

論文2002-39SC-1-3

성장과 소거 알고리즘을 이용한 모듈화된 웨이블릿 신경망의 적응구조 설계

(Adaptive Structure of Modular Wavelet Neural Network Using Growing and Pruning Algorithm)

徐在龍*, 金鏞宅**, 趙鉉讚*, 全洪兌**

(Jae-Yong Seo, Yong-Taek Kim, Hyun-Chan Cho, and Hong-Tae Jeon)

요 약

본 논문에서는 F-투영법과 기하학적인 성장기준을 적용하여 모듈화된 웨이블릿 신경망의 최적구조를 설계할 수 있는 성장과 전지 알고리즘을 제안한다. 기하학적인 성장기준은 지역오차를 고려한 예측 오차기준과 기존의 웨이블릿 함수와의 준직교성을 보장하는 웨이블릿 함수를 배치하기 위한 각도기준으로 구성되어 있다. 이러한 성장기준은 모듈화된 웨이블릿 신경망을 설계자 의도에 부합하도록 구성할 수 있는 방법론을 제시한다. 제안한 성장 알고리즘은 모듈화된 웨이블릿 신경망의 모듈과 망의 크기를 증가시킨다. 또한 소거 알고리즘은 모듈화된 웨이블릿 신경망의 모듈로 사용되는 웨이블릿 신경망의 지역화 특성으로 인해 모듈의 크기가 증가하는 문제점을 극복하기 위해 불필요한 모듈의 노드를 제거한다. 제안한 모듈화된 웨이블릿 신경망의 최적구조 설계알고리즘을 1차원과 2차원의 함수 근사화 문제에 적용하여 제안한 알고리즘의 성능을 검증하였다.

Abstract

In this paper, we propose the growing and pruning algorithm to design the optimal structure of modular wavelet neural network(MWNN) with F-projection and geometric growing criterion. Geometric growing criterion consists of estimated error criterion considering local error and angle criterion which attempts to assign wavelet function that is nearly orthogonal to all other existing wavelet functions. These criteria provide a methodology which a network designer can construct MWNN according to one's intention. The proposed growing algorithm increases in number of module or the size of modules of MWNN. Also, the pruning algorithm eliminates unnecessary node of module or module from constructed MWNN to overcome the problem due to localized characteristic of wavelet neural network which is used to modules of MWNN. We apply the proposed constructing algorithm of the optimal structure of MWNN to approximation problems of 1-D function and 2-D function, and evaluate the effectiveness of the proposed algorithm.

* 正會員, 韓國技術教育大學校 情報技術工學部

(School of Info. Tech. Eng. Korea Univ. of Tech. and Edu.)

** 正會員, 中央大學校 電子電氣工學部

(School of Electrical and Electronics Eng. Chung-Ang Univ.)

※ 본 연구는 과학기술부의 뇌과학연구 프로그램의 지원하에 수행되었습니다.

接受日字:2001年6月21日, 수정완료일:2001年11月23日

I. 서 론

신경망에 대한 모델 선택과 망의 성장 및 소거 알고리즘에 대한 연구가 활발하게 진행되고 있다^[1]. 또한 모듈화된 설계방식의 장점이 인식되면서 Jordan et al. 이 제안한 모듈화된 신경망(Modular Neural Network: MNN)의 성장 및 소거 알고리즘에 대한 연구결과가 보고되고 있다^[2-5].

제안된 성장(growing)과 소거(pruning) 알고리즘들은 엑스퍼트 네트워크(Expert Network; EN)을 추가하거나 제거하는 방식이다. 성장 알고리즘은 초기 하나의 EN을 사용하고 가중된 평균 제곱 오차가 증가할 경우 새로운 EN을 추가한다. 소거 알고리즘은 임의의 크기로 MNN을 구성한 후 게이팅 네트워크(Gating Network; GN)의 지역화 특성을 참조하여 전역오차의 감소에 기여하지 못하는 EN을 제거하는 방식이다. 이러한 방식들은 소거 알고리즘을 적용하기 위해 초기에 구성하는 MNN의 적합한 크기를 결정하는 문제와 다분히 시행착오적인 방법을 사용하였다.

따라서 본 논문에서는 시행착오적인 방법보다는 대상 문제에 적응적인 성장과 소거 알고리즘을 제안한다. 제안한 알고리즘은 기존에 새로운 구조로 제안된 모듈화된 웨이블릿 신경망(Modular Wavelet Neural Network; MWNN)^[6]의 적응구조 설계에 적용하여 직관적인 성장기준을 바탕으로 전역오차가 감소하도록 EN의 팽창과 이동계수를 결정하여 노드를 추가한다. 전역오차가 원하는 문턱치에 도달하여 성장 알고리즘이 종료되면 전역오차의 감소에 기여하지 못하는 각 EN의 노드를 제거하는 소거 알고리즘을 수행하여 최적의 MWNN의 구조를 얻는다.

MWNN의 GN은 비선형성이 강한 MLNN으로 구성하고, 학습 알고리즘은 기울기 학습 방법과 Expectation-Maximization(EM) 알고리즘을 결합하여 사용한다. EM 알고리즘은 수렴 속도가 우수하다는 장점이 있다^{[7][8]}.

II. EM 알고리즘을 이용한 MWNN의 학습 방법

EM 알고리즘은 극대 유사 추정(maximum likelihood estimation)을 위한 반복적인 접근기법 중에 한가지 방법이다. EM 알고리즘의 반복(iteration)은 추정 단계(E-step)와 극대화 단계(M-step)인 두 단계로 구성된다. 극대화 단계는 추정 단계의 각 반복(iteration) 과정에서 재 정의된 유사 함수를 극대화하는 과정이다^{[7][8]}. EM 알고리즘은 먼저 주어진 데이터나 현재 모델에서 완전한 데이터 유사 함수의 기대값을 정의하고, 극대화 단계에서는 새로운 매개변수 θ^{i+1} 을 추정하기 위해 θ 와 관련된 함수를 극대화한다. 이러한 일반적인 EM 알고리즘을 Xu et al.^[9]가 MNN에 적용하여 학습 알고리즘을 개발하였다.

미지의 변수에 대한 기대값으로서 i 번째 반복에서 k 번째 EN이 선택될 기대값 $h_k^{(i)}$ 는 아래의 식과 같이 표현할 수 있다.

$$h_k^{(i)} = E[I_k^i | X, D, \Theta^{(i)}] = P(I_k^i = 1 | \mathbf{x}^i, \mathbf{d}^i) = \frac{g_k(\mathbf{x}^i, \theta_g^{(i)})P(\mathbf{d}^i | \mathbf{x}^i, \theta_k^{(i)})}{\sum_{j=1}^K g_j(\mathbf{x}^i, \theta_g^{(i)})P(\mathbf{d}^i | \mathbf{x}^i, \theta_j^{(i)})} \quad (1)$$

여기서, $P(k | \mathbf{x}^i, \mathbf{y}^i)$ 는 입출력 쌍 $\{\mathbf{x}^i, \mathbf{y}^i\}$ 에 대해 k 번째 EN의 출력에서 생략된 변수를 얻을 수 있는 확률을 나타낸다. 또한 i 번째 반복에서 완전 데이터의 로그유사 함수에 대한 기대값은 다음과 같다.

$$C_F(\theta_k, \theta_g) = \sum_{i=1}^N \sum_{k=1}^K h_k^i \ln g_k(\mathbf{x}^i, \theta_g) + \sum_{i=1}^N \sum_{k=1}^K h_k^i \ln P(\mathbf{d}^i | \mathbf{x}^i, \theta_k) \quad (2)$$

EM 알고리즘에서 고려대상 매개변수는 EN의 분산 Σ_k , EN의 가중치 벡터 \mathbf{w}_k , GN의 가중치 벡터 \mathbf{a}_k 등이 있다.

1. 추정 단계

E단계에서는 완전한 데이터의 로그유사 함수를 구하기 위해 미지의 변수에 대한 기대값 h_k 를 구한다. EN의 출력이 가우스 확률밀도 함수일 때 미지 변수의 기대값은 다음과 같이 다시 표현할 수 있다.

$$h_k^i = \frac{g_k(\mathbf{x}^i, \theta_g) \frac{1}{(2\pi)^{Q/2}} \frac{1}{\Sigma_k} \exp\left(-\frac{(\mathbf{d}^i - \mathbf{y}_k(\mathbf{x}^i, \theta_k))^2}{2\Sigma_k^2}\right)}{\sum_{j=1}^K g_j(\mathbf{x}^i, \theta_g) \frac{1}{(2\pi)^{Q/2}} \frac{1}{\Sigma_j} \exp\left(-\frac{(\mathbf{d}^i - \mathbf{y}_j(\mathbf{x}^i, \theta_j))^2}{2\Sigma_j^2}\right)} \quad (3)$$

여기서 \mathbf{w}_k 는 EN의 가중치 벡터이고, \mathbf{a}_g 는 GN의 가중치 벡터, $\mathbf{y}_k(\mathbf{x}^i, \theta_j)$ 는 k 번째 EN의 출력, \mathbf{d}^i 는 원하는 출력이다.

2. 극대화 단계

EN과 GN이 MLNN이나 RBFN으로 구성될 경우, MLNN의 비선형성 때문에 폐형식(closed form)해가 존재하지 않기 때문에 기울기 학습(gradient learning) 방법을 사용해야 한다^{[7][8]}. 식 (2)를 다시 정리하면 다음과 같다.

$$C_F(\theta_k, \theta_g) = \sum_{i=1}^N \sum_{k=1}^K h_k^i \ln(g_k(\mathbf{x}^i, \theta_g)) - \frac{1}{2} \sum_{i=1}^N \sum_{k=1}^K h_k^i \left[\frac{(\mathbf{d}^i - \mathbf{y}_k(\mathbf{x}^i, \theta_k))^2}{\Sigma_k^2} + \ln \Sigma_k^2 + Q \ln 2\pi \right] \quad (4)$$

극대화 단계에서는 매개변수 값을 조정하기 위해 비용함수를 극대화시키는 단계이다.

(1) EN의 가중치 갱신 규칙

MWNN의 구조에서 t 번째 학습 데이터에 대한 MWNN의 출력은 다음과 같다^[6].

$$y_k^j(\mathbf{x}^t, \mathbf{w}_k^j) = \sum_{m=1}^M w_{km}^j \psi_{k,mn}^j(\mathbf{x}^t) \quad (5)$$

t 번째 학습 데이터가 주어질 때 식 (4)의 비용함수에 대한 k 번째 모듈에서 j 번째 가중치 벡터 $\mathbf{w}_k^j = [w_{k1}^j \ w_{k2}^j \ \dots \ w_{kM}^j]^T$ 의 변화량은 체인룰을 적용하면 다음과 같이 표현할 수 있다.

$$\frac{\partial C_F}{\partial \mathbf{w}_k^j} = \sum_{t=1}^N h_k^t \frac{(d^t - y_k^j(\mathbf{x}^t, \mathbf{w}_k^j))}{\sum_k} \frac{\partial y_k^j(\mathbf{x}^t, \mathbf{w}_k^j)}{\partial \mathbf{w}_k^j} \quad (6)$$

결국 극대화 단계에서 k 번째 모듈인 WNN의 가중치 벡터는 다음 식과 같이 미세조정 값에 의해서 매 반복마다 갱신된다.

$$\mathbf{w}_k^j(i+1) = \mathbf{w}_k^j(i) + \eta \sum_{t=1}^N h_k^t \frac{e_k^j}{\sum_k} \cdot \boldsymbol{\phi}_{k,mn}(\mathbf{x}^t) \quad (7)$$

여기서 i 는 반복을 나타낸다.

(2) GN의 가중치 갱신 규칙

t 번째 학습 데이터가 주어질 때 식 (4)의 비용함수에 대한 GN의 출력층 가중치 a_k 의 변화량은 체인룰을 적용하면 다음과 같은 식을 얻을 수 있다.

$$\frac{\partial C_F}{\partial a_k} = \sum_{t=1}^N \sum_{k=1}^K h_k^t (1 - g_k(\mathbf{x}^t, a_k)) \frac{\partial u_k}{\partial a_k} \quad (8)$$

여기서 $g_k(\mathbf{x}^t, a_k)$ 는 t 번째 학습 데이터에 대한 GN의 k 번째 사전확률을 의미한다. k 번째 출력층 가중치 a_k 는 확률적인 기울기 상승법에 의해 매 반복마다 다음과 같은 식에 의해 갱신된다.

$$a_k(i+1) = a_k(i) + \eta \sum_{t=1}^N \sum_{k=1}^K h_k^t (1 - g_k(\mathbf{x}^t, a_k)) f'(\text{net}_k^t) v_m \quad (9)$$

위와 유사한 방법으로 유사함수에 대한 m 번째 입력층과 k 번째 은닉층의 가중치 a_k^m 의 갱신 규칙은 다음과 같이 정리할 수 있다.

$$a_k^m(i+1) = a_k^m(i) + \eta \sum_{t=1}^N \sum_{k=1}^K \{h_k^t (1 - g_k(\mathbf{x}^t, a_k^m)) f'(\text{net}_k^t) a_v^m\} v_m x_t^i \quad (10)$$

(18)

위의 가중치 갱신규칙은 MWNN의 확률적인 기울기 상승 학습^[6]과 유사한 형태이지만 모든 학습 데이터에 대해서 사후확률과 오차를 고려한다는 것이 상이하다.

III. MWNN의 성장과 소거 알고리즘

1. MWNN의 성장 기준

WNN의 구조에서 은닉층의 노드가 매핑을 수행하고, 은닉층의 전체 개수가 K 이고, $\psi_j \in \mathbf{H}, j=1, \dots, K$ 일 때 이산화된 웨이블릿 변환을 적용하면 WNN의 최종 출력은 다음과 같이 표현된다.

$$f(x) = \sum_m \sum_n c_{m,n} \psi_{m,n}(x) \quad (11)$$

WNN이 K 번째 시간에서 관측(observation) 데이터 O_K 를 이용하여 학습을 수행할 때 출력한 결과를 f_K 로 표현하자. 단계별 최적추정은 다음과 같은 F-투영(F-projection)의 원리로 주어질 수 있다^[10].

$$f_K = \arg \min_{f \in \mathbf{H}} \|f - f_{K-1}\| \text{ s.t. } f_K \in \mathbf{H}_K \quad (12)$$

여기서 \mathbf{H}_K 는 억제조건 $f(t_K) = y_K$ 을 만족하는 \mathbf{H} 공간에서 모든 함수로 구성된 집합이다. 따라서 사후추정은 WNN에 새로운 은닉층 노드를 추가할 경우

$$f_K(t) = \sum_{j=1}^{K-1} c_j \psi_{m,n}^j(t_j) + e_K \psi_{m,n}^K(t_K) \quad (13)$$

이고, 관련 매개변수는 다음과 같다.

$$c_K = e_K, \quad t_K = \frac{nb}{a^m}, \quad a_K = a^{-m} \quad (14)$$

여기서, 예측오차 $e_K = y_K - f_{K-1}(t_K)$ 이고, c_K, t_K, a_K 는 WNN의 파라미터이다. 이러한 성장 알고리즘은 관측 데이터를 연속적으로 얻을 경우, 망의 크기가 무한정 커질 수 있기 때문에 성장 기준을 정하는 것이 필요하다.

망 해의 기하학적인 모양에서 새로운 은닉층 노드를 추가하기 위한 기준은 [6]의 식 (1)과 같고, 기하학적인 기준은 다음과 같이 예측 오차 기준(prediction error criterion)과 각도 기준(angle criterion)으로 단순화해서 생각할 수 있다([6]-식 (2), [6]-식 (3)).

일반적으로 각도 α_k 는 계산하기 어렵기 때문에 각도 기준치를 근사적으로 선택하여 사용한다. WNN의 경우, $j=K-1$ 의 시간에 존재하는 $\{\psi_1, \dots, \psi_{K-1}\}$ 과 ψ_K 의 $\cos(\alpha_k)$ 는 [6]의 식 (4)와 같이 계산할 수 있으며, Mexicanhat 웨이블릿 함수를 고려할 경우 [6]의 식 (6)과 같다. 이 결과에서 만약 각도 $\alpha_k = 90^\circ$ 를 얻을 수 있으면 두 노드사이의 직교성을 보장할 수 있다. 이러한 개념을 바탕으로 다음과 같은 조건을 만들 수 있다.

$$|d| > d_{min}(m) \tag{15}$$

여기서 $|d| = a^{-m}b$ 이고, $d_{min}(m)$ 은 팽창계수 m 이 주어지면 두 웨이블릿 함수의 거리를 나타내는 고정된 값이다.

2. MWNN의 성장 알고리즘

MWNN의 성장 알고리즘은 추가되는 EN(WNN)은 입력공간을 포괄하고 전역 오차를 감소시키게 된다. 또한 기하학적인 성장 기준을 사용하기 때문에 추가되는 EN은 준 직교성을 보장하는 프레임으로 구성되며 최적의 노드 구성이 가능하게 된다.

(1) 초기화 과정

MWNN의 성장 알고리즘은 초기에 하나의 모듈에서 시작한다. 따라서 초기화 과정에서는 EN 모듈의 초기 크기를 결정하고 관련 매개변수를 설정하는 과정이다.

[단계 1] 초기 팽창 계수 결정

입력의 두 노드 의해 완전한 시간영역을 표현하도록 $L(l)^{[6]}$ 을 정의하고 초기 팽창계수를 결정한다.

[단계 2] 노드 사이의 거리 값 초기화

준 직교성을 유지하도록 노드를 추가하기 위한 판단 기준이 되는 두 노드사이의 거리 $d(l)$ 을 계산한다.

[단계 3] 초기 노드의 이동계수 결정

EN을 구성하는 웨이블릿 프레임의 초기 이동 계수의 정수 값을 선택한다.

[단계 4] 초기 가중치 결정

원하는 출력 값의 RMSE를 사용하여 가중치를 초기화한다.

[단계 1] ~ [단계 4]의 과정을 통해 WNN 초기 모듈의 크기와 매개변수를 결정하고 MWNN을 대상 문제에 적용하여 학습을 수행하여 오차가 더 이상 감소하지 않으면 새로운 모듈을 추가하는 적응과정을 수

행한다.

(2) 적응 과정

추가되는 모듈을 구성하는 WNN의 매개변수는 발생하는 지역 오차와 입력시간을 반영하여 결정한다. 전역 오차가 원하는 문턱치에 도달할 때까지 모듈의 성장을 반복한다. 수행 목표치가 원하는 문턱치(ϵ_{MAE})에 도달하면 성장 알고리즘을 종료한다.

[단계 1] 새로운 모듈 추가

- 새롭게 추가되는 모듈의 매개변수 결정

새로운 모듈이 추가될 경우 위의 초기화 [단계 1 - 4]를 이용하여 이동과 팽창계수를 결정할 수 있다.

- GN의 추가된 노드의 가중치 결정

학습 데이터에 대한 각 모듈의 수행정도를 판단할 수 있는 정규화된 지역 MSE의 합(Sum of Normalized Local MSE; SNLMSE)을 다음과 같이 정의한다.

$$E_{L_j} = \frac{\sum_{i=1}^N (d^i - y^i)^2}{\sum_{j=1}^J (d^i - y^i)^2} \tag{16}$$

여기서 y^i 는 i 번째 학습 데이터에 대한 모듈의 출력이고, J 는 모듈의 개수이다. E_{L_j} 는 전체 학습데이터에 대한 j 번째 모듈의 수행정도를 판단하는 기준을 제공한다. $\max(E_{L_j})$ 을 이용하여 GN의 가중치를 $a_j = a^c$ 로 설정한다. c 는 가장 큰 E_{L_j} 을 가진 모듈의 출력과 연결된 GN의 가중치 번호이다.

[단계 2] 새롭게 추가된 모듈 학습

기존 모듈의 매개변수 값을 고정시키고 새롭게 추가되는 모듈만 학습하여 근사 학습을 수행한다.

[단계 3] 전체 모듈 학습

[단계 2]가 완료되면 고정된 기존 모듈과 새롭게 추가된 모듈의 매개변수와 동시에 학습하여 조정한다. 전역 오차가 원하는 문턱치(ϵ_h)에 도달하면 MWNN의 성장 알고리즘을 종료하고, 전역 오차가 ϵ_h 에 도달하지 못하면 [단계 1]을 반복한다.

전체 모듈의 학습 종료단계에서 각 모듈의 과도적응 상태를 판단한다. 기존의 모듈이 $J-1$ 개이고, J 번째 새로운 모듈이 추가되었을 경우 j 번째 모듈(EN)의 가중된 MSE (Weighted MSE; WMSE)를 다음과 같이 정의할 수 있다^[8].

$$E_j = \frac{\sum_{i=1}^N h_j^i (d^i - y^i)}{\sum_{i=1}^N h_j^i}, \quad j=1, \dots, J \quad (17)$$

여기서 N 은 학습 데이터의 개수를 나타내며, h_j^i 는 EM 알고리즘의 추정 단계에서 i 번째 학습 데이터에 대한 j 번째 모듈의 사후확률에 해당된다.

전체 모듈의 학습 종료단계에서 전역 오차가 더 이상 감소하지 않은 상태일 때 E_j 값이 문턱치 ϵ_p 보다 큰 모듈은 과도 적응한 것으로 판단할 수 있다. 따라서 과도적응이 발견될 경우 해당 모듈의 매개변수를 고정시켜 추후 성장 단계에서 학습되지 않도록 한다. 성장 알고리즘이 종료된 후에는 MWNN의 관련 매개변수와 구조를 저장한다. 이렇게 저장된 값들은 소거 알고리즘에 재사용 된다.

3. MWNN의 소거 알고리즘

현재 일부 연구자들이 개발한 MNN과 HMNN의 소거 알고리즘을 함수근사화 문제와 패턴인식에 적용하여 좋은 수행결과를 얻고 있다^{[3][5]}. 그러나 제안한 방식들은 전체 모듈을 제거하는데 소거 알고리즘을 사용하고 초기 MNN의 구성에도 적합한 구조를 찾는 데 시행착오 방법을 사용하였다. 따라서 초기 구성한 MNN의 크기에 따라 최적의 MNN 구조를 찾는 데 실패할 가능성이 존재한다.

따라서 본 논문에서는 수행능력이 부족한 모듈을 제거하는 방식을 사용하지 않고, 각 모듈의 노드를 제거하도록 설계하였다.

제안한 MWNN의 성장 알고리즘은 새로운 모듈을 추가할 때 전체 입력시간을 고려하기 때문에 GN에 의해 부분적으로 입력시간이 분할되어 특정 입력시간에서만 각 GN이 동작한다. 따라서 불필요한 노드를 제거해도 전체 수행능력에는 큰 영향을 미치지 못한다. 이것은 모듈로 사용하는 WNN의 지역화된 특성(웨이블릿 함수의 지역화 특성)에 기인한다. 위와 같은 원리를 바탕으로 소거 알고리즘은 불필요한 모듈의 노드를 제거함으로써 MWNN의 최적 구조를 설계할 수 있는 방법을 제공할 수 있다.

[단계 1] 모듈의 불필요한 노드 제거

각 모듈의 노드로 사용되는 웨이블릿 함수의 이동계수와 팽창계수에 해당되는 입력시간을 고려하여 GN의

출력을 검사하고, GN의 출력이 문턱치(ϵ_p) 이하인 경우 노드를 제거하며 제거한 노드와 연결된 가중치를 0으로 초기화하여 새로운 구조의 모듈을 생성한다. 위의 과정을 다른 모듈에 반복적으로 적용하여 불필요한 노드를 제거함으로써 모든 모듈의 최적구조를 찾는다. 만약 모듈의 전체 노드가 제거된다면 모듈자체를 제거시키고, GN의 해당 출력층 노드를 제거하며 가중치를 0으로 초기화하는 과정이 추가된다.

[단계 2] 학습단계

[단계 1]에서 각 모듈의 불필요한 노드를 제거하여 최적구조로 설계된 MWNN의 재학습을 수행하는 단계이다. 이는 제거된 노드가 지역 오차 감소에 어느 정도 기여하고 있기 때문이다.

IV. 모의실험 및 검토

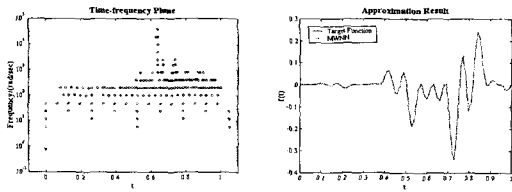
본 장에서 제안한 MWNN의 성장과 소거 알고리즘을 함수근사화 문제에 적용하여 수행능력을 검증하였다. WNN의 기저 함수는 Mexicanhat 웨이블릿 프레임을 사용하였다. MWNN의 초기 모듈의 노드 개수는 $m(l) = L(l)$ 로 설정하였다. 초기 모듈이 하나의 노드에서 시작할 경우, 이 노드는 전체 입력시간을 커버하도록 결정되기 때문에 DC 성분처럼 작용하여 시스템의 안정성을 보장해 줄 수 있다. 그리고 GN은 MLNN으로 구성하였으며 비교적 단순한 함수를 근사화하기 때문에 은닉층 개수는 2~4개로 설정하였다.

1. 모의실험 1

모의실험 1에서의 근사화 대상함수는 식 (18)과 같은 비주기함수이며 시간축 상에서 $[0, 1]$ 을 고려했고, 100개의 학습 데이터를 사용하였다.

$$f(t) = t \sin t \cos 5t \sin 10t \cos 30t \sin 50t \quad (18)$$

그림 1은 MAE를 0.005로 설정하고 성장 알고리즘을 적용한 결과를 보인 것이다. 제시한 근사화 수행결과와 같이 MWNN의 성장 알고리즘이 성공적으로 적용되었음을 확인할 수 있다. 성장과정에서 15개의 모듈이 사용되었다. 그림 2는 MWNN의 성장 알고리즘이 적용되는 성장패턴을 나타낸 것이다. 가로축은 추가되는 노드의 개수를 나타내고, 세로 축은 새로운 노드가 추가되기 전의 RMSE를 나타낸 것이다.



(a) 노드의 시간-주파수 분포 (b) 함수의 근사화 결과
 그림 1. 성장 알고리즘의 적용한 결과 (MAE : 0.005)
 Fig. 1. Applied results of growing algorithm

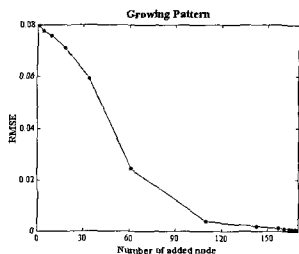
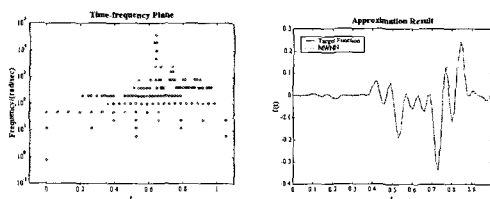


그림 2. MWNN의 성장 패턴(MAE : 0.005)
 Fig. 2. Growing patterns of MWNN

그림 3은 MWNN의 소거 알고리즘을 적용한 결과를 보인 것이다. 모듈의 노드를 제거하는 기준은 $\epsilon_p=0.001$, $\epsilon_v=0.01$ 로 설정하였다. 그림 3에서 확인할 수 있듯이 첫 번째 모듈과 두 번째 모듈은 각각 입력시간의 왼쪽과 오른쪽으로 분할되어 지역화 되었기 때문에 소거 알고리즘을 적용할 경우, 각 모듈이 분할된 해당 영역에서 출력을 내지 못하는 노드가 제거되었다. 나머지 다른 모듈도 유사한 원리로 소거 알고리즘에 의해 잉여노드가 제거되었다. 그러나 잉여노드가 발생하지 않은 모듈도 존재한다(9, 11 - 15 모듈). 또한 특정 모듈의 노드가 삭제된 후에도 수행결과가 크게 악화되지 않았음을 확인할 수 있다. 이것은 모듈의 구성하는 WNN의 지역화 특성에 기인한 것이다.



(a) 노드의 시간-주파수 분포 (b) 함수의 근사화 결과
 그림 3. 소거 알고리즘의 적용 결과(MAE : 0.005)
 Fig. 3. Applied results of pruning algorithm

표 1은 MAE를 0.005와 0.002로 설정하고

MWNN의 성장과 소거 알고리즘을 적용한 수치결과를 나타낸 것이다. 소거 알고리즘을 적용한 후에 전체 노드의 개수가 감소되었음을 확인할 수 있다. RMSE는 성장과 소거 알고리즘이 종료된 후 테스트 데이터를 이용하여 수행한 결과를 나타낸 것이다.

표 1. 성장과 소거 알고리즘
 Table 1. Growing and pruning algorithm.

MAE : 0.005	전체 노드의 개수	RMSE
성장 알고리즘 적용	169	0.008364
소거 알고리즘 적용	126	0.011672
MAE : 0.002	전체 노드의 개수	RMSE
성장 알고리즘 적용	226	0.000368
소거 알고리즘 적용	197	0.000648

2. 모의실험 2

모의실험 2에서는 식 (19)와 같은 주기성을 갖고 있는 근사화 대상 함수를 사용하였다. 시간축상에서 [0, 2]를 고려하고 학습과 테스트 데이터는 201개를 사용하였다.

$$y(t) = \begin{cases} \sin(2\pi \times 2t) + \sin(2\pi \times 4t) & \text{if } t \leq 0.51 \\ \sin(2\pi \times 5t) + \sin(2\pi \times 10t) & \text{if } 0.51 < t \leq 1.51 \\ \sin(2\pi \times 3t) + \sin(2\pi \times 6t) & \text{if } t > 1.51 \end{cases} \quad (19)$$

본 모의실험에서는 MWNN의 모듈화식 설계 방식이 대상함수의 입력시간을 적절하게 분할하여 근사화 수행능력의 향상을 달성할 수 있는지 여부와 제안한 소거 및 성장 알고리즘의 성능평가를 위한 것이다. 관련 매개변수는 각각 $\epsilon_p=0.05$, $\epsilon_v=0.12$ 로 설정하였으며, 입력시간이 좀 더 복잡하기 때문에 GN의 은닉층은 4개를 사용하였다.

표 2. 성장과 소거 알고리즘
 Table 2. Growing and pruning algorithm.

MAE : 0.02	전체 노드의 개수	RMSE
성장 알고리즘 적용	240	0.038132
소거 알고리즘 적용	202	0.039842
MAE : 0.01	전체 노드의 개수	RMSE
성장 알고리즘 적용	296	0.024445
소거 알고리즘 적용	234	0.026842

표 2는 MWNN의 성장과 소거 알고리즘을 적용한

수치적인 결과를 나타낸 것이다. MAE를 0.02로 설정하여 모의실험을 수행한 결과에서 추가되는 모듈의 노드가 MAE 값을 기준으로 입력시간에서 지역 오차가 크게 발생하는 영역에만 추가되어 지역화가 부분적으로 달성되기 때문에 모듈에서 삭제되는 노드의 수가 크게 증가하지 않았다.

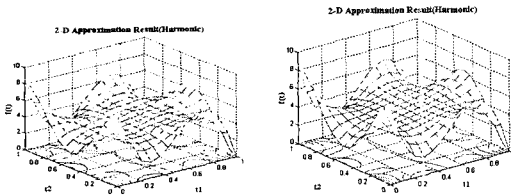
3. 모의실험 3

모의실험 3에서의 근사화 대상함수는 식 (20)과 같은 Harmonic 함수이고, 시간축 상에서 $[0, 1] \times [0, 1]$ 을 고려하였으며, 학습과 테스트 데이터는 225개를 사용하였다.

$$f(t_1, t_2) = 42.659[0.1 + \hat{\epsilon}_1(0.05 + \hat{\epsilon}_1 - 10 \hat{\epsilon}_1 \hat{\epsilon}_2 + 5 \hat{\epsilon}_2^2)] \quad (20)$$

여기서, $\hat{\epsilon}_1 = t_1 - 0.5$ 이고, $\hat{\epsilon}_2 = t_2 - 0.5$ 이고, $\epsilon_p = 0.01$, $\epsilon_v = 0.11$ 로 설정하고, GN은 6개의 은닉층을 사용하였다.

그림 4는 MAE를 0.4로 설정하고 각각 성장과 소거 알고리즘이 종료된 후 테스트 데이터를 이용해서 근사화를 수행한 결과를 나타낸 것이다. 모의실험 결과 14개의 모듈이 사용되었다. 등교선을 살펴보면 근사화가 비교적 양호하게 수행되었음을 확인할 수 있다.



(a) 성장 알고리즘 적용 (b) 소거 알고리즘 적용

그림 4. Harmonic 함수의 근사화 결과(MAE : 0.4)
Fig. 4. Approximation Results of Harmonic function

표 3. 성장과 소거 알고리즘

Table 3. Growing and pruning algorithm.

MAE : 0.4	전체 노드의 개수	RMSE
성장 알고리즘 적용	189	0.038904
소거 알고리즘 적용	139	0.041236
MAE : 0.2	전체 노드의 개수	RMSE
성장 알고리즘 적용	265	0.004401
소거 알고리즘 적용	216	0.006498

표 3에서는 모의실험을 수행한 수치결과를 보였다.

MAE를 0.2와 0.4로 설정한 모의실험 결과를 비교하면 소거 알고리즘에 의해 제거되는 노드의 개수는 유사하다. 이것은 MWNN의 성장 단계에서 모듈의 지역화가 달성되어 지역 오차가 급격하게 감소하였기 때문이다.

V. 결 과

본 논문에서는 MWNN의 성장과 소거 알고리즘을 개발하였다. 제안한 알고리즘의 목표는 MWNN의 초기 매개변수 결정과 대상문제에 적합한 모듈의 개수 및 모듈의 노드 개수를 결정하여 MWNN의 최적구조를 찾는 것이다. MWNN의 성장 알고리즘은 지역오차를 감소시키도록 모듈의 크기를 결정하여 자동으로 모듈을 추가하게 된다. 또한 성장 알고리즘이 종료된 후에 모듈의 잉여노드를 제거하여 MWNN의 최적구조 설계를 완성한다.

제안한 MWNN의 성장과 소거 알고리즘을 함수근사화 문제에 적용하여 최적구조 설계 알고리즘의 우수성과 근사화 능력을 검증하였다. 또한 제안한 알고리즘은 단일차원의 함수근사화 문제에 성공적으로 적용될 수 있음을 검증하였다. 그리고 모의실험 3을 통해 시간-주파수 분석법^[6]을 이용한 최적구조 설계 알고리즘의 문제점인 다차원 확장 문제를 해결할 수 있음을 확인하였다. 이러한 결과를 통해서 모듈화식 설계가 갖는 장점을 적용하고, 적합한 망의 크기를 결정할 수 있는 가능성을 확인하였다.

참 고 문 헌

- [1] R. Reed, "Pruning Algorithm - A Survey," IEEE Trans. on Neural Networks, Vol. 6, pp. 610-622, 1995.
- [2] J. Fristch, M. Finke, and A. Waibel, "Adaptively Growing Hierarchical Mixtures of Experts," *Advances in Neural Information Processing Systems-9*, M. I. Jordan, M. C. Mozer, and T. Petsche, Eds. Cambridge, MA: MIT Press, pp. 459-465, 1997.
- [3] S. R. Waterhouse and A. J. Robinson, "Pruning and Growing Hierarchical Mixtures of Experts," *The 4th Int. Conf. on Artificial Neural Networks*, Cambridge, UK, pp. 341-346, 1995.

- [4] C.C. whitworth and V. Kadirkamanathan, "Cross-Entropy Based Pruning of The Hierarchical Mixtures of Experts," *Proc of the IEEE Workshop Neural Networks for Signal Processing*, pp. 375-383, 1997.
- [5] Viswanath Ramamuri and Joydeep Ghosh, "Structurally Adaptive Modular Networks for Nonstationary Environments," *IEEE Trans. on Neural Networks*, Vol. 10, No. 1, pp.152-160, 1999.
- [6] 서재용, 김용택, 조현찬, 전홍태, "시간-주파수 분석을 이용한 모둘라 웨이블릿 신경망의 최적 구조 설계," *대한전자공학회 논문지*, 제38권, SC편, 제2호, pp. 12-19, 2001.
- [7] Michael I. Jordan and Robert A. Jacobs, "Hierarchical Mixtures of Experts and the EM Algorithm," *Neural Computation*, Vol. 6, No. 1, pp. 181-214, 1994.
- [8] Lei Xu and M. I. Jordan, "On Convergence Properties of the EM Algorithm for Gaussian Mixtures," *Neural Computation*, Vol. 8, pp. 129-151, 1996.
- [9] 서재용, 김용택, 조현찬, 김용민, 전홍태, "F-투영법을 이용한 웨이블릿 신경망의 성장 알고리즘," *대한전자공학회 2001 하계종합학술대회 논문집*, 제24권 제1호, Vol. 3, pp. 165-168, 2001.
- [10] Visakan Kadirkamanathan and Mehasan Niranja, "A Function Estimation Approach to Sequential Learning with Neural Networks," *Neural Computation*, Vol. 5, pp. 954-975, 1999.

 저 자 소 개

徐 在 龍(正會員)

1969年 7月 21日生. 1996년 2월 중앙대학교 전자공학과 졸업(학사). 1998년 2월 중앙대학교 대학원 전자공학과 졸업(석사). 2001년 8월 중앙대학교 대학원 전자공학과 졸업(박사). 2000년 8월~현재 : 한국기술교육대학교 정보기술공학부 BK 객원교수. 주관심분야 : 퍼지, 유전알고리즘, 신경망, FNN, 웨이블릿 신경망 등임

趙 鉉 讚(正會員)

현재 : 한국기술교육대학교 정보기술공학부 교수

金 鏞 宅(正會員)

1967年 2月 23日生. 1992년 2월 중앙대학교 전자공학과 졸업(학사). 1994년 8월 중앙대학교 대학원 전자공학과 졸업(석사). 1996년 9월~현재 : 중앙대학교 대학원 전자공학과 박사과정 재학중. 주관심분야 : 신경망, 로보틱스, 퍼지 등임

全 洪 兌(正會員)

현재 : 중앙대학교 전기전자공학부 교수