

Back-Propagation방법의 수렴속도 및 학습정확도의 개선

Acceleration the Convergence and Improving the Learning Accuracy of the Back-Propagation Method

李 允 燮* · 禹 廣 芳**
(Youn-Seop Lee · Kwang-Bang Woo)

요 약

본 논문은 신경회로망에서 Back-Propagation(BP)방법의 1) 급격한 수렴속도 감소에 대한 해석으로 Sigmoid 활성화함수(Sigmoid logistic activation function)의 미분값이 급격히 감소되는 부분에서 신경회로망이 학습될 경우 수렴속도가 급격히 감소됨을 밝혀내고 2) 해석결과를 기반으로 수렴계수를 정의하여 활성화함수를 수정함에 따른 BP방법의 수렴속도와 학습정확도 개선에 관한 연구이다. 수정된 방법에 의해 binary 출력패턴학습의 적용예로 X-OR문제를 학습시킨 결과, 수렴속도와 학습정확도가 개선되고 연결하중(weight)과 문턱값(threshold)이 특정값으로 수렴하여 신경회로망의 안정성이 향상되었다. 또한, analog출력패턴학습에 대하여 학습의 초기과도현상은 커지나 학습오차는 더욱 작아져서 학습의 정확도를 증가시킨다.

Abstract- In this paper, the convergence and the learning accuracy of the back-propagation (BP) method in neural network are investigated by 1) analyzing the reason for decelerating the convergence of BP method and examining the rapid deceleration of the convergence when the learning is executed on the part of sigmoid activation function with the very small first derivative and 2) proposing the modified logistic activation function by defining the convergence factor based on the analysis. Learning on the output patterns of binary as well as analog forms are tested by the proposed method. In binary output pattern, the test results show that the convergence is accelerated and the learning accuracy is improved, and the weights and thresholds are converged so that the stability of neural network can be enhanced. In analog output pattern, the results show that with extensive initial transient phenomena the learning error is decreased according to the convergence factor, subsequently the learning accuracy is enhanced.

*正 會 員 : 延世大 大學院 電氣工學科 博士課程
**正 會 員 : 延世大 工大 電氣工學科 教授 · 工博
接受日字 : 1990年 2月 10日
1次修正 : 1990年 7月 3日

1. 서 론

신경회로망은 대단위 병렬처리와 더불어 적응학

습능력때문에 신경회로망에 의한 학습 알고리즘을 로보트제어[1, 2, 3], 공정제어기의 구성[4, 5], 음성인식 및 비선형 신호처리[6, 7]등에 응용하는 연구가 진행되고 있다. 학습모델로서 Back-Propagation(BP)방법에 대하여 많은 관심이 집중되고 있다[6]. BP방법은 복잡한 논리나 지식의 학습이 가능케 하나 수학적으로 global optimum에 수렴된다는 보장이 없고, 실제 시스템 적용시 완전한 global optimum을 얻기가 매우 어렵다. 수렴속도면도 어느 정도의 정확도까지는 비교적 빨리 도달하나 그 이후는 급격히 감소하여 학습결과와의 계속된 향상을 기대할 수가 없다. 따라서, 신경회로망 응용에서 어느 정도의 학습결과를 얻기까지 시간 소모가 많고 제어기로 구성되면 학습오차에 대한 제어기의 성능 저하가 초래될 수 있다.

BP방법의 수렴속도 개선에 대한 연구에서, Vogl[7]은 입출력 전체 패턴에 대한 오차의 총화를 계산하여 연결하중을 보정하고 학습율과 모멘텀계수를 학습오차에 따라 동적으로 가변함으로서 수렴속도를 개선하였다. 각 입력, 중간, 출력 단자의 변화범위를 $-1/2$ 과 $1/2$ 사이의 값을 택함으로써 수렴속도를 향상시켰다[8]. 학습율을 반복선형-추적 알고리즘을 이용하는 연구가 추진되고[9], BP방법의 연결하중정정을 위해 학습오차에 대한 2계 미분형태를 이용한 알고리즘이 제시되었다[10].

본 논문에서는 BP방법의 급격한 수렴속도 감소에 대한 수학적 해석이 수행되어 net값이 $\pm\infty$ 로 증감함에 따른 활성화함수 미분값의 급격한 감소와 신경회로망 학습의 수렴속도와와의 관계에 대해 검토한다. 이 해석 결과에 따라 수렴계수의 정의, 활성화함수의 수정을 통해 BP방법의 수렴속도와 학습정확도의 개선이 연구된다.

수정된 방법에 의해 binary출력패턴 학습의 적용으로서 X-OR문제를 학습시키고, 수렴속도와 학습정확도의 개선, net값, 연결하중 및 문턱값의 특정치 수렴과 아울러 안정화된 신경회로망 구축이 제시된다. 미미한 local optimum에 대해서도 global optimum에의 빠른 속도로 보다 정확한 학습 가능성을 보여준다. analog출력패턴 학습의 적용예에서 간단한 플랜트를 학습시켜 수렴속도와 학습정확도에 대하여 검토된다.

2. Back-Propagation 방법

Rumelhart[1]등에 의해 개발된 BP방법은 다

층, 전향 적응 신경회로망에서 반복학습에 의해 연결하중의 결정을 위한 gradient descent방법이다. 학습은 먼저 신경회로망의 초기화를 위해 각 연결하중값을 임의의 작은 값으로 선정하고, 입력패턴의 군과 이에 대응하는 출력패턴의 군을 기반으로 학습규칙에 의해 이루어진다.

입력패턴 P가 신경회로망에 입력되어 전향전달(foward propagation)되면, 각 단자의 활성화(activation)은 식(1)과 같은 미분가능, 비감소 활성화함수(logistic activation function)에 의해 동적으로 결정된다.

$$f(net_{pj}(n+1)) = O_{pj}(n+1) = \frac{1}{1 + \exp(-net_{pj}(n+1))} \quad (1)$$

$$net_{pj}(n+1) = \sum_i W_{ji}(n+1) \cdot O_{pi}(n+1) + \theta_j(n+1) \quad (2)$$

여기서, O_{pi} 는 패턴 p에 의한 단자 i의 활성화값, W_{ji} 는 단자 i와 단자 j간의 연결하중, θ_j 는 단자 j의 문턱값이다. 식(1)에 의한 신경회로망의 실출력과 각 입력패턴에 대응되는 출력패턴과의 오차는 다음 식에 의해 결정된다.

$$출력 단자 : F_{pj}(n) \triangleq \delta_{pj}(n) = f'(net_{pj}(n)) \cdot (t_{pj}(n) - O_{pj}(n)) \quad (3)$$

$$중간 단자 : G_{pj}(n) \triangleq \delta_{pj}(n) = f'(net_{pj}(n)) \sum_k \delta_{pk}(n) \cdot W_{kj}(n) \quad (4)$$

$$f'(net_{pj}(n)) = O_{pj}(n) \cdot (1 - O_{pj}(n)) \quad (5)$$

여기서, t_{pj} 는 출력패턴이다. 식(3)과 (4)에 의해 구해진 오차는 다음 규칙에 의해 각 단자간의 연결하중과 문턱치가 보정되도록 신경회로망을 통하여 후향전달(backward propagation)된다.

$$연결하중의 보정 ; W_{ji}(n+1) = W_{ji}(n) + \Delta W_{ji}(n+1) \quad (6)$$

$$\Delta W_{ji}(n+1) = \eta_w \cdot \delta_{pj}(n) \cdot O_{pi}(n) + \alpha \Delta W_{ji}(n) \quad (7)$$

$$문턱값의 보정 ; \theta_j(n+1) = \theta_j(n) + \Delta \theta_j(n+1) \quad (8)$$

$$\Delta \theta_j(n+1) = \eta_\theta \cdot \delta_{pj}(n) + \alpha \Delta \theta_j(n) \quad (9)$$

여기서, η_w , η_θ 는 학습율로 $0 < \eta < 1$, α 는 모멘텀계수로 $0 < \alpha < 1$ 이다.

2.1 Binary 출력패턴의 학습

Binary 출력패턴 [1, 0]의 학습에서, 수렴조건을 만족하는 연결하중의 초기치를 선정하고 식(1)에서 식(9)까지의 학습규칙을 이용한다.

2.2 Analog 출력패턴의 학습

±Range의 analog출력패턴의 학습을 위해 중간단자의 활성화함수는 식(1)을 이용하며 출력단자의 활성화함수(activation function)는 식(10)과 같이 단순 변환한다.

$$f(net_{pj}(n+1) = O_{pj}(n+1) \pm 2 \cdot Range \cdot \left(\frac{1}{1 + \exp(-net_{pj}(n+1))} - \frac{1}{2} \right) \tag{10}$$

출력단자의 활성화함수를 식(10)과 같이 정의함으로써 신경회로망의 입력력과 각 입력패턴에 대응되는 출력패턴과의 오차가 되는 식(3)은 다음 식과 같이 유도되어 사용된다.

$$\delta_{pj}(n) = f'(net_{pj}(n)) \cdot (t_{pj}(n) - O_{pj}(n)) = 2Range \cdot \left\{ \frac{1}{4} - \left(\frac{O_{pj}(n)}{2Range} \right)^2 \right\} \cdot (t_{pj}(n) - O_{pj}(n)) \tag{11}$$

따라서, Analog출력패턴에 대한 학습은 식(1), (2)와 식(4)에서 식(11)까지의 학습규칙을 이용하여 수행한다.

3. 수렴속도 감소의 수학적 해석

BP방법의 수렴속도에 대한 급격한 감소특성을 binary출력패턴을 이용하여 해석한다. 출력패턴 [0, 1]에 대해, 정확도 높은 학습을 위하여 출력단자의 net_{pj}값이 ±∞로 증가 또는 감소하여야 한다. 이러한 BP방법의 구조적 특성을 감안하여 어느 수준의 정확도 학습 후 net_{pj} 증감의 급격한 둔화와 수렴속도의 급격한 감소특성을 수학적 해석으로 보여준다 즉, net값이 ±∞로 증감함에 따라 활성화함수의 미분값이 급격히 감소하여 이 부분에서 신경회로망 학습이 수행되는 경우 수렴속도가 급격히 감소됨을 보여준다.

식(2)에서 식(9)까지를 이용하여 출력단자의 net_{pj}(n+1)을 전개하면 식(12)와 같다.

$$net_{pj}(n+1) = \sum_i [W_{ji}(n) + \eta_w \cdot F_{pj}(n) \cdot O_{pi}(n) + \alpha \Delta W_{ji}(n)] \cdot O_{pi}(n+1) + \theta_j(n) + \eta_\theta \cdot F_{pj}(n) + \alpha \Delta \theta_j(n) \tag{12}$$

식(12)에서 ΔW_{ji}와 Δθ_j항의 소거를 위해 식(12)

를 반복 전개하고 연결하중과 문턱값에 대한 초기값 W_{ji}(1)=임의의 작은 값, θ_j(1)=임의의 작은 값 또는 0, ΔW_{ji}(1)=Δθ_j(1)=0가 되어 식(12)는 다음 식과 같이 된다.

$$net_{pj}(n+1) = \sum_i W_{ji}(n) \cdot O_{pi}(n+1) + \theta_j(n) + \sum_{k=1}^n \alpha^{n-k} \cdot \eta_w \cdot F_{pj}(k) \cdot \sum_i O_{pi}(k) \cdot O_{pi}(n+1) + \sum_{k=1}^n \alpha^{n-k} \cdot \eta_\theta \cdot F_{pj}(k) \tag{13}$$

여기서, W_{ji}와 θ_j의 일반항을 소거하기 위해 식(12)에서 식(13)까지를 반복적으로 전개하면 다음과 같다.

$$net_{pj}(n+1) = \sum_{L=1}^n \sum_{K=1}^L \alpha^{L-K} \cdot \eta_w \cdot F_{pj}(K) \sum_i O_{pi}(n+1) + \sum_{L=1}^n \sum_{K=1}^L \alpha^{L-K} \cdot \eta_\theta \cdot F_{pj}(K) + \sum_i W_{ji}(1) \cdot O_{pi}(n+1) + \theta_j(1) \tag{14}$$

식(14)의 세제, 네제항은 무한 반복학습에서 일정 범위내의 값이 되고 그 값을 각각 WED, THETA라 한다. F_{pj}(k)의 부호는 출력패턴 값 [0, 1]에 따라 식(3)에 의해 각각 [-, +]이 되고, 0 < O_{pi} < 1이라는 조건에서 net_{pj}(n+1)은 다음과 같다. [부록 참조]

출력 패턴이 1인 경우 ;

$$net_{pj}(n+1) < \sum_{L=1}^n \sum_{K=1}^L \alpha^{L-K} \cdot (\eta_w \cdot N_h + \eta_\theta) \cdot F_{pj}(K) + WED + THETA < \left(\frac{\eta_w N_h}{1-\alpha} + \frac{\eta_\theta}{1-\alpha} \right) \cdot (F_{pj}(1) + F_{pj}(2) + \dots + F_{pj}(n-1)) + (\eta_w \cdot N_h + \eta_\theta) \cdot F_{pj}(n) + WED + THETA \tag{15}$$

출력 패턴이 0인 경우 ;

$$net_{pj}(n+1) > \sum_{L=1}^n \sum_{K=1}^L \alpha^{L-K} \cdot (\eta_w \cdot N_h + \eta_\theta) \cdot F_{pj}(K) + WED + THETA > \left(\frac{\eta_w N_h}{1-\alpha} + \frac{\eta_\theta}{1-\alpha} \right) \cdot (F_{pj}(1) + F_{pj}(2) + \dots + F_{pj}(n-1)) + (\eta_w \cdot N_h + \eta_\theta) \cdot F_{pj}(n) + WED + THETA \tag{16}$$

여기서, N_h 는 중간단자의 수로 모든 $O_{pi}=1$ 이 될 때의 $\sum O_{pi}(k) \cdot O_{pi}(n+1)$ 의 값이다. 식(15)와 (16)의 결과식에서 1회 반복후의 net_{pj} 의 증감분은 식(17)과 같다.

$$|\Delta net_{pj}(n+1)| < (\eta_w \cdot N_h + \eta_\theta) \cdot F_{pj}(n) \quad (17)$$

식(17)에서 $F_{pj}(n)$ 의 함수는 식(3)에 의해 각각 식(18), (19)가 된다.

$$\text{출력 패턴이 1인 경우 ; } F_{pj}(n) = O_{pj}(n)(1 - O_{pj}(n))^2 \quad (18)$$

$$\text{출력 패턴이 0인 경우 ; } F_{pj}(n) = -O_{pj}^2(n)(1 - O_{pj}(n)) \quad (19)$$

식(18)과 (19)에서 출력 패턴 [1, 0]에 따라 $F_{pj}(n)$ 의 절대값은 각각 $[O_{pj} > 1/3, O_{pj} < 2/3]$ 이 후 급격히 감소한다. 실제 출력 패턴 1을 학습시키고자 할 경우 어느 정도 이상의 학습 결과를 얻기 위한 해당 출력단자의 net_{pj} 값과 이 결과에 의한 $F_{pj}(n)$ 값은 표 1과 같다.

표 1에서 어느 수준의 정확한 학습결과 후, F_{pj} 의 절대값이 급격히 감소함에 따라 보다 정확한 학습결과를 얻기 위하여, 즉 $ABS(net_{pj})$ 를 큰값으로 학습하고자 하면, 일정 학습 횟수에 대한 net_{pj} 의 증감분이 급격하게 감소하여 반복학습 횟수는 기하급수적으로 증가하고 수렴속도는 급격히 감소한다. 수렴속도의 급격한 감소원인은 F_{pj} 값이 신경회로망 실출력과 출력패턴의 차이와 활성화함수의 미분값과의 곱으로 net 값이 $\pm\infty$ 로 증감함에 따라 활성화함수의 미분값이 급격히 감소하게 되고 이에 따라 F_{pj} 값이 급격히 감소하기 때문이다. 그러므로, BP방법의 수렴특성 개선을 위해서는 활성화함수의 미분값이 급격히 감소되는 부분을 제외하고 상대적으로 큰 부분에서의 신경회로망 학습이 수행되도록 하는 것이다.

표 1 출력 패턴 1에 대한 학습결과 재료의 net_{pj}, F_{pj} 값

Table 1 net_{pj}, F_{pj} values according to the results of learning for output pattern 1

학습결과	net_{pj}	F_{pj}
0.95	2.9444	2.375×10^{-3}
0.96	3.1781	1.536×10^{-3}
0.97	3.4761	8.7×10^{-4}
0.98	3.8919	3.92×10^{-4}
0.99	4.5951	9.9×10^{-5}
0.999	6.9068	9.99×10^{-7}
0.9999	9.2103	9.999×10^{-9}

4. BP방법의 학습특성 개선

4.1 Binary출력패턴의 학습

어느 수준의 정확한 학습 후 수렴속도는 급격히 감소하고 완전한 global optimum에 수렴하기 위한 연결하중의 초기값 선정이 매우 어렵다. 따라서 반복학습함에 따라 수학적으로는 net 값이 $\pm\infty$ 로 증가하여 증가하여 학습 정확도의 개선이 가능하다. 기술적 실제 응용 분야에 적용하는 경우 net 값이 특정값 이상으로 증가치 않고 이에 따라 반복학습횟수가 증가된다해도 global optimum으로의 학습이 힘들고 어느 수준의 학습오차를 갖는 학습결과를 얻게 된다. 이 두 가지 면에서, 반복 학습 효과를 증진시키기 위해 활성화함수의 미분값이 상대적으로 큰 부분만을 활용하여 출력패턴 [1, 0]을 학습할 수 있도록 수렴계수 β_b 를 제안하고 식(1)의 활성화함수를 식(20)과 같이 수정한다.

$$f(net_{pj}(n+1)) = O_{pj}(n+1) = \frac{1 + \beta_b}{1 + \exp(-net_{pj}(n+1))} - \frac{\beta_b}{2} \quad (20)$$

여기서, 수렴계수 β_b 는 $0 < \beta_b$ 가 되는 작은 상수이다. 식(20)과 같이 주어지는 활성화함수에 대해서 식(5)의 미분식은 식(21)과 같이 유도되어 사용된다. 연결하중과 문턱값에 대한 학습규칙은 식(2)-(4), 식(6)-(9), 식(20), (21)을 이용한다.

$$f'(net_{pj}(n)) = \left(O_{pj}(n) + \frac{\beta_b}{2} \right) \left[1 - O_{pj}(n) + \frac{\beta_b}{1 + \beta_b} \left(O_{pj}(n) + \frac{\beta_b}{2} \right) - \frac{\beta_b}{2} \right] \quad (21)$$

4.2 Analog출력패턴의 학습

\pm Range의 analog출력패턴범위에 대한 학습이 활성화함수 미분값이 상대적으로 큰 부분에서 학습될 수 있도록 수렴계수 β_a 를 제안하여 식(10)의 활성화함수를 식(22)와 같이 수정한다.

$$f(net_{pj}(n+1)) = O_{pj}(n+1) = 2 \cdot \beta_a \cdot \text{Range} \cdot \left(\frac{1}{1 + \exp(-net_{pj}(n+1))} - \frac{1}{2} \right) \quad (22)$$

여기서, 수렴계수 β_a 는 $1 < \beta_a$ 인 상수이다. 식(22)와 같이 주어지는 출력단자의 활성화함수에 대하여 식(11)의 미분식은 식(23)과 같이 유도되어 사용

된다. 연결하중과 문턱값에 대한 학습규칙은 식 (1)-(9)와 식(22), (23)을 이용한다.

$$\begin{aligned} \delta_{pj}(n) &= f'(net_{pj}(n)) \cdot (t_{pj}(n) - O_{pj}(n)) \\ &= 2 \cdot \beta_a \cdot \text{Range} \cdot \left\{ \frac{1}{4} - \left(\frac{O_{pj}(n)}{2 \cdot \beta_a \cdot \text{Range}} \right)^2 \right\} \\ &\quad \cdot (t_{pj}(n) - O_{pj}(n)) \end{aligned} \quad (23)$$

4.3 수정된 방법의 효과

Binary출력패턴학습에서, 식(1)의 활성화함수를 이용하면 [0, 1]의 정확한 학습을 위하여 net값이 $[-\infty, \infty]$ 이어야 하나, 식(21)의 활성화함수를 사용하면 학습이 수렴되는 연결하중의 초기값에 대해 net값은 그림 1의 [NET1, NET2]로 수렴한다. net값이 특정값으로 수렴하며 이에 따라 각 연결하중과 문턱값도 특정값으로 수렴하게 되어 식(1)의 활성화함수 이용시 연결하중과 문턱값이 $\pm\infty$ 로 증감하는 경향이 개선되어 신경회로망을 안정화할 수 있다. 수렴속도와 학습정도면에서는 식(1)의 활성화함수 이용시의 문제점이 지적되었으나, 식(18)의 활성화함수 이용시 특정 net값으로 수렴하게 되며 학습속도가 증대되고 보다 정확한 학습결과가 얻게 된다. 따라서 많은 횟수의 반복학습후 어느 수준의 학습오차로서 local optimum에 잔류하는 경우에도 global optimum의 결과를 신속한 수렴속도로 얻게 된다.

$\pm \text{Range}$ 의 analog출력패턴학습에서, 식(10)의 활성화함수를 이용하면 그림 2의 함수 f 의 전 구간에서 학습이 수행되고, 식(22)의 활성화함수를 이용하면 함수 f_m 의 [NET1, NET1]의 부분에서 학습된다. 그림 2에서 analog출력패턴 학습범위에서

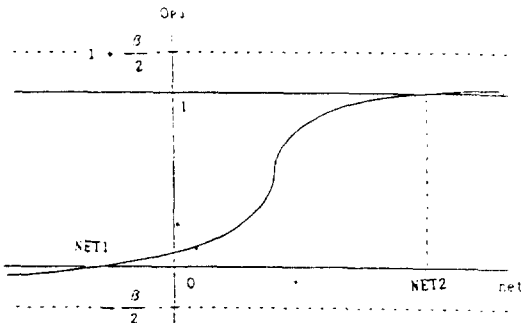


그림 1 Binary출력패턴학습을 위한 수정된 활성화 함수

Fig. 1 Modified logistic activation function for learning the binary output pattern

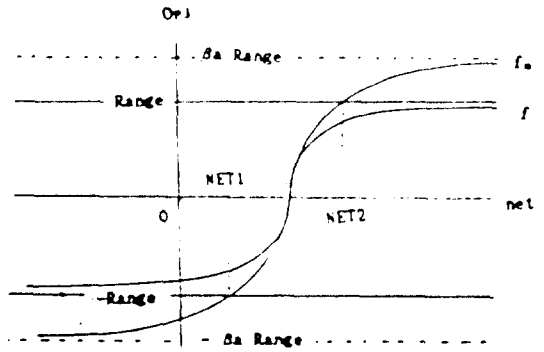


그림 2 Analog출력패턴학습을 위한 수정된 활성화 함수

Fig. 2 Modified logistic activation function for learning the analog output pattern

함수 f_m 의 미분값이 함수 f 의 미분값보다 크기 때문에 수렴속도는 증가한다. 또한, 식(10)의 활성화함수를 사용하면 활성화함수의 미분값이 작은부분에서 신경회로망이 학습될 경우 신경회로망 학습오차에 대하여 상당히 큰 net값의 변화량이 요구되어 학습이 어려워지기도 하나 식(22)의 함수 f_m 을 사용하면 이러한 문제를 없앨 수 있고 학습 정확도를 높일 수 있다.

5. 수정된 방법의 적용

5.1 Binary출력패턴의 학습

수정된 방법에 의한 binary출력패턴의 수렴속도와 학습정확도 검토를 위해 X-OR학습이 진행된다. 이 때 사용된 신경회로망은 입력단자 2, 중간

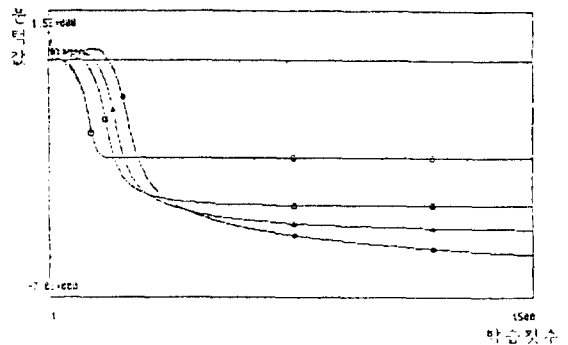


그림 3 출력단자 문턱값의 학습결과

Fig. 3 Learning results of threshold on output unit.

표 2 기존 방법에 의한 연결하중 및 문턱값의 학습결과

Table 2 Learning results of weights and thresholds by original method

단자명, 연결하중 및 문턱값 중간 단자명 문턱값		연결하중			문턱값
		입력단자		출력단자	
		단자1	단자2		
단자1	학습결과	-8.00000	-8.00000	-17.51962	3.71791
	수렴된 학습횟수	62×10^4	63×10^4	140×10^4	275×10^4
단자2	학습결과	-4.27261	-4.30123	6.95565	6.64241
	수렴된 학습횟수	296×10^4	296×10^4	452×10^4	452×10^4
단자3	학습결과	-3.07756	-3.09426	-3.43269	4.73415
	수렴된 학습횟수	175×10^4	296×10^4	140×10^4	452×10^4
단자4	학습결과	-5.16905	-5.20099	9.18373	7.94662
	수렴된 학습횟수	256×10^4	256×10^4	128×10^4	452×10^4
문턱값	학습결과	-	-	-9.96723	-
	수렴된 학습횟수	-	-	452×10^4	-

표 3 기존 방법에 의한 X-OR의 각 입출력 패턴의 학습결과

Table 3 Learning results for each pattern of X-OR problem by original method

입출력 패턴	[1, 1] → [0]		[0, 0] → [0]		[1, 0] → [1]		[0, 1] → [1]	
	net	출력	net	출력	net	출력	net	출력
반복 횟수								
1×10^4	-4.81837	0.00802	-4.98479	0.00679	5.02317	0.99346	5.01681	0.99342
5×10^4	-5.69570	0.00335	-5.81652	0.00297	5.88086	0.99722	5.87692	0.99720
10×10^4	-6.06313	0.00232	-6.16817	0.00209	6.24043	0.99805	6.23856	0.99805
20×10^4	-6.42518	0.00162	-6.51622	0.00148	6.59877	0.99864	6.59839	0.99864
50×10^4	-6.88220	0.00102	-6.97073	0.00094	7.05721	0.99914	7.05322	0.99914
100×10^4	-7.26592	0.00070	-7.33543	0.00065	7.44562	0.99942	7.41356	0.99940
200×10^4	-7.46486	0.00057	-7.42570	0.00060	7.58116	0.99949	7.61590	0.99951
300×10^4	-7.55693	0.00052	-7.49440	0.00056	7.63311	0.99952	7.67402	0.99954
400×10^4	-7.64268	0.00048	-7.50953	0.00055	7.66533	0.99953	7.70541	0.99955
452×10^4	-7.67178	0.00047	-7.50338	0.00055	7.67775	0.99954	7.71776	0.99956
500×10^4	-7.67178	0.00047	-7.50338	0.00055	7.67775	0.99954	7.71776	0.99956

단자 4, 출력단자 1개의 구조이며, 학습율은 0.9, 모멘텀 계수는 0.9를 사용한다. 모든 적용예에서 각 연결하중의 초기값은 동일하며, ± 0.5 범위내의 임의값을 사용하였다.

기존 방법에 의한 각 연결하중 및 문턱값은 표 2와 같이 반복학습된 후 수렴되었으며, X-OR의 각 입출력 패턴에 대한 학습결과는 표 3과 같이 얻었다. 이 경우의 학습 결과에서 452×10^4 번 학

습한 후 각 입출력패턴에 대한 학습오차를 갖고 local optimum에 머문다는 것을 알 수 있다. BP 방법에 대한 학습특성 개선정도를 고찰하기 위해 식(18)에서 β_0 값을 각각 0.0(기존방법), 0.1, 0.2, 0.5의 값을 사용하였으며, 그 결과는 그림 3에서 그림 7, 그리고 표 4에서 보여준다. 이 결과 수정된 BP방법에 의하여 다음과 같은 학습특성을 얻는다.

표 4X-OR 각 입출력 패턴에 대한 학습결과
 Table 4 learning results about each pattern of X-OR problem

입출력 패턴	[0, 0]→[0]				[0, 1]→[1]			
	β 값 학습 횟수	0.0	0.1	0.2	0.5	0.0	0.1	0.2
10	0.51657	0.48723	0.48096	0.54907	0.46323	0.43645	0.43890	0.49272
20	0.49646	0.47346	0.44790	0.22487	0.47925	0.47855	0.49381	0.55527
30	0.42146	0.31269	0.18708	0.11930	0.50780	0.55390	0.60110	0.80009
40	0.22630	0.14693	0.15535	0.00948	0.60372	0.61432	0.64383	0.98768
50	0.15455	0.15966	0.09944	0.00041	0.62287	0.70356	0.86251	1.00040
60	0.16997	0.10578	0.04173	0.00021	0.69617	0.86250	0.04729	1.00028
70	0.13448	0.06045	0.02215	0.00004	0.83046	0.92780	0.97194	1.00002
80	0.09356	0.04117	0.01344	0.00001	0.89180	0.95168	0.98295	1.00001
90	0.07345	0.03080	0.00873	0.00000	0.91797	0.96422	0.98888	1.00000
100	0.06198	0.02423	0.00589	0.00000	0.93206	0.97199	0.99246	1.00000
150	0.03902	0.01012	0.00108	0.00000	0.95892	0.98835	0.99860	1.00000
200	0.03056	0.00527	0.00023	0.00000	0.96837	0.99389	0.99970	1.00000
250	0.02588	0.00301	0.00005	0.00000	0.97349	0.99646	0.99993	1.00000
300	0.02282	0.00180	0.00001	0.00000	0.97680	0.99789	0.99999	1.00000
350	0.02062	0.00111	0.00000	0.00000	0.97915	0.99869	1.00000	1.00000
375	0.01973	0.00088	0.00000	0.00000	0.98010	0.99896	1.00000	1.00000

입출력 패턴	[1, 0]→[1]				[1, 1]→[0]			
	β 값 학습 횟수	0.0	0.1	0.2	0.5	0.0	0.1	0.2
10	0.46532	0.44499	0.45369	0.47996	0.52530	0.52807	0.55829	0.60075
20	0.47947	0.47846	0.49566	0.63521	0.55435	0.57303	0.60367	0.68197
30	0.51123	0.57183	0.65244	0.76791	0.59514	0.65494	0.68298	0.33683
40	0.62864	0.68432	0.74667	0.98588	0.67567	0.63271	0.49336	0.02406
50	0.68589	0.74516	0.86841	1.00099	0.61374	0.44135	0.20043	—
60	0.73199	0.86343	0.94578	1.00037	0.44211	0.19626	0.08368	0.00136
70	0.82478	0.92473	0.97125	1.00007	0.23818	0.10680	0.04694	—
80	0.88810	0.94973	0.98235	1.00001	0.14756	0.07275	0.02990	0.00012
90	0.91535	0.96264	0.98844	1.00000	0.11038	0.05490	0.02016	—
100	0.93009	0.97069	0.99214	1.00000	0.09058	0.04370	0.01403	0.00000
150	0.95803	0.98774	0.99853	1.00000	0.05352	0.01943	0.00276	—
200	0.96779	0.99355	0.99969	1.00000	0.04265	0.01193	0.00081	0.00000
250	0.97306	0.99629	0.99993	1.00000	0.03393	0.00633	0.00013	—
300	0.97645	0.99776	0.99998	1.00000	0.02955	0.00390	0.00003	0.00000
350	0.97886	99861	1.00000	1.00000	0.02645	0.00245	0.00001	—
375	0.97983	0.99890	1.00000	1.00000	—	0.02520	0.00196	0.00000

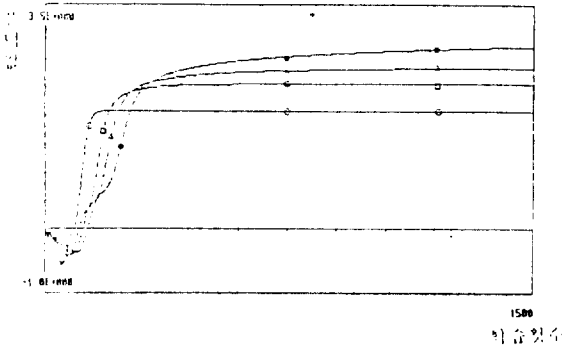


그림 4 중간단자1의 문턱값 학습결과
 Fig. 4 Learning results of threshold on 1st hidden unit.

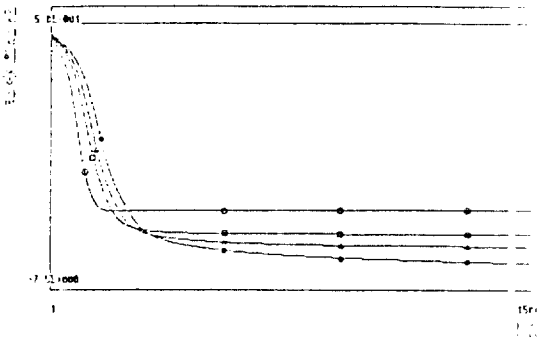


그림 5 입력단자2와 중간단자1간의 연결하중 학습결과
 Fig. 5 Learning results of weights between 2nd input unit and 1st hidden unit.

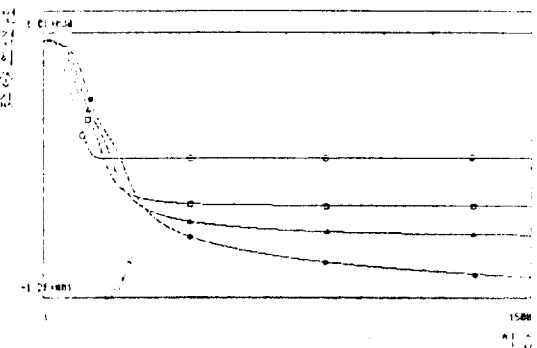
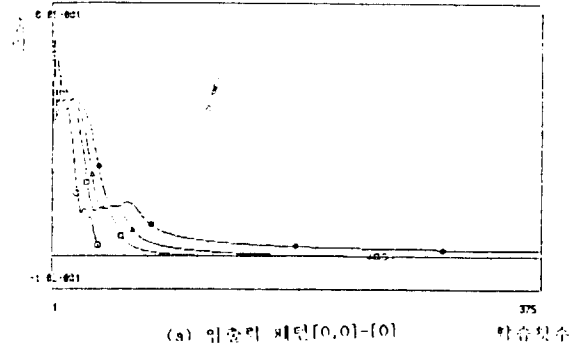
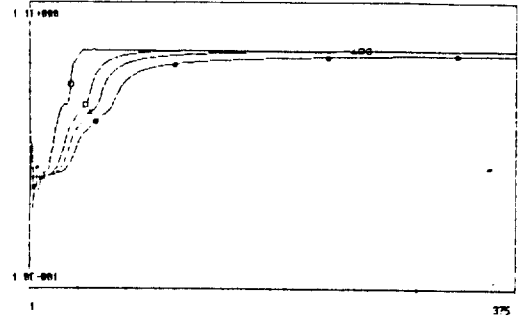


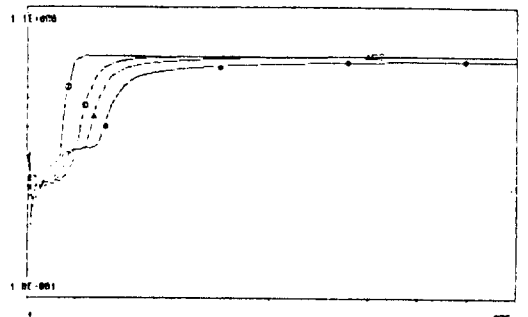
그림 6 중간단자1와 출력단자간의 연결하중 학습결과
 Fig. 6 Learning results of weights between 1st hidden unit and output unit.



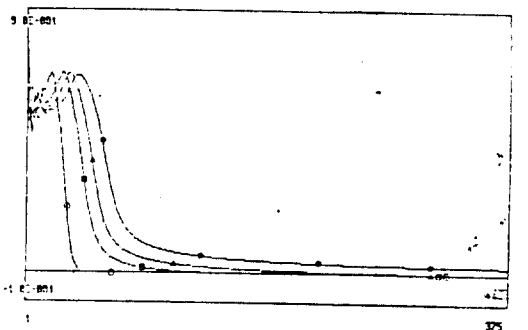
(a) 입출력 패턴[0,0]-[0] 학습횟수



(b) 입출력 패턴[1,0]-[1]



(c) 입출력 패턴[0,1]-[1]



(d) 입출력 패턴[1,1]-[0]

그림 7 XOR각 입출력 패턴에 대한 학습결과
 Fig. 7 Learning results about each pattern of X-OR problem.

1) 입출력에 대한 학습 수렴속도가 기존 방법보다 상당히 개선되었다. 2) 입출력에 대한 학습정도가 개선되었다. 특히 기존 방법에 의해 표 2와 3과 같은 미미한 local optimum을 갖는 학습에 대해서도 빠른속도로 정확하게 학습함을 알 수 있다. 3) 기존 방법에 의한 학습시 연결하중과 문턱값은 매우 느리지만 반복학습함에 따라 $\pm\infty$ 로 증감하는 경향이 있으나 개선된 방법에 의하면 학습의 정확도를 개선하면서도 특정값에 수렴하여 안정화된 신경회로망이 구축될 수 있다. 4) β_0 값이 클수록 수렴속도는 빠르나 overshoot(표 4에서 $\beta_0=0.5$ 인 경우 참조)되는 경향이 있다.

모든 학습결과 그래프에서 다음과 같이 각각 표시하였다. $\beta_0=0.0$ 인 경우는 ●, $\beta_0=0.1$ 인 경우는 △, $\beta_0=0.2$ 인 경우는 □, $\beta_0=0.5$ 인 경우는 ○, 표 2와 3에서 학습결과 자료의 학습횟수는 각 입출력패턴이 학습될 때마다 누산된 숫자이다. 그림 3에서 그림 7까지 및 표 4에서 입출력패턴 학습결과와 반복학습횟수는 패턴별로 누산된 숫자(375회)이고, 연결하중과 문턱값 학습결과와 반복학습횟수는 전 패턴에 대해 누산된 숫자(1500회)이다.

5.2 Analog출력패턴의 학습

수정된 방법에 의한 analog출력패턴의 학습능력 검토를 위해 식(24)의 입력에 대해 식(25)의 플랜트를 그림 8과 같이 학습하여 학습결과 및 학습오차를 구한다. 매 $\pi/180(\text{sec})$ 마다 플랜트 입출력을 측정하여 $0 \leq t \leq 4\pi(\text{sec})$ 동안 720회 신경회로망을 학습하여 결과를 얻는다.

$$\text{입력 ; } U_1(t) = \sin(t), \quad U_2(t) = 2\cos(t) \quad (24)$$

$$\text{플랜트 ; } Z_1(t) = 0.2U_1^2(t) + 0.05U_2^3(t) + 0.1U_1(t) + 0.3U_2(t) + 0.2 \quad (25)$$

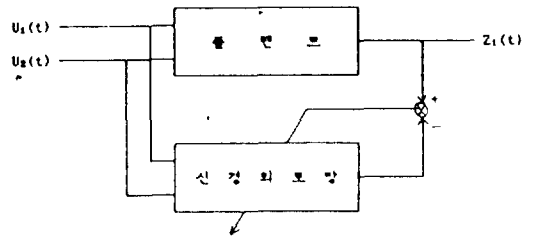


그림 8 신경 회로망의 학습구조
Fig. 8 Learning structure of neural network

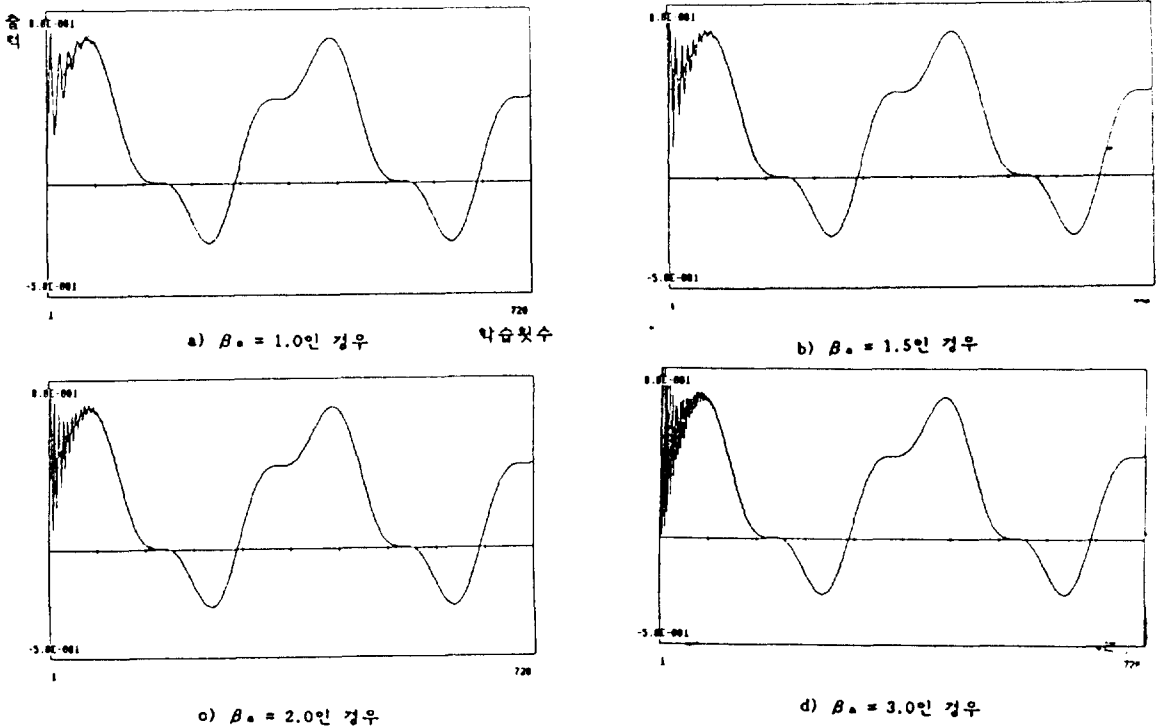


그림 9 플랜트의 출력 패턴과 신경회로망의 출력
Fig. 9 output pattern of plant and actual output of neural network

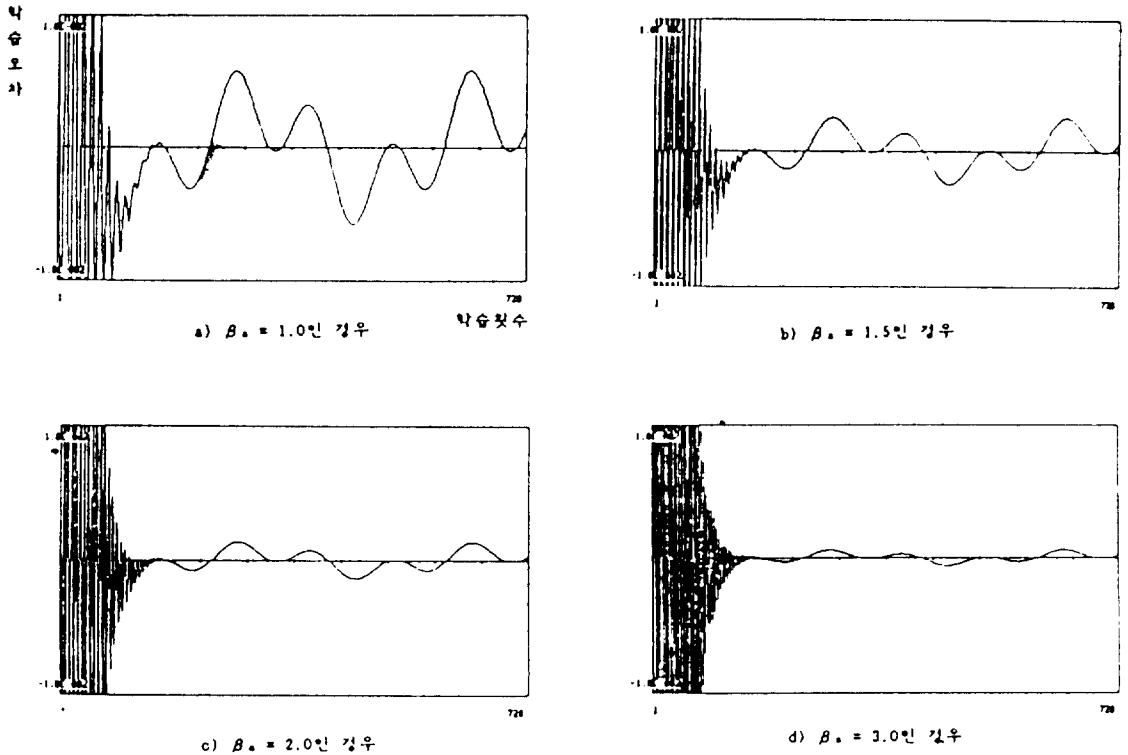


그림 10 플랜트의 각 입출력 패턴에 대한 학습오차
 Fig. 10 Learning error about pattern of plant

이때 사용되는 신경회로망은 입력단자 2, 중간 계층 2, 각 중간계층에는 4개의 중간단자, 출력단자 1개의 구조이며 출력단자에 연결된 연결하중과 문턱값의 학습률은 0.01, 그 외의 연결하중과 문턱값의 학습률은 0.1, 출력단자와 중간단자의 모멘텀 계수는 0.9를 사용한다. 출력단자의 수렴계수 β_a 는 1.0, 1.5, 2.0, 3.0으로 하여 학습결과를 얻으며 동일 조건에서의 결과를 비교검토하기 위해 모든 적용예에서 각 연결하중의 초기값은 동일하며, 초기값은 ± 0.005 범위내의 임의의 값을 사용한다. 수렴계수 β_a 에 따른 학습결과는 그림 9와 같고 학습오차는 그림 10과 같다. 이 결과에서 다음과 같은 학습 특성을 얻었다.

1) 수렴계수 β_a 가 클수록 초기 과도현상이 크다 [그림 9]. 2) 수렴계수 β_a 가 클수록 학습오차는 작아진다[그림 10]. 3) 학습의 수렴속도는 수렴계수 β_a 가 클수록 다소 크나 그 차이는 크지 않다. 그 이유는 수렴속도 감소의 원인이 되는 활성화함수의 미분값이 작은 부분에서의 학습횟수가 전 analog출력패턴학습횟수에 비해 상대적으로 적기 때문이다.

6. 결론

본 논문에서는 BP방법의 급격한 수렴속도 감소에 대한 해석에서 net값이 $\pm\infty$ 로 증감함에 따라 활성화함수의 미분값이 급격히 감소되고 이 부분에서 신경회로망이 학습되는 경우 수렴속도가 급격히 감소된다는 결론을 얻었다. 이런 수학적 해석 결과를 기반으로 수렴계수를 정의하여 활성화함수의 수정을 제안하므로서 BP방법의 수렴속도와 학습정확도를 개선하였다.

제안된 방법에 의한 binary출력학습을 X-OR의 적용예로 학습시킨 결과, 수렴계수 β_a 에 따라 BP방법의 수렴속도와 학습정확도가 개선되어 미미한 local optimum에 대하여도 보다 정확한 학습을 빠른 속도로 수렴하는 것을 알 수 있다. 기존 방법은 정확한 학습결과를 얻기 위해서 반복학습함에 따라 연결하중과 문턱값이 매우 느리게 $\pm\infty$ 로 증가하는 경향이 있으나 수정된 방법은 특정값으로 수렴하여 안정화된 신경회로망 구축이 가능하였다. 또한 analog출력패턴학습을 임의의 플랜트에 대하여 적용시킨 결과, 수렴계수 β_a 에 따라 초기

과도현상은 커지나 학습후의 학습오차는 작아져서 학습의 정확도를 증가시켰다.

수정된 BP방법의 응용범위는 공정 제어기의 구성, 음성인식 및 비선형 신호처리 등이 되고 신속한 학습결과의 취득 및 향상된 제어성능의 시스템 설계가 가능하리라 기대된다.

참 고 문 헌

[1] K.Esley, "A learning Achitecture for control based on back propagation neural network", IEEE 2nd Int. Conf. on neural network II, -587-594, July 24-27 1988
 [2] G. Josin, O. Charney and D. White, "Robot Control using neural network", IEEE 2nd Int. Conf. on neural network II-625-631, July 24-27, 1988
 [3] D. Psaltis, A. Sideris and A. Yamamura, "Neural Controller", IEEE 1st Int. Conf. on neural network, June 21-24 1987
 [4] D. Psaltis, A Sideris and A. Yamamura "A Multilayered neural network controller", IEEE system magazine April 1988
 [5] "DARPA Neural Network Study;Chap. 6" AFCEA International Press 1988
 [6] D. E. Rumelhart, J. c. McClland "Parallel Distributed Processing, Voll, chapter 8",

MIT Press, 1988
 [7] T. P. Vogl, J. K. Margis etc, "Accelerating the convergence of the back-propagation method", Biol. Cybern. 59, 257-263, 1988
 [8] W. Scott Stornetta "An improved three-layer, back-propagation algorithm", IEEE 1st Int. Conf. on neural network II-637-643, June 21-24, 1987
 [9] Edward Denning Dahl, "Accelerated Learning usin te eneralized delta rule", IEEE 1st Int. Conf. on neural network, II-523-530, June 21-24, 1987
 [10] S. E. Fahlman, "Faster-Learning Variation of Back-Propagation:An Empirical Study", Proeedings f he 1988 connectionist odels, 38-51, Carnegi Mellon University
 [11] M. W. Hirsch, "Convergence in Neural Nets", IEE 1st Int. Conf. on neural network. II -115-125, June 21-24, 1987
 [12] B. Widrow, "Learning phenomina in layered neural networks" IEEE 1st Int. Conf. on neural network, II-411-429, June 21-24, 1987
 [13] J. Wang, B. Malakooi, "On trainingof artificial neural networks", IEEE 3rd Int. Conf. on Neural Network, II-387-393, June 18-22, 1989

부 록

$$\begin{aligned}
 NET &= \sum_{L=1}^n \sum_{k=1}^L \alpha^{L-k} \cdot (\eta_w \cdot N_h + \eta_\theta) \cdot F_{pj}(k) \\
 &= (\eta_w N_h + \eta_\theta) \cdot F_{pj}(1) \\
 &\quad + (\alpha \eta_w N_h + \alpha \eta_\theta) \cdot F_{pj}(1) \quad + (\eta_w N_h + \eta_\theta) \cdot F_{pj}(2) \\
 &\quad + (\alpha^2 \eta_w N_h + \alpha^2 \eta_\theta) \cdot F_{pj}(1) + (\alpha \eta_w N_h + \alpha \eta_\theta) \cdot F_{pj}(2) + (\eta_w N_h + \eta_\theta) \cdot F_{pj}(3) \\
 &\quad \vdots \\
 &\quad + (\alpha^{n-1} \eta_w \cdot N_h + \alpha^{n-1} \eta_\theta) \cdot F_{pj}(1) + (\alpha^{n-2} \eta_w + \alpha^{n-2} \eta_\theta) \cdot F_{pj}(2) + \dots + \dots \\
 &\quad + \dots \dots \dots + (\eta_w N_h + \eta_\theta) \cdot F_{pj}(n) \\
 &= \left\{ \frac{1-\alpha^n}{1-\alpha} \cdot \eta_w \cdot N_h + \frac{1-\alpha^n}{1-\alpha} \cdot \eta_\theta \right\} \cdot F_{pj}(1) \\
 &\quad + \left\{ \frac{1-\alpha^{n-1}}{1-\alpha} \cdot \eta_w \cdot N_h + \frac{1-\alpha^{n-1}}{1-\alpha} \cdot \eta_\theta \right\} \cdot F_{pj}(2) \\
 &\quad \dots \dots \dots + (\eta_w \cdot N_h + \eta_\theta) \cdot F_{pj}(n)
 \end{aligned}$$

식(C1)은 $F_{pj}(n)$ 의 부호가 출력패턴 [0, 1]에 따라 식(3)에 의해 각각 [-, +]이므로 출력 패턴이 1인 경우 ;

$$\begin{aligned}
 NET < \left\{ \frac{\eta_w \cdot N_h}{1-\alpha} + \frac{\eta_\theta}{1-\alpha} \right\} \cdot (F_{pj}(1) + F_{pj}(2) + \dots + F_{pj}(n-1)) \\
 + (\eta_w \cdot N_h + \eta_\theta) \cdot F_{pj}(n) \\
 \text{출력패턴이 0인 경우 :} \\
 NET > \left\{ \frac{\eta_w \cdot N_h}{1-\alpha} + \frac{\eta_\theta}{1-\alpha} \right\} \cdot (F_{pj}(1) + F_{pj}(2) + \dots + F_{pj}(n-1)) \\
 + (\eta_w \cdot N_h + \eta_\theta) \cdot F_{pj}(n)
 \end{aligned}$$