술한 Williams^[22]의 신경망을^[6]의 기호에 따라 정리해 보겠다. 이 신경망의 구조는 그림 1과 같은데 $x_i, i = 1, 2, \dots, n$ 은 입력이며 $y_j, j = 1, 2, \dots, m$ 은 외부로 나타나는 출력이며 그 외의 y_j 는 내부 유닛으로 밖으로 드러나지않는 신호들이다.

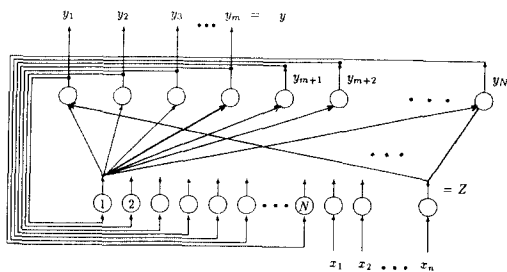


그림 1. 리커런트 신경망
Fig. 1. Architecture of the recurrent network.

설명을 위해 다음의 양을 정의한다. N ; 두번째 층의 유닛의 갯수(y)n ; 입력 변수의 갯수(x)L ; 첫 번째 층의 유닛의 갯수(z)

그러면 순방향 신호전달은 다음의 식으로 기술된다. 먼저 첫째 층의 각 유닛의 출력은 bias항을 포함하여

$$z_j(t) = \begin{cases} 1 & j = 0, \\ x_j(t) & j = 1, \dots, n \\ y_{(j-n)}(t) & j = (n+1), \dots, L \end{cases} \quad (1)$$

가 되는데, 여기서 $z_0(t)$ 는 bias 항이고 둘째 층의 각 유닛의 출력은 다음식으로 기술된다.

$$y_i(t) = S_i(I_i(t)), \quad i = 1, 2, \dots, N \quad (2)$$

$$I_j(t) = \sum_{i=0}^L w_{ij}(t)z_i(t-1), \quad j = 0, 1, \dots, L \quad (3)$$

여기서 $S_i(*)$ 는 각 유닛의 처리함수(processing function)로 보통은 시그모이드 함수나 준선형(semi-linear) 함수가 사용된다. 이때 이 신경망에 의한 오차의 에너지를 다음으로 정의한다.

$$J(t) = \frac{1}{2} \sum_{p \in P(t)} [e_p(t)]^2 = \frac{1}{2} \sum_{p \in P(t)} [y_p^d(t) - y_p(t)]^2, \quad p = 1, \dots, M \quad (4)$$

단, $P(t)$ 는 시각 t에 원하는 출력($y^d(t)$)이 주어져 있는 출력 유닛의 번호들의 집합이다. 그리고 에너지를 연결강도로 미분한 값의 음의 방향으로 변화하게 하 기위해 미분하고 음의 방향으로 연결강도의 증분을 다음으로 구한다.

$$\Delta w_{ij}(t) = -\frac{\partial J(t)}{\partial w_{ij}} = \sum_{p \in P(t)} e_p(t) \frac{\partial y_p(t)}{\partial w_{ij}} \quad (5)$$

여기서 문제가 되는 항을 더욱 자세히 정리하면 다음으로 기술된다.

$$\begin{aligned} \frac{\partial y_p(t)}{\partial w_{ij}} &= S'_i(I_p(t)) \frac{\partial I_p(t)}{\partial w_{ij}} \\ &= S'_i(I_p(t)) \left(\sum_{i=0}^L \frac{\partial}{\partial w_{ij}} [w_{pi}(t)z_i(t-1)] \right) \\ &= S'_i(I_p(t)) \sum_{i=0}^L \frac{\partial}{\partial w_{ij}} (w_{pi}z_i(t-1)) \\ &= S'_i(I_p(t)) \left(\sum_{i=0}^L \left[\delta_{pi} \delta_{ij} z_i(t-1) + w_{pi} \frac{\partial z_i(t-1)}{\partial w_{ij}} \right] \right) \\ &= S'_i(I_p(t)) \left(\delta_{pi} z_i(t-1) + \sum_{i=0}^N \left[w_{p(i+n)} + \frac{\partial y_i(t-1)}{\partial w_{ij}} \right] \right) \end{aligned} \quad (6)$$

여기서 δ_{ij} 는 Kronecker delta 함수이다. ($\alpha = \beta$ 이면 $\delta_{\alpha\beta} = 1$ 아니면 $\delta_{\alpha\beta} = 0$) 이상의 수식안의 반복 부분을 정리하여 리커시브(recursive) 꼴로 표현하면

$$r_{pi}(t) \equiv \frac{\partial y_p(t)}{\partial w_{ij}} = S'_p(I_p(t)) \left(\delta_{pj} z_j(t-1) + \sum_{m=0}^N w_{p(m+n)} r_{pi}(t-1) \right) \quad (7)$$

가 되고 이식의 초기조건은

$$r_{pi}(0) = 0 \quad (8)$$

으로한다. 이렇게 반복꼴로 주어진 식에 의해 매시각으로 학습시키면 (4)의 에너지가 감소하는 쪽으로 신경망이 수렴한다.

$$w_{ij}(t+1) = w_{ij}(t) + \eta \Delta w_{ij}(t) \quad (9)$$

2. 동적시스템의 학습

앞서 정리된 수식을 이용하여 recurrent 신경망을 일반적인 동적시스템의 근사화에 적용하여 본다. 학습에 사용된 동적 시스템은 다음으로 주어지는 전달 함수를

$$H(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad (10)$$

ζ 는 0.3, ω_n 은 0.3, 샘플링시간 T_s 는 1초로 하여 이산시간화한 다음식을 이용하였다.

$$\begin{bmatrix} x_1(t+1) \\ x_2(t+1) \end{bmatrix} = \begin{bmatrix} 0.7956 & -0.0811 \\ 0.9015 & 0.9579 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0.0811 \\ 0.0421 \end{bmatrix} u(t) \quad (11)$$

$$y(t) = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} \quad (12)$$

여기서 x 는 스테이트이고 u 는 입력, y 는 출력이 다. 이러한 시스템으로부터 입력은 유닛스텝함수로 하고 샘플링타입은 1초로 시스템을 40초동안 가동하여 학습용 데이터를 얻었다. (그림 2의 (d)) 이러한 입출력쌍을 둘째층의 유닛의 갯수를 2개로 하고 (왜냐하면 주어진 대상 시스템의 스테이트의 갯수가 2개이므로 원하는 동적특성을 얻기 위해선 최소한 2개의 유닛이 필요하다) 각 유닛의 처리함수는

$$S_i(x) = x \quad (13)$$

으로하고 학습율 $\eta=0.001$ 로하고 바이어스항은 없이, 그리고 각 연결강도는 -0.1에서+0.1 사이의 임의의 값으로 초기화시켰을때 학습횟수에 따른 동적특성 근사정도가 그림 2에 나타났다. 10000회 학습후의 결과는 학습 대상 시스템과 거의 오차가 없는 수준까지 이르고 있음을 관찰할 수가 있다. 대상 시스템의 찾수를 모르는 경우에는 신경망을 응용하는 다른 경우와 마찬가지로 신경망의 유닛의 갯수를 적절히 늘려가며 학습을 시작하면 된다. 제어입력을 입력을 (-1,+1) 사이의 임의의 값을 매번 사용한 경우에도 비슷한 결과를 나타내어 현재 모의 실험한 상황은 입력 함수의 선정에 학습성능이 크게 영향받지는 않는 것으로 나타났으나 적응제어 기법에서 여러 주파수성분이 다양하게 포함되어 있는 입력이 시스템의 파라미터 추정에 도움이 되는 결과를 생각하면 이 신경망에 의한 동적시스템의 파악에도 적절한 입력(forcing function)을 찾는것도 흥미있는 연구과제라고 할 수 있겠다.

또한, 이 모의 실험에 사용된 시스템은 선형인데 비선형 시스템의 파악에는 앞서의 경우처럼 스테이트 2개에 출력층 유닛 2개로는 문제가 발생할 것으로 이러한 리커런트 신경망의 비선형함수 근사 능력에 대한 연구가 추후 진행되어야 할 것이다.

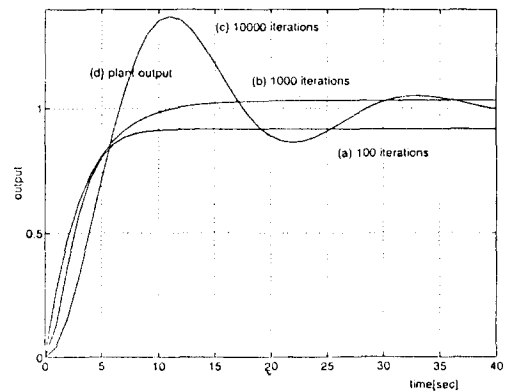


그림 2. 동적시스템의 학습결과

Fig. 2. Output of the recurrent network to simulate a dynamic system.

Ⅲ. 리커런트 신경망의 비선형대응에의 적용

여기서는 XOR 문제와 숫자인식 문제를 다루는데 비선형대응의 성능을 증가시키기 위한 변형된 형태 (MRBPN)를 제안하고 XOR 문제에 대한 결과 비

고, 그리고 MRBPN에 의한 숫자인식 결과를 기술한다.

1. 변형된 리커런트 신경망:MRBPN

MRBPN의 기본개념은 복잡한 함수를 다항식으로 근사화시킬때 고차항이 많으면 많을수록 근사의 정밀도가 증가한다는 정리에 기초하여 신경망의 내부에 고차항이 형성되도록 신호의 흐름을 정한 것이다. 전체적인 형태는 그림 3과 같은데 각 층의 유닛의 갯수를 RBPN의 경우와 같은 기호를 사용했으며 순방향 신호전달과정은 다음과 같다. 단, 그림에서 connection weight 부분은 그림으로 그리기가 복잡해서 간략히 표시했으므로 내부의 구조는 수식을 참조하면 이해가 쉬우리라 생각된다.

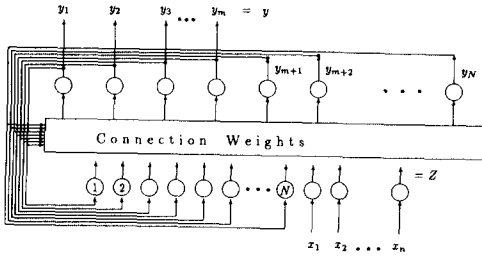


그림 3. 변형된 새로운 리커런트 신경망
Fig. 3. Architecture of the modified recurrent network.

$$z_j(t) = \begin{cases} 1 & j = 0, \text{bias} \\ x_j(t) & j = 1, \dots, n \\ y_{(j-n)}(t) & j = (n+1), \dots, L \end{cases} \quad (14)$$

$$y_i(t) = S_i(I_i(t)), \quad i = 1, 2, \dots, N \quad (15)$$

$$I_i(t) = \sum_{j=0}^L w_{ij}(t)z_j(t), \quad j = 0, 1, \dots, L \quad (16)$$

MRBPN의 핵심 아이디어는 다음 식으로 기술되는데 어떤 시각의 신경망의 연결강도의 값을 기존의 RBPN처럼 상수로 두는 것이 아니라 둘째층의 출력의 함수가 되도록 하는 것이다. 즉,

$$w_{ij}(t) = \sum_{k=0}^N v_{ijk}y_k(t-1), \quad y_0 = 1: \text{bias} \quad (17)$$

그리고 시각 t에서의 에너지를 다음식으로 정의하면

$$J(t) = \frac{1}{2} \sum_{p \in P(t)} [e_p(t)]^2 = \frac{1}{2} \sum_{p \in P(t)} [y_p^d(t) - y_p(t)]^2, \quad p = 1, \dots, M \quad (18)$$

시각 t에서의 보조연결강도 v 의 증분은

$$\Delta v_{ijk} = -\frac{\partial J(t)}{\partial v_{ijk}} = \sum_{p \in P(t)} e_p(t) \frac{\partial y_p(t)}{\partial v_{ijk}} \quad (19)$$

가 되고 이식을 앞서와 같이 리커시브한 꼴로 정리하면

$$\begin{aligned} \frac{\partial y_p(t)}{\partial v_{ijk}} &= S_i'(I_p(t)) \frac{\partial I_p(t)}{\partial v_{ijk}} \\ &= S_i'(I_p(t)) \left(\sum_{l=0}^L \frac{\partial}{\partial v_{ijk}} [w_{pl}(t)z_l(t-1)] \right) \\ &= S_i'(I_p(t)) \sum_{l=0}^L \frac{\partial}{\partial v_{ijk}} \left(\sum_{m=0}^N v_{plm}(t)y_m(t-1)z_l(t-1) \right) \\ &= S_i'(I_p(t)) \frac{\partial}{\partial v_{ijk}} \left(\sum_{l=0}^L \sum_{m=0}^N v_{plm}(t)y_m(t-1)z_l(t-1) \right) \\ &= S_i'(I_p(t)) \left(\delta_{ip}v_k(t-1)z_j(t-1) + \sum_{l=0}^L \sum_{m=0}^N v_{plm} \frac{\partial y_m(t-1)}{\partial v_{ijk}} z_l(t-1) + \sum_{l=0}^L \sum_{m=0}^N v_{p(l+n)m} y_m(t-1) \frac{\partial y_l(t-1)}{\partial v_{ijk}} \right) \end{aligned} \quad (20)$$

로부터

$$\begin{aligned} r_{pjk}(t) &\equiv \frac{\partial y_p(t)}{\partial v_{ijk}} \\ &= S_i'(I_p(t)) \left(\delta_{ip}v_k(t-1)z_j(t-1) + \sum_{l=0}^L \sum_{m=0}^N v_{plm} r_{mjik}(t-1)z_l(t-1) + \sum_{l=0}^L \sum_{m=0}^N v_{p(l+n)m} y_m(t-1)r_{ljk}(t-1) \right) \end{aligned} \quad (21)$$

가 유도된다. 그리고 초기값은

$$r_{ijk}(0) = 0 \quad (22)$$

으로 하고 이 식에 의해 각 보조연결 강도는 다음식에 의해 학습된다.

$$v_{ijk}(t+1) = v_{ijk}(t) + \eta \Delta v_{ijk} \quad (23)$$

2. XOR 문제에서의 적용

XOR 문제는 단층 신경망으로는 해결할 수 없는 대표적인 비선형 문제이므로 이 문제에 RBPN과 MRBPN의 두가지 리커런트 신경망을 적용하여 본다. 두 경우 모두, 둘째층의 유닛의 갯수를 2개 그리고 각 층에 바이어스항을 하나씩, 그리고 학습율 η 는 0.1, 각연결강도는 -0.1에서 +0.1사이의 임의의 값으로잡으므로 초기설정하고 운전시간은 50 단위, 즉 t 를 1에서 50까지 제어입력 u(t)를 일정한 값으로 가한채 학습을 시키는 것을 4가지 종류의 입력((0,0),

(1,0),(0,1),(1,1)을 돌아가면서 100회 수행한 후의 RBPN의 시험결과는 그림 4에 나타내었다. 같은 조건으로 MRBPN을 학습시켰을때는 그림 5의 결과를 얻었다. 그리고 두 경우의 학습횟수에 따르는 오차감소 추이를 그림 6에 비교하였다. 이 그림에서 보면 MRBPN의 경우가 RBPN의 경우에 비해 작은 학습횟수의 학습으로도 주어진 과제를 잘 수행하고 있음을 알 수 있다. MRBPN은 RBPN에 비해 실질적으로 연결수가 증가한다는 단점이 있으나 이로인한 증가 단위가 현재 여러 연구등에 의해 구현되고 하드웨어의 크기에 비해 미미하고 현재 모의실험 단계의 많은 연구가 신경망의 하드웨어적 구현이 상당한 수의 연결을 실현할 수 있을것을 가정하고 이루어지고 있고 또한 현실적으로 발표되고 있어 큰 문제는 되지 않을 것으로 판단된다.

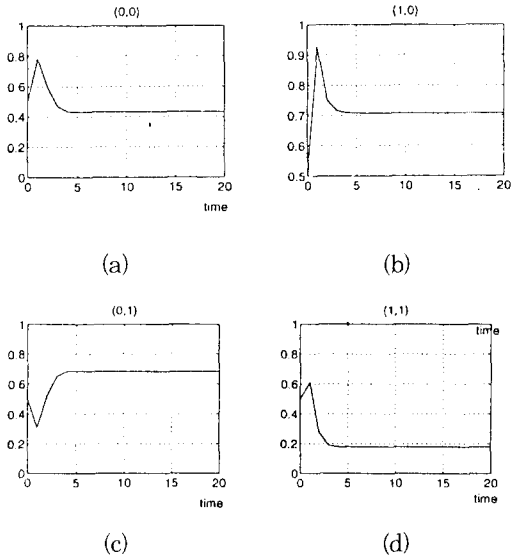


그림 4. 기존 리커런트 신경망으로 XOR를 100번 학습한 결과
 Fig. 4. Output of Williams' recurrent network for XOR after 100 training.

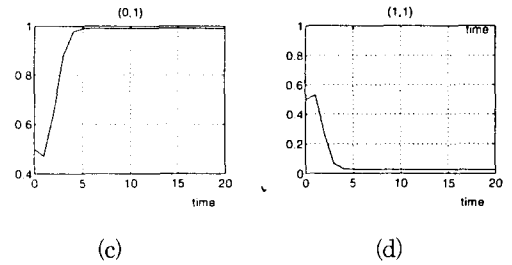
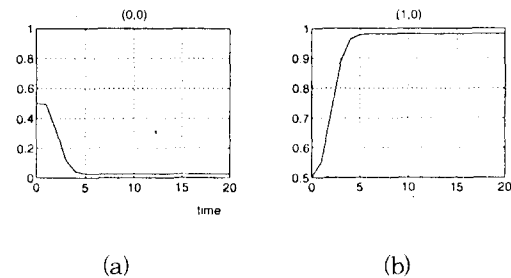


그림 5. 변형된 리커런트 신경망으로 XOR를 100번 학습한 결과
 Fig.5. Output of modified recurrent network for XOR after 100 training.

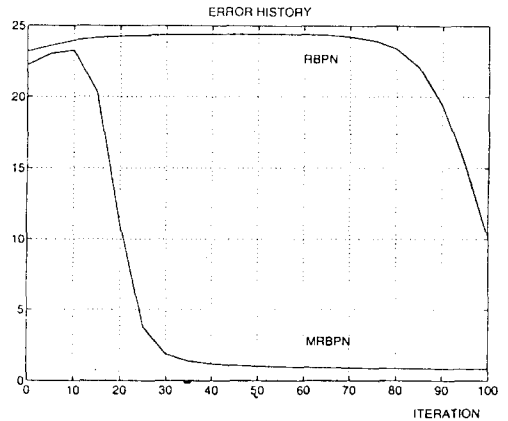


그림 6. 두 리커런트 신경망의 XOR 학습속도 비교
 Fig. 6. Comparison of learning speed of two types of recurrent network for XOR.

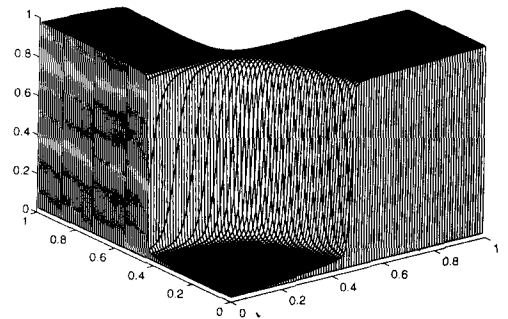


그림 7. 전 입력공간에서의 리커런트 신경망의 XOR 학습결과
 Fig.7. Outputs of modified recurrent network for XOR with contaminated inputs.

앞서 서론에서 언급한바와 같이 리커런트 신경망은 보간형과 대표형의 장점을 가진다. 이를 알아보기 위해 XOR의 입력값을 변화시키며 출력을 구해보면 100번학습 후의 변형된 리커런트 신경망은 그림 7의 결과를 보여준다. 결과에 의하면 입력에 상당한 노이즈가 포함되었어도 강인한 결과를 보이고 있다.

3. 숫자인식에의 적용

변형된 리커런트 신경망의 비선형대응 능력을 보여주는 다른 예로 숫자인식문제를 적용하여 본다.

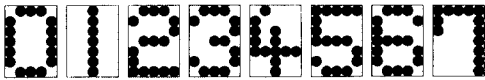


그림 8. 숫자인식을 위한 학습용 숫자
Fig. 8. Patterns for training the network to recognize numbers.

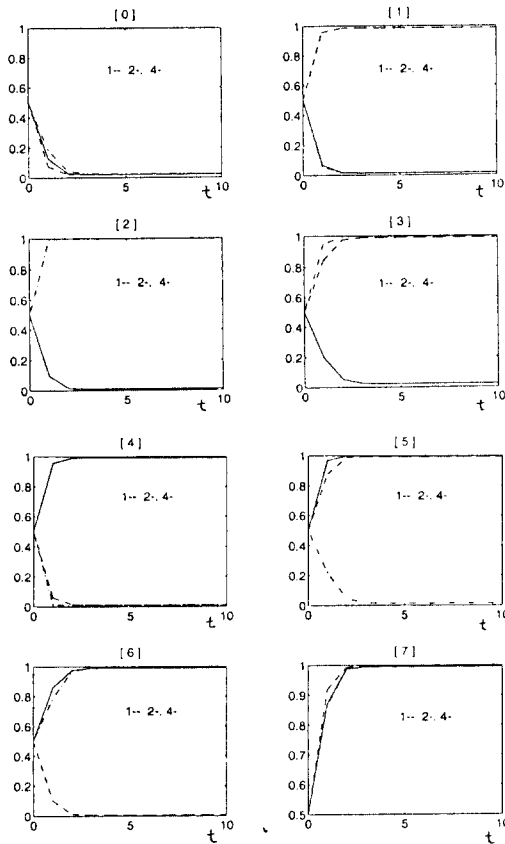


그림 9. 숫자인식과정의 출력
Fig. 9. Outputs during recognizing numbers.

학습에 사용된 숫자는 그림 8의 8개 숫자이고 출력은 3bit로 표시하기 좋게 3개로 정하였다. 그 결과 각 숫자에 대해 50번 학습후에 그림 9의 결과를 얻었다. 초기값으로부터 시간이 지나면 각 bit의 출력이 옳은 값으로 안정되어 간다는 것을 알 수 있다. 또한 약간씩 변형한 여러 숫자에 대해 중간값을 기준으로 인식에 성공한 오염된 숫자의 예는 그림 10에 나타내었다. 이 신경망은 평행이동, 축소, 회전등에 의한 변형은 대처하지 못하고 있으나 인식해야 할 숫자가 어떤 정해진 창(window)안에 들어 있다고 가정된 상태에서는 심하게 변형된 숫자에 대해서도 잘 인식하고 있음을 알 수 있다.

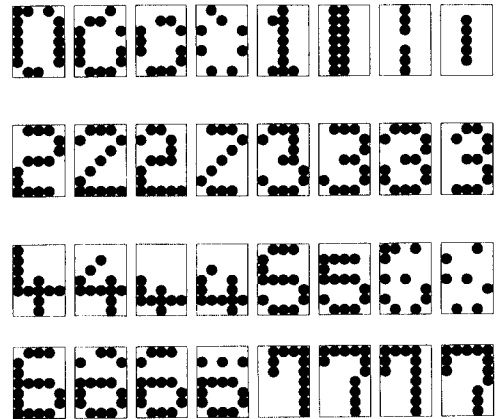


그림 10. 인식에 성공한 숫자의 예
Fig. 10. Some examples of deformed input patterns which the network recognized correctly.

IV. 검토 및 결론

아직 시작 단계에 불과하다고 할 수 있지만 BPTT (Backpropagation ThroughTime) 개념에서 출발하여 최근에 연구되고 있는 리커런트 신경망을 제어 문제에 응용하고 비선형대응등의 문제에 이용할 수 있음을 보였다.

특히 본 논문에서는 리커런트신경망이 신호처리 메카니즘에 동적인 특성을 갖는다는 점과 그 동적특성을 학습에 의해 변형함이 가능하다는 점에 착안하여 제어 대상시스템이 다이내믹스를 갖고 있는 경우에 이 리커런트 신경망을 이용하여 그 시스템을 근사하게 흉내낼 수 있음을 보이고 그 과정을 기술하였다. 이러한 결과는 제어대상 시스템의 모델이 필요한 많

은 경우에 응용될 수 있으리라고 판단된다. 또한 리커런트신경망이 보간형(Interpolative type)과 대표형(Associative type)의 장점을 동시에 가진다는 점에 착안하여 대표적인 비선형 대응 문제인 XOR 문제에 효과적으로 적용이 가능한 변형된 형태의 리커런트 신경망(MRBPN)을 제시하여 기존의 구조와 성능을 비교하여 우수함을 보였다. 이 변형된 구조는 내부 신호 처리과정에 고차항, 즉 신호끼리의 곱해진 꼴이 저절로 형성되도록 하여 비선형 대응에 보다 효과적으로 대처할 수 있도록 고안되었다. 이 MRBPN을 이용하여 숫자인식 문제에서도 만족할만한 결과를 얻었다.

앞으로 이 리커런트 네트워크에 대한 연구는 첫째 네트워크 자체의 저장능력에 대한 해석이 선행되어야 할 것이다. 즉 어느 정도의 갯수의 유닛이면 얼마만큼 복잡한 시간 함수를 구현할 수 있는지, 각 유닛의 처리함수의 형태는 전체 신경망의 성능에 어떤 영향을 미치는지 등의 해석이다. 둘째, 기존의 여러 네트워크의 상관 관계를 분석해 볼 필요가 있다고 생각된다. 이는 기존의 여러 연구 결과를 효과적으로 응용하기위해서 필요하며, 기존의 신경망으로 못하는 어떤 것들을 리커런트 신경망이 해결할 수 있는지 등에 대한 연구이다. 셋째, 기존의 다층 신경망으로 한계에 부딪친 여러 문제에 대한 적용과 아울러, 효율적인 학습 방법에 대한 계속적인 연구가 필요하다고 판단된다. 이는 학습 법칙이 반복적(recursive)인 꼴로 정리되기 때문에 과거의 정보가 필요하다는 점과 이를 이용하기 때문에 기존의 BPN보다 학습에 필요한 계산이 많다는 점 때문이다.

參考文獻

- [1] Yoshua Bengio, Paolo Frasconi, and Patrice Simard. "The Problem of Learning Long-Term Dependencies in Recurrent Networks". In *Proc. of IEEE International Conference on Neural Networks*, pages 1183-1188, 1993.
- [2] D. Chen, C.L. Giles, G.Z. Sun, H.H. Chen, Y.C. Lee, and M. W. Goudreau. "Constructive Learning of Recurrent Neural Networks". In *Proc. of IEEE International Conference on Neural Networks*, pages 1196-1201, Sanfransisco, California, 1993.
- [3] J. P. DeCruyenaere and H. M. Hafez. "A Comparison Between Kalman Filters and Recurrent Neural Networks". In *Proc. of International Joint Conference on Neural Networks*, volume~4, pages 247-251, Baltimore, Maryland, 1992.
- [4] Yan Fang and Terrence~J. Sejnowski. "Faster Learning for Dynamic Recurrent Backpropagation". *Neural Computation*, 2(3):270-273, 1990.
- [5] L.A. Feldkamp, G.V. Puskorius, Jr. L.I. Davis, and F. Yuan. "Neural Control Systems Trained by Dynamic Gradient Methods for Automotive Applications". In *Proc. of International Joint Conference on Neural Networks*, volume~2, pages 798-804, Baltimore, Maryland, 1992.
- [6] Robert Hecht-Nielsen. "Neurocomputing", chapter~6, pages 183-191. Addison-Wesley, 1990.
- [7] Liang Jin, Peter N. Nikiforuk, and Madan M. Gupta. "Dynamics and Stability of Multilayered Recurrent Neural Networks". In *Proc. of IEEE International Conference on Neural Networks*, pages 1135-1140, Sanfransisco, California, 1993.
- [8] John~G. Kuschewski, Stefen Hui, and Stanislaw H. Zak. "Application of Feedforward Neural Network to Dynamical System Identification and Control". *IEEE Transactions on Control Systems Technology*, 1(1):37-49, 1993.
- [9] Lian Li and Simon Haykin. "A Cascade Recurrent Neural Networks for Real-Time Nonlinear Adaptive Filtering". In *Proc. of IEEE International Conference on Neural Networks*, pages 857-862, Sanfransisco, California, 1993.
- [10] K. S. Narendra and K. Parthasarathy. "Identification and Control of Dynamical Systems Using Neural Networks". *IEEE Transactions on Neural Networks*, 1(1):4-27, 1990.
- [11] K. S. Narendra and K. Parthasarathy.

- ". *IEEE Transactions on Neural Networks*, 1(1):4-27, 1990.
- [11] K. S. Narendra and K. Parthasarathy. "Gradient Methods for the Optimization of Dynamical Systems Containing Neural Networks". *IEEE Transactions on Neural Networks*, 2(2):252-262, 1991.
- [12] Barak A. Pearlmutter. "Learning State Space Trajectory in Recurrent Neural Networks". *Neural Computation*, 1(2):263-269, 1989.
- [13] Fernando J. Pineda. "Recurrent Backpropagation and the Dynamical Approach to Adaptive Neural Computation". *Neural Computation*, 1(2):161-172, 1989.
- [14] G.V. Puskorius and L.A. Feldkamp. "Model Reference Adaptive Control with Recurrent Network Trained by the Dynamic DEKF Algorithm". In *Proc. of International Joint Conference on Neural Networks*, volume 2, pages 106-113, Baltimore, Maryland, 1992.
- [15] G.V. Puskorius and L.A. Feldkamp. "Practical Considerations for Kalman Filter Training of Recurrent Neural Networks". In *Proc. of IEEE International Conference on Neural Networks*, pages 1189-1195, Sanfransisco, California, 1993.
- [16] U. Ramacher and B. Schurmann. "Unified Description of Neural Algorithms for Time-Independent Pattern Recognition". In *VLSI Design of Neural Networks*, pages 255-270. Kluwer Academic Publishers, 1991.
- [17] Sathyanarayan S. Rao and Viswanath Ramamurti. "A Hybrid Technique to Enhance the Performance of Recurrent Neural Networks for Time Series Prediction". In *Proc. of IEEE International Conference on Neural Networks*, pages 52-57, Sanfransisco, California, 1993.
- [18] David Rumelhart, James McClelland, and The PDP Research Group. "Parallel Distributed Processing", chapter 8, pages 354-361. The MIT Press, 1987.
- [19] V. Sterzing and B. Schurmann. "Recurrent Neural Networks for Temporal Learning of Time Series". In *Proc. of IEEE International Conference on Neural Networks*, pages 843-850, Sanfransisco, California, 1993.
- [20] Paul J. Werbos. "Backpropagation Through Time: What It Does and How to Do It". *Proceedings of the IEEE*, 78(10):1550-1560, October 1990.
- [21] Ronald J. Williams. "Training Recurrent Networks Using the Extended Kalman Filter". In *Proc. of International Joint Conference on Neural Networks*, volume 4, pages 241-246, Baltimore, Maryland, 1992.
- [22] Ronald J. Williams and David Zipser. "A Learning Algorithm for Continually Running Fully Recurrent Neural Networks". *Neural Computation*, 1(2):270-280, 1989.

 著者紹介



李志弘(正會員)

1983年 2月 서울대학교 전자공학과 졸업(공학사). 1985년 2월 한국과학기술원 전기 및 전자공학과 석사학위 취득. 1991년 2월 한국과학기술원 전기 및 전자공학과 박사학위 취득.

1983년 1월 ~ 1988년 2월 현대중공업 로보트사업부 근무. 1991년 3월 ~ 1991년 6월 한국과학기술원 전기 및 전자공학과 연수연구원. 1991년 6월 ~ 1993년 12월 금오공과대학교 전자제어공학과 조교수. 1994년 3월 ~ 현재 충남대학교 메카트로닉스공학과 조교수. 주관심 분야는 지능제어, 신경회로망제어, 퍼지제어, 다중로보트 경로계획 등임.