

## 다중 역전파 신경회로망을 이용한 비선형 시스템의 모델링

백재혁\* · 이정문\*\*

### Nonlinear System Modeling Based on Multi-Backpropagation Neural Network

Jae-Huyk Baeg\* · Jung-Moon Lee\*\*

---

#### ABSTRACT

In this paper, we propose a new neural architecture. We synthesize the architecture from a combination of structures known as MRCCN (Multi-resolution Radial-basis Competitive and Cooperative Network) and BPN (Backpropagation Network). The proposed neural network is able to improve the learning speed of MRCCN and the mapping capability of BPN. The ability and effectiveness of identifying a nonlinear dynamic system using the proposed architecture will be demonstrated by computer simulation.

---

#### 1. 서 론

복잡한 시스템의 정확한 비선형 모델을 구하는 것은 어려우며, 모델을 수학적으로 구하는 것 자체가 설계 비용이 많이 드는 작업이다. 시변 특성, 부정확한 모델링 등에 대한 대처방법은 실시간 적응 보상 (adaptive compensation)을 필요로 한다. 또한 모르는 시스템을 모델링하기 위해 비선형 시스템의 식별 (system identification) 기법을 사용할 수가 있

다. 이때 임의의 비선형 시스템을 식별하기 위해서 범용 함수 근사자 (universal approximator)가 사용되며, 신경회로망이 범용근사자로서의 역할을 할 수 있음이 알려져 있다.

비선형 함수 근사자로서 신경회로망의 사상 (mapping) 기능이 많이 연구되어왔다. 이들 연구결과는 은닉층의 갯수, 샘플의 수, 학습횟수가 충분하다면 신경회로망이 임의의 비선형 사상을 학습하여 잘 표현할 수 있음을 보여준다. 이런 결과들에 근거하여 비선형 제어기법에서 신경회로망을 활용할 수 있다. 만약 1대1

---

\* 강원대학교 제어계측공학과 석사과정

\*\* 강원대학교 제어계측공학과 부교수

혹은 다대1 사상의 동특성을 학습할 수 있고, 동특성의 차수가 알려져 있으며, 시스템의 상태를 관측할 수 있으면 BPN이나 방사형 기저함수망 (radial-basis function networks)과 같은 신경회로망의 학습으로 비선형 시스템을 효과적으로 식별 (identification)하거나 제어할 수 있다.

기존의 사상 신경회로망 (mapping neural networks)은 1대1 사상에 대한 비선형 시스템에 대하여 적용되었다. 그러나 비선형 시스템의 역제어 (inverse control)는 때로 1대다 사상을 다루어야 한다. 이것은 기존의 신경회로망으로 해결할 수 없는 문제였다. 이 문제를 해결하기 위해 MRCCN[1]이 제시되었는데, MRCCN은 역사상 (inverse mapping)이 가능하지만 클러스터의 갯수가 많이 생겨서 학습시간이 상당히 길어진다. 이런 단점을 보완하기 위하여 본 논문에서는 MRCCN[2]과 BPN[3]을 결합한 형태의 신경회로망을 제시한다. 기존의 MRCCN에 비하여 클러스터의 크기를 크게하는 대신 갯수를 줄여서 학습속도를 향상시켰다. 그리고 각 클러스터마다 간단한 형태의 BPN을 사용하여 사상기능을 개선하였다. 비선형 시스템에 이와 같은 다중 역전과 신경회로망을 적용한 모의실험을 통해 성능을 평가하였다.

## 2. 비선형 시스템의 표현

이 논문에서 다룰 문제를 정의하기 위하여 비선형 SISO 시스템을 아래와 같이 정의하자.

$$y(k+1) = f(y(k), y(k-1), \dots, y(k-p+1), u(k), u(k-1), \dots, u(k-q+1)) \quad (2.1)$$

여기서  $u(k)$ 와  $y(k)$ 는 시스템의 입출력을 나타내고,  $f(\cdot)$ 는 비선형 동특성을 표현하는 비선

형함수이다. 식(2.1)은 가관측성 조건이 성립한다고 가정한다. 식(2.1)을 간단히 하기 위하여 아래 식과 같이 정의하면

$$\phi(k) \equiv [y(k) \ y(k-1) \ \dots \ y(k-p+1) \ u(k-1) \ \dots \ u(k-q+1)]^T \quad (2.2)$$

모든  $k$ 에 대해서  $\phi \in \Phi \subset R^{(p+q-1)}$ 이다.

식(2.2)를 이용하여 식(2.1)을 다시 쓰면 식(2.3)과 같이 표현할 수 있다.

$$y(k+1) = f(\phi(k), u(k)) \quad (2.3)$$

식(2.3)을  $u(k)$ 와  $y(k)$ 의 입출력관계를 강조하여 표현하면

$$y(k+1) = f(u(k)|\phi(k)). \quad (2.4)$$

앞으로 다룰 비선형 시스템에 대하여 비선형 함수  $f(\cdot|\phi): R \rightarrow R$ 는 모든  $\phi \in \Phi \subset R^{p+q-1}$ 에 대하여 1대1 혹은 다대1 사상 또는 일반적인 다대다사상까지도 표현한다. 은닉층의 갯수가 충분히 주어지고 학습 샘플의 수와 학습 횟수가 충분하다면 BPN과 같은 기존의 신경망회로망으로 식(2.1)과 같은 비선형시스템의 모델을 학습할 수 있다.

## 3. 신경회로망의 소개

### 3.1 MRCCN

시스템의 동특성  $y(k+1) = f(u(k)|\phi(k))$ 은  $(u, \phi)$ 와  $y$ 사이의 일반적인 다대다 비선형 사상을 표현한다. 여기서  $y \in R^n$ ,  $u \in R^m$  이고  $\phi \in R^r$  이다. 이런 일반적인 다대다 사상을

3-tuple의 관계를 갖는 데이터  $\{(y, u, \phi)\}$ 로 표현할수 있다. 다시 말하면, 데이터  $\{(y, u, \phi)\}$ 는

$y, u, \phi$ 로 표현되는  $(n+m+d)$  차원의 입출력 공간에서의 샘플점  $\{z=(y, u, \phi)\}$ 을 나타낸다.

MRCCN은 초타원 (hyper-ellipsoidal) 형태의 지역 클러스터 (local cluster)들에 의해서  $(y, u, \phi)$  공간의 샘플 분포를 근사화하도록 학습하는 것에 목표를 두고 있다. 지역 클러스터는 초타원  $C(z; c, \Sigma, r)$ 으로 정의되며, 중심  $c \in R^{n+m+d}$ 와  $(n+m+d) \times (n+m+d)$ 차원의 양의 모양행렬 (positive definite shape matrix)  $\Sigma$  및 반경  $r \in R$ 로 표현된다:

$$C(z; c, \Sigma, r) = \{z \in R^{n+m+d} \mid (z-c)^T \Sigma^{-1} (z-c) \leq r^2\} \quad (3.1)$$

클러스터의 모양행렬  $\Sigma$ 와 중심  $c$ 는 클러스터에 속한 샘플의 평균과 공분산을 나타낸다.

MRCCN의 학습은 서로 다른 중심, 모양, 크기를 갖는 지역 클러스터의 필요한 갯수를 자기조직화하여 사상의 정확성을 높이는 것이다. 즉, MRCCN의 자기조직화 기능은 전체 샘플을 포함할수 있는 지역 클러스터를 자동적으로 생성하는 것이다. 반면에 학습이 진행될수록 사상의 정확성을 높이기 위하여 지역 클러스터의 중심, 모양, 크기를 수정하게 된다.

MRCCN[5]은 크게 globally competitive clustering 알고리즘과 locally cooperative interpolation 알고리즘으로 나누어진다. 첫 번째 알고리즘은 입출력공간을 다양한 모양, 위치, 크기를 가진 지역 클러스터로 나누고, 두 번째 알고리즘을 이용하여 사상을 형성할 수 있다. 본 논문에서는 사상을 위해 두 번째 알고리즘을 사용하지 않고 대신 간단한 형태의 BPN을 사용하였다.

### 3.1.1 Globally competitive clustering 알고리즘

이 알고리즘은 아래와 같은 기본적인 원칙에 따라서 지역 클러스터의 중심, 모양, 크기를 결정한다.

- (1) 샘플은 자신과 지역 클러스터의 중심간의 Mahalanobis 거리가 최소로 되는 지역 클러스터에 할당되고 이 샘플을 포함한 지역 클러스터의 중심만 갱신한다.
- (2) 지역 클러스터의 갯수는 모든 샘플이 적어도 하나의 지역 클러스터에 속하도록 선정한다. 한 샘플이 특정 지역 클러스터의 반경안에 있으면 그 지역 클러스터가 샘플을 포함하게 된다.
- (3) 지역 클러스터의 모양은 클러스터에 포함된 샘플의 공분산에 의해서 결정된다.
- (4) 지역 클러스터의 위치, 모양, 크기는 계층적 학습(hierarchical learning)에 의해서 결정된다. 즉, 초기에는 반경을 크게하고 학습이 진행됨에 따라서 반경을 작게 하여 크기와 모양을 조정한다.

각 지역 클러스터의 중심과 주어진 입력에 대한 Mahalanobis 거리는 아래와 같은 식으로 정의된다:

$$d(z, C) = [(z-c)^T \Sigma^{-1} (z-c)]^{1/2} \quad (3.2)$$

위의 원칙에 따른 재귀적인 갱신 방법은 아래와 같다.

알고리즘 : Clustering with iterative update

입력 :

$\{z_i \mid z_i = (y_i, u_i, \phi_i), i=1, \dots, N_d\}$  : 샘플

$r_0$  : 초기의 클러스터의 반경

$E_0$  : 오차한계치

출력 :

$$\{C_j | C_j = C(z_i, \Sigma_j, r_j), j=1, \dots, N_c\}$$

: 중심, 모양, 크기로 정의되는 지역 클러스터의 집합

방법 :

step 1. 초기화

초기샘플  $z_i$ 을 임의로 선택하여 지역 클러스터  $C_1$ 에 다음과 같이 할당한다:

$$c_1 = z_i, \quad \Sigma_1 = I_{n+m+1}, \quad r_1 = r_0.$$

학습주기  $k$ 와 지역 클러스터의 갯수  $N_c$ 를 각각 1로 초기화한다.

step 2. 주어진 샘플을 이용하여 각 지역 클러스터의 중심과 모양을 다음과 같이 갱신한다.

- (1) 임의의 샘플  $z_i$ 을 선정한다.
- (2) 샘플과 지역 클러스터의 중심간의 Mahalanobis 거리가 최소인 winning cluster  $C_w$ 를 결정한다.

$$d(z_i; C_w) = \min d(z_i; C) \quad (3.3)$$

- (3) 샘플이 지역 클러스터의 반경안에 포함되면, 즉  $(z_i - c_w)^T \Sigma_w^{-1} (z_i - c_w) \leq r_w^2$  이면 지역 클러스터의 중심과 모양을 아래식으로 갱신한다:

$$\begin{aligned} c_w &\leftarrow c_w + \alpha(n_w) \cdot (z_i - c_w) \\ \Sigma_w &\leftarrow \Sigma_w + \beta(n_w) \cdot (z_i - c_w) \\ &\quad \cdot (z_i - c_w)^T \end{aligned} \quad (3.4)$$

학습계수  $\alpha(n_w)$ 와  $\beta(n_w)$ 는 클러스터에 포함된 샘플의 수가 증가함에 따라서 단조 감소한다.

- (4) 만약 샘플이 어떤 지역 클러스터에도 포함되지 않으면  $N_c = N_c + 1$ 로 하고 새로운 클러스터  $C_{N_c}$ 를 생성하여 아래와 같이 초기화 한다:

$$c_{N_c} = z_i, \quad \Sigma_{N_c} = I_{n+m+1}, \quad r_{N_c} = r_k (\leq r_{k-1}).$$

step 3. 다음 샘플을 학습시킨다.

step 4. 모든 학습샘플에 대하여 전부 학습할 때까지 step 2로 되돌아간다.

step 5. 모든 학습샘플을 사용한 후에 지역 클러스터를 평가한다.

각 지역 클러스터  $C_j$ 의 사상 성능  $E_{C_j}$ 을 평가한다. 샘플이 속한 지역 클러스터의 중심과 샘플간의 평균제곱 Mahalanobis 거리를 구한다:

$$\begin{aligned} E_{C_j} &= \frac{1}{M_{C_j}} \sum_{j=1}^{M_{C_j}} e_j^2, \\ e_j^2 &= (z_j - c_{C_j})^T \Sigma_{C_j}^{-1} (z_j - c_{C_j})^T \end{aligned} \quad (3.5)$$

step 6. 수렴조건을 판별한다.

각 지역 클러스터의 사상 성능  $E_{C_j} \leq E_0$  이면 학습을 중단하고 그렇지 않을 경우에는 그 지역 클러스터를 지운다.

step 7. 학습의 종료를 판별한다.

모든 지역 클러스터가 수렴조건을 만족하거나 줄일 클러스터의 크기가 없으면 학습을 종료한다. 그렇지 않을 경우에는 지역 클러스터를 줄이고 step 2로 되돌아 간다.

## 3.2 Backpropagation Neural Network

참고문헌[4]에 언급된 BPN 알고리즘을 그대로 적용했고, 다만 뉴런의 활성화 함수로 비선형 특성을 갖는 하이퍼볼릭 탄젠트 함수를 사용했다.

## 3.3 다중 역전파 신경회로망

3.1절에 제시한 MRCCN 알고리즘을 이용하

여 입출력공간을 여러 개의 지역 클러스터로 나눈다. 각각의 지역클러스터는 비선형특성의 선형화와 같이 보다 단순화된 입출력 특성을 표현한다. 지역클러스터 내에서의 정확한 사상은 해당 지역 클러스터에 속한 샘플들을 이용하여 BPN을 학습함으로써 이루어진다. Fig.1은 MRCCN과 BPN을 결합한 다중역전파 신경회로망을 보여주고 있다.

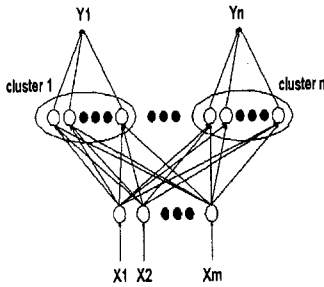


Fig. 1. Multit-BPN

다중역전파 신경회로망 알고리즘은 다음과 같다:

- step 1. 샘플을 MRCCN에 인가한다.
- step 2. MRCCN으로 클러스터링을 수행한다.
- step 3. 각 지역 클러스터에 대하여 BPN을 수행한다.
- step 4. 모든 학습샘플에 대하여 학습할때 까지 step 1로 돌아간다.
- step 5. 학습샘플을 다 사용한 후에 지역 클러스터의 성능을 평가한다.
- step 6. 수렴조건을 판별한다.  
각 지역 클러스터의 사상성능을 평가하여 정해진 수렴조건을 만족하면 지역 클러스터를 유지하고 만족하지 않으면 지역 클러스터를 지운다.
- step 7. 각 지역 클러스터의 반경을 미리정

한 크기로 줄인다.

- step 8. 각 지역 클러스터의 크기를 다 사용했으면 학습을 종료하고 그렇지 않을 경우 step 1로 되돌아 간다.

## 4. 모 의 실험

아래와 같은 비선형 시스템에 대한 모델링의 모의 실험 결과를 BPN, MRCCN, 다중역전파 신경회로망에 대하여 비교하였다.

$$y(k+1) = \frac{4y(k)}{1+y^2(k)} + u^2(k) - 2 \quad (4.1)$$

Fig. 2는 식(4.1)의 동특성을 보여주고 있다. Fig. 2의 (b)를 보면  $u(k)$ 와  $y(k+1)$ 의 사상이 다대1 사상이며 입력 구간  $[-2, 2]$ 에 대해서 대칭된 형태를 보인다. 참고문헌[5]에 따르면 BPN의 경우 평균을 학습하기 때문에 1대1 사상인 경우에는 제대로 학습할 수 없게 된다. 또 MRCCN의 경우에는 정확성을 높이기 위하여 지역 클러스터의 반경을 작게 하기 때문에 지역 클러스터의 갯수가 늘어나서 학습 시간이 상당히 길어지게 된다.

우선 다중역전파 신경회로망의 경우 다대1 모델링 능력을 알아보기로 하겠다. 비선형 사상  $y(k+1) = f(u(k)|y(k))$ 를 학습하기 위하여 무작위로 9만개의 학습샘플을 만들었다. 다중역전파와 신경회로망은 MRCCN의 지역 클러스터의 반경을 4.0, 3.5, 3.0순으로 선정하였다. MRCCN의 클러스터링과 병행하여 각 지역 클러스터에 대해  $N_{2,5,1}$  구조의 BPN을 학습시켰다. 이때 결선강도에 대한 학습계수는 0.13을 사용했다. 단일 BPN의 경우 결선강도에 대해 0.05, 바이어스항에 대해 0.12의 학습계수를, 관성에 대해 0.25를 사용하여  $N_{2,40,20,1}$  구조를 학습하였다.

$$u(k) = \sin \frac{2\pi k}{25} + \sin \frac{2\pi k}{10} \quad (4.2)$$

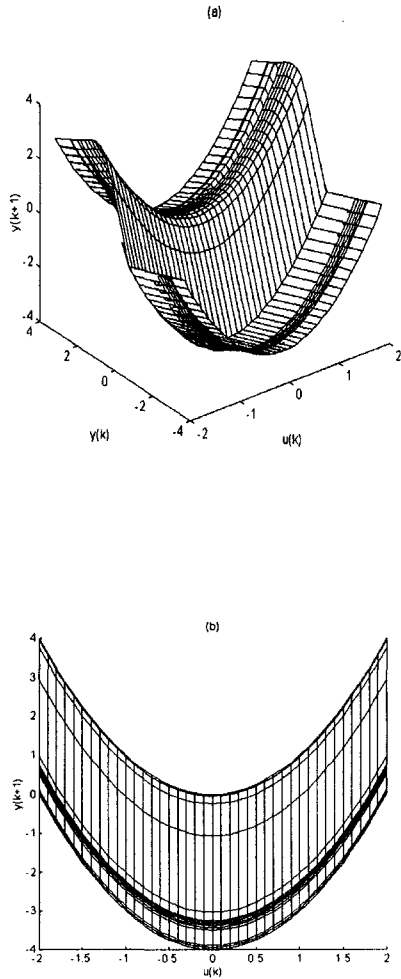


Figure 2. The dynamics of Eq. (4.1)

MRCCN의 경우 지역 클러스터의 크기를 2.0, 1.2, 0.8, 0.6, 0.4로 선정하여 학습시켰다. 각 신경회로망의 학습이 끝난 이후 식 (4.2)를 사용하여 얻은 결과는 Fig. 3~5와 같다.

Fig. 3~5에서  $\hat{y}$  은 원하는 출력을 의미하고  $y$  는 각 신경회로망의 실제 출력을 나타낸다. 그림에서 (b)의  $\hat{y} - y$  는 원하는 출력과 실제 출력간의 오차를 의미한다.

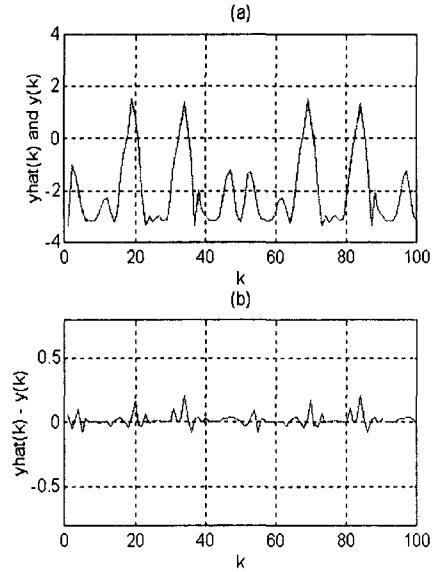


Figure 3. The Results of Mult-BPN

Table 1에서 시간은 workstation Axil 320에서 수행한 결과이다. 다대1 모의 실험 결과 각 신경회로망은 거의 비슷한 수준의 출력을 보이고 있다. 하지만 학습 시간의 측면에서 보면 BPN이 가장 빠르고 다중 역전파 신경회로망, MRCCN 순으로 나타났다.

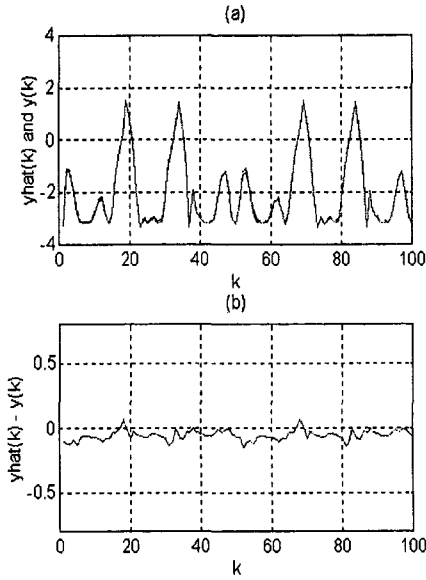


Figure 4. The Results of single BPN

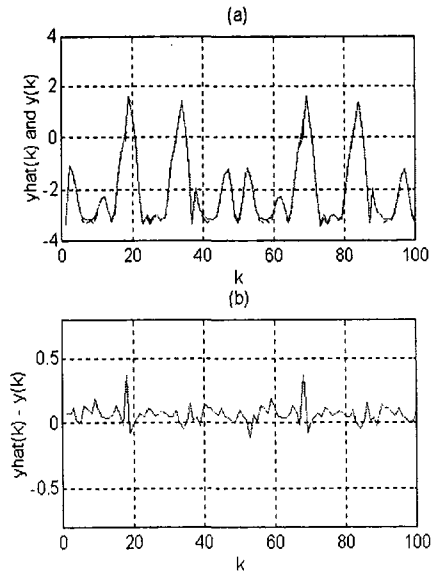


Figure 5. The Results of MRCCN

Table 1. Comparison of the number of connections and time (many to one mapping)

( time : 초 )

	cluster갯수	BPN총결선수	시간
다중-BPN	5	$5 \times 15 = 75$	1275
MRCCN	75		30295
BPN		881	20

비선형 1대다 사상 모델링 능력을 알아보기 위해 역사상  $u(k) = f^{-1}(y(k)y(k+1))$ 에 대해 학습샘플을 무작위로 9만개를 만들어 오프라인으로 각 신경회로망을 학습하였다. 다중 역전파 신경회로망과 MRCCN은 위와 동일한 인자를 사용하였고 단일BPN경우만  $N_{2,3,2,1}$ 구조로 학습하였다. 학습이 끝난 후 기준 입력은 식(4.3), 실제 시스템은 식(4.1)을 사용하고 제어기는 학습시킨 신경회로망을 사용하여 나온 결과가 Fig. 6~8에 나타나 있다.

$$y_m(k+1) = 0.5y_m(k) + r(k) \quad (4.3)$$

$$r(k) = \sin \frac{2\pi k}{50} + \sin \frac{2\pi k}{10}$$

Fig. 7의 단일 BPN의 경우에는 학습이 불가능하지만 MRCCN의 경우와 다중 역전파 신경회로망은 학습하였다. Table 2에 따르면 다중 역전파 신경회로망이 MRCCN보다 학습속도가 빠른 것을 알 수 있다.

Table 2. Comparison of the number of connections and time (one to many mapping)  
( time : 초 )

	cluster갯수	BPN총결선수	시간
다중-BPN	5	$5 \times 14 = 70$	1437
MRCCN	64		24652
BPN		학습불가	20

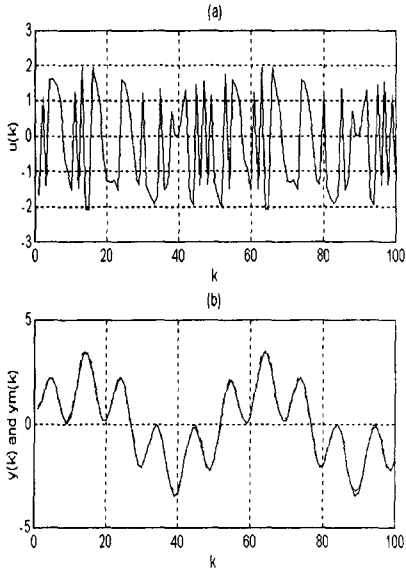


Figure 6. The Results of Multi-BPN

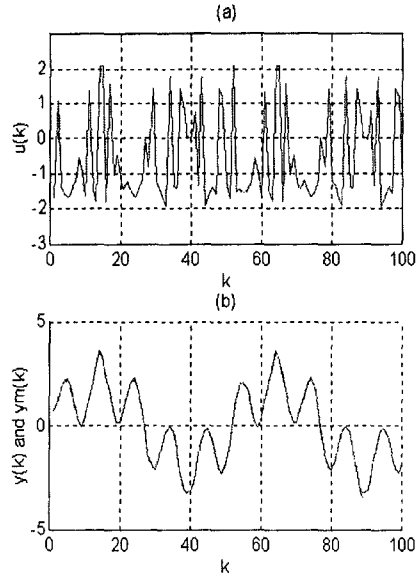


Figure 8. The Results of MRCCM

## 5. 결 론

다중 역전파 신경회로망은 MRCCN의 다대다 사상기능을 가지고 있을 뿐만 아니라 BPN의 비선형함수 근사기능을 가지고 있다. MRCCN의 학습에서는 지역 클러스터의 크기를 크게 함으로써 학습속도에 악 영향을 주는 지역 클러스터의 갯수를 줄였고, 지역 클러스터 내의 입출력 특성만을 BPN으로 학습함으로써, BPN의 총결선수를 줄일 수 있었다. 다중 역전파 신경회로망을 비선형 시스템의 모델링에 적용한 결과, 기존의 신경회로망 보다 적은 규모의 신경회로망으로 동등한 수준의 성능을 얻을 수 있었다.

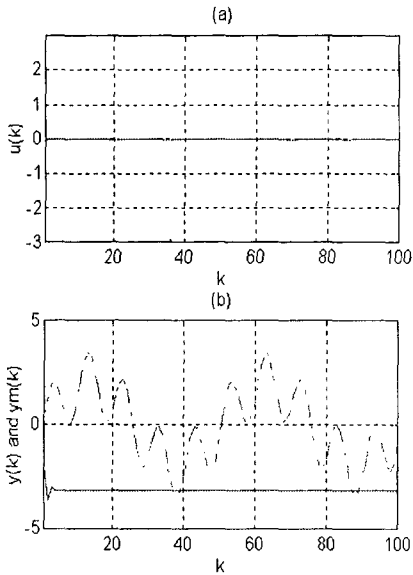


Figure 7. The Results of single-BPN



## 참 고 문 헌

- [1] Jianting Cao and Takashi Yahagi, Nonlinear Adaptive Digital Filters Using Parallel Neural Networks, *ICNN*, Vol. 2, pp.850-853, 1995.
- [2] C.H. Chen, Fuzzy Logic and Neural Network Handbook, *IEEE*, 1996.
- [3] James A. Freeman and David M. Skapura, Neural Networks, Algorithms, Applications and Programming Techniques, *Addison-Wesley*, 1992.
- [4] Patrick M. Kelly, Don R. Hush, James M. White, Adaptive Algorithms for Modifying Hyperellipsoidal Decision Surfaces, *Proc IJCNN* Vol. 4, pp.196-201, 1992.
- [5] Sukhan Lee and Jung-Moon Lee, Nonlinear system control based on multi-resolution radial-basis competitive and cooperative networks, *Neurocomputing*, pp.187-206 1995.