

기계상태의 변화를 온라인으로 탐지하기 위한 Radial Basis 하이브리드 뉴럴네트워크 모델링

Radial Basis Hybrid Neural Network Modeling for On-line Detection of Machine Condition Change

왕지남*, 김광섭*, 정윤성*

Gi-Nam Wang*, Gwang-Sub Kim*, Yoon-Seong Jeong*

Abstract

A radial basis hybrid neural network (RHNN) is presented for an on-line detection of machine condition change. Two-phase modeling by RHNN is designed for describing a machine condition process and for predicting future signal. A moving block procedure is also designed for detecting a process change. A fast on-line learning algorithm, the recursive least square estimation, is introduced. Experimental results showed the RHNN could be utilized efficiently for on-line machine condition monitoring.

1. 서론

최근 생산자동화와 컴퓨터에 의한 통합된 생산체제에서는 인간에 의한작업과 관리를 기계와 컴퓨터가 많은 부분을 대신하게 되었다. 이에 따라 작업과정중의 기계나 설비의 이상현상을 감시하고 진단하며 각종 센서에 의하여 실시간에 수집된 신호를 분석하여 시스

템의 이상원인을 컴퓨터에 의하여 자동감시하는 시스템에 많은 관심이 모아지고 있다. 자동화된 시스템에서 기계나 설비의 이상은 전체 생산성과 품질에 영향을 미칠뿐만 아니라 커다란 재해나 위험을 야기 시킬 수 있기 때문이다.

일반적으로 시스템의 진단과 예측은 정상 상태 혹은 이상상태를 미리 정해진 모델이나

* 아주대학교 산업공학과

본 연구는 아주대학교 교내 연구비 지원에 의하여 수행 되었음

법칙을 만들어 시스템을 예측하거나 진단하는 방법에 의존하고 있다. 발생가능한 시스템이 완전하게 정확하게 알려진 모델들 내에서 표현 가능하면 시스템에 대한 예측은 정확하다. 그러나, 시스템이 복잡하고 변화되는 경우에 센서에서 얻어진 신호들에 대한 모델링 작업은 일반적으로 어렵고 특히 모델구조가 비선형인 경우는 더욱 힘든 과제다. 또한 모든 이상상태에 대한 시스템의 특성을 미리 알아서 각각의 경우를 참고모델(reference model)로 구축하는 작업은 가능한 이상상태를 전부 생성시켜야 하므로 공정이 동적으로 변하며 이상여부도 복합적으로 나타나는 상황에서는 어려운 문제인 것이다.

이런 문제점을 극복하고 효율적인 시스템의 감시 및 진단시스템을 설계하기 위해서는 첫째로 센서로부터 수집된 신호들의 관계를 정확하며 쉽게 모델링할수 있는 방법이 요구된다. 또한 모든 발생 가능한 이상현상을 미리 알수 없으므로 정상상태에서 벗어나는 시점을 빨리 찾아 내고 변화된 이상원인을 현재의 정보로 진단할수 없을 경우에는 학습하여 앞으로 유사한 상황에 당면 하였을때 정확하게 원인을 알려줄 수 있도록 온라인 이상원인 학습체제가 필요하다.

설비나 장비의 상태를 온라인으로 감시(Monitoring)하는 방법은 최근에 여러가지 형태로 연구되어왔다. 기본적인 방법은 다음과 같은 3가지로 요약할수 있다. 첫째로, 각종센서에 의하여 시스템 상태가 측정되어 진다. 둘째로, 측정된 신호들의 유형들을 분석한다. 셋째로, 분석된 신호의 유형으로부터 시스템 상태를 진단한다. 그림 1은 일반적인 기계 및 설비상태를 센서에 의한 감시 및 진단하는 과

정을 다음과 같은 그림으로 나타 낼수 있다.

본 연구에서는 우선 회전하는 설비를 진동 센서에 의하여 설비의 가동상황을 모델링을 통하여 설비상태를 나타낼수 있는 과형을 예측하며 회전설비에 이상이 발생하였을때 발생시점을 탐지하는 방법을 제시한다. 회전설비에 이상이 발생하였을때 탐지하는 과정과 각각의 설비상태를 온라인으로 모델링하는 방법은 적응적인 학습과정이 필요하다. 온라인 모델링 알고리즘으로 Recursive Least Square Estimation을 이용한 Radial 함수에 근거한 복합적인 신경망(Radial basis hybrid neural network)을 제시 한다.

2. 신경망에 의한 시스템 모델링

신경망의 모델링 성능은 많은 문헌에서 우수성을 보였으며 [7,8,11] 특히 최근에는 시간지연 입력형 신경망 [12](Time delay neural network)을 설계하여 모델링의 성능을 더 하였다. 특히 계층적인 신경망을 정규분포의 난수로 생성된 임의의 Noise들을 입력과 함께 사용하여 시스템 모델링의 정확도를 향상시킨 것은 주목 할만 하다 [13]. 그러나 시스템이 변하는 상황에서는 미리 발생될수 있는 모든 경우의 모델을 찾아 Off-line에서 학습하기는 어렵다. 또한 하나의 모델로 변하는 시스템을 서술하는 것은 모델링의 정확도를 감소 시킬것 이다. 따라서 상이한 시스템이 발생할때 경계점을 미리 탐지하는 것은 서로 구별된 모델구조 및 모델변수를 사용하여 모델링의 정확도를 증대 시킬 것이다. 또한 신경망에 의한 모델링시 상이한 모델로 구분함으로써 정보의 충돌현상을 피할 것이다. 이

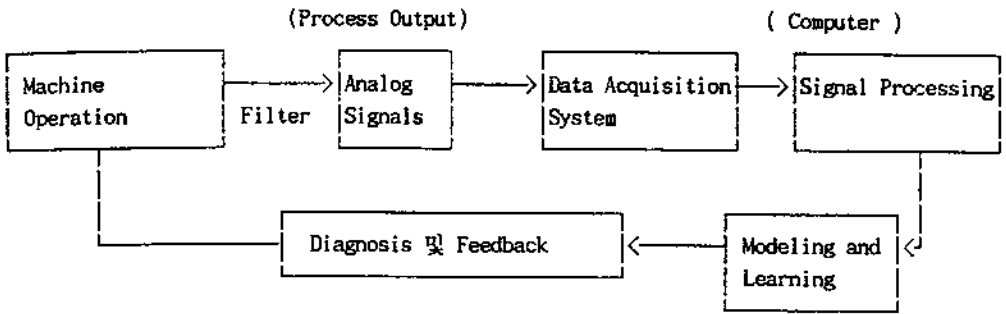


Figure 1. An Overview of Machine Condition Monitoring System

정보의 충돌현상은 신경망 학습시 수렴되지 않는 경우를 야기시키며 학습된 정보의 정확도를 감소시키는 것이다. 그러므로 본 논문에서는 학습의 충돌현상을 막고 모델링의 정확도를 증가시키수 있는 변화탐지기능을 지닌 2단계 신경망 모델링 방법을 제시한다.

그림 2는 제시되는 Radial 함수에 근거한 하이브리드 신경망 방법을 간단히 나타낸 것이다.

이용하여 얻어진 시그널에 대한 전체적인 모델링 과정이며 모든 입력시그널의 전체모델(Global model) 확립과정으로 Radial 함수 신경망 1으로 표시된다. 두번째 모델링 과정은 확립된 전체모델을 사용하여 얻어진 모델링 에러(Innovation)들을 입력으로 하여 재모델링하는 과정으로 세부모델(Local model) 확립과정으로 표현된다. 첫번째 모델링 과정이 대략적인 근접(Approximation)된 모델링 이

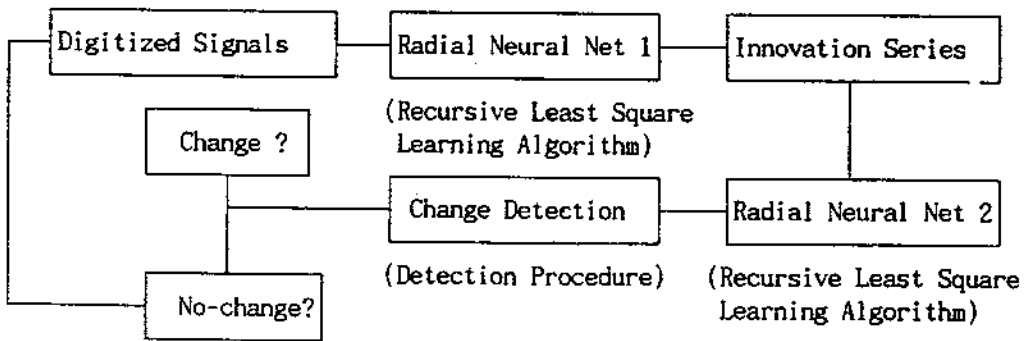


Figure 2. Radial Basis Hybrid Neural Network Scheme

그림 2에서 나타난 것처럼 Radial함수에 근거한 2개의 뉴럴네트워크를 이용하여 2단계 모델링 과정과 변화탐지과정이 함께 있다. 우선 첫번째 모델링 과정은 뉴럴네트워크 1을

라면 두번째 방법은 보다 정확하게 정밀화(Polishing)하는 과정으로 설명된다. 이러한 양분된 뉴럴네트워크에 의한 모델링 방법을 본 논문에서는 2단계 모델링(Two-Phase

Modeling)이라 정의한다. 2단계 모델링의 목적은 근사화(Approximation)과 정밀화(Polishing)의 2단계를 거쳐 모델링의 정확도를 높여 정확한 파형을 예측하는데 있으며 예측된 파형의 에러를 분석하여 파형의 변화를 탐지하는데 있다. 2단계 모델링시 Radial함수에 근거한 뉴럴네트워크를 사용하였는데 이는 기존의 Sigmoid나 기타 함수를 사용하는 방법보다 온라인 학습알고리즘을 설계 가능하려는데 목적이 있다 [2][5].

3. Radial Basis Modeling

Radial 함수는 최근에 분포된 데이터의 보간(Interpolation)이나 함수나 곡면의 곡선을 추정하는데 널리 사용되어 왔으며 Radial 함수들의 선형조합을 이용하여 함수들을 표현하였다. 절삭공정에서의 공구마모 상태를 모델링하는데 신경망에 Radial 함수를 사용하여 공구마모를 추정하였다 [2].

본논문에서 사용된 Radial 함수는 Elanayar 과 Shin[2]이 사용한 함수와 유사한 함수를 사용하였으며 그림 3과 같이 기계설비상태의 센서에 의한 입력신호에 대하여 입출력 관계를 설계하였다. 입출력의 구조는 입력으로써 현재 시간을 기준으로 과거 10개(일반적으로 N개)의 신호들을 사용하였고 출력으로는 현재시간의 신호로 구성되어 있다. 그러므로 10개의 입력과 1개의 출력을 갖는 구조로 과거의 신호들과 현재의 신호와의 관계를 모델링하는데 사용된다. 일반적인 Multi-layer perceptron에 비교하여 Radial 함수에 근거한 신경망은 학습시간을 줄일수 있는 장점이 있으며 이는 온라인으로 연결변수들을 학습할 수

있는 것이 주요 장점인 것이다.

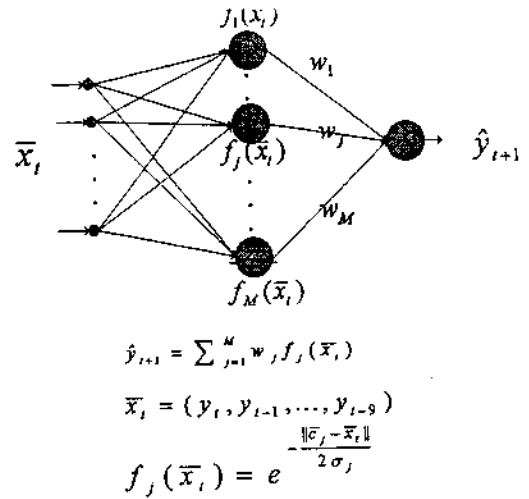


Figure 3. The Radial Basis Neural Network

그림 3은 입력벡터에 대한 하나의 출력과 하나의 중간층으로 구성된 Radial함수 신경망을 보여 준다. 중간층에는 Radial 함수로 구성되고 일반적으로 사용되는 변환함수는 정규분포 확률함수의 형태로 중심벡터에 대하여 대칭인 함수형태를 갖는다. 주어진 입력, \bar{x}_i 에 대하여 중간층의 Node j 를 통과하는 식은

$$f_j(\bar{x}_i) = e^{-\frac{\|\bar{x}_i - \bar{c}_j\|^2}{2\sigma_j^2}} \quad (1)$$

로 표시되며 여기서 \bar{c}_j 는 중간층에 있는 Node j 의 Radial 함수 중심벡터이며 σ_j 는 j Radial 함수의 분산변수이다. 그러므로 주어진 입력 \bar{x}_i 에 대하여 만약 M 개의 Node를 갖는 신경망의 출력 \hat{y}_{i+1} 는 각 함수들을 연결부의 선형조합으로 다음과 같이 주어진다.

$$\hat{y}_{i+1} = \sum_{j=1}^M w_j(t) f_j(\bar{x}_i) \quad (2)$$

신경망의 중간층에 있는 전체 Node의 수와 각 Node에 있는 Radial 함수의 변수인 중심벡터와 분산은 문제의 입력 space에 따라 결정하는데 앞으로 입력될 데이터의 전체영역(space)을 각함수에 있는 중심벡터가 잘 나타내고 왜곡(skewness)이 없도록 결정함으로 전체 시스템 모델링의 성능을 증대 시킬수 있는 것이다. 중심벡터의 의미는 모든 가능한 입력공간(Input Space)을 정해진 Node의 수로 Clustering되어 분할되어 질때 각분할된 Cluster의 대표이며 이는 수학적으로 해당 Cluster의 평균이다. 분산변수는 각 Cluster대표들의 인접된 정도를 의미한다.

Node의 수가 증가 할수록 많은 분할된 Cluster의 수를 증대시켜 주어진 입력공간내의 각 Cluster간의 거리를 감소 시킨다. 이는 Radial 함수의 분산변수를 작게하나 학습해야 할 연결부가 많아져 복잡한 입력신호에 적합하나 학습시간이 길어질 것이다. 그러므로 중간층에 있는 각 Node에는 두개의 결정변수가 존재하며 이들은 중심벡터와 분산변수이다. 미리 주어진 학습 데이터를 이용하여 Off-line으로 Kohonen 신경망 [6]이나 K-means 알고리즘 [15] 혹은 ART 2 [3] 등의 Clustering Algorithm을 사용하여 중간층의 전체 Node수와 중심벡터 그리고 분산변수를 결정한다. 본 논문에서는 각 Node에 있는 함수의 분산변수는 주어진 중심벡터들의 가장 가까운 이웃 Node들간의 중심벡터 거리의 절반으로 결정 하였다 [12].

4. 기계상태의 2단계(two-phase) 모델링 과정

센서에서 얻어진 신호들은 시간 영역에서 시계열(Time series)로 표시되며 얻어진 신호들에 대한 모델링 과정은 우선 모델구조와 정해진 구조(e.g. 입출력의 형태, 중간층의 Node수)에 해당하는 변수들을 결정하는 단계로 시작한다. 또한 추정된 모델의 성능평가는 모델의 1단계 예측에러(Innovation)들의 분석으로 일반적으로 모델로 부터 얻어지는 일단계 예측에러(Innovation)들의 정확도나 예측에러에 대한 White-noise 검사로 평가된다. 본 논문에서는 Radial함수의 2단계 신경망에 의한 모델링방법을 제시한다.

4.1. 2단계 모델링의 신경망 구조

기계설비들의 조건을 나타나는 시간별 신호에 대하여 본 논문에서 설계할 신경망의 입력은 시간지연된 신호들을 입력벡터로 사용한다. 시간지연 입력은 이미 존재하는 논문[13,14,17]에서 모델링시 높은 정밀도를 보였다. 출력은 신호들의 미래의 예측치로 구성하며 다단계 예측시에는 해당수의 다수의 출력을 구성할 수있다.

그림 4.1과 그림 4.2는 본 연구에서 제안된 2단계 신경망 모델링 구조를 나타내고 있다. 각단계 모델링시 결정변수는 그림 3에서 보여준 중간층의 Node의 수 M 과 각 Node j 안의 변환함수에 있는 분산변수 σ_j 와 중심벡터 \bar{c}_j 이다. 1단계 모델링 즉 전체모델링(Global Modeling)에서는 입력으로 과거 10개의 신호 벡터, $\bar{x}(t) = (y_t, y_{t-1}, y_{t-2}, \dots, y_{t-9})$, 를 사용하여 바로 1단계의 신호 \hat{y}_{t+1} 예측하고 예측된 출

력, \hat{y}_{t+1} , 과 실제의 신호, y_{t+1} , 를 이용하여 1단계 예측에러, e_{t+1} 를 계산하고 있다.

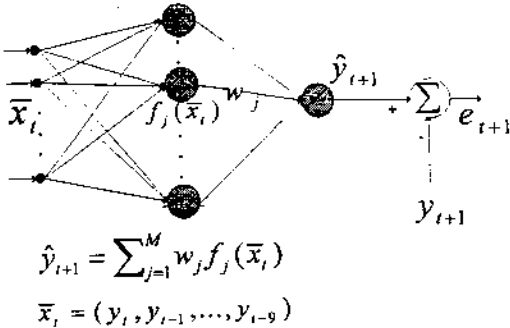


Figure 4.1 Phase-1 Radial Basis Neural Modeling

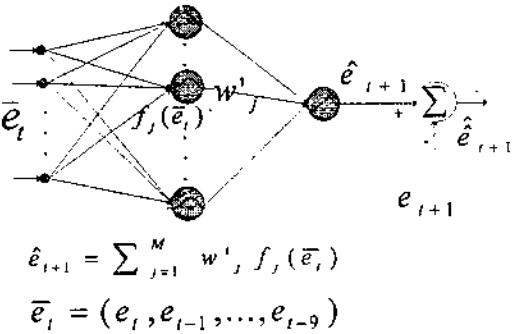


Figure 4.2 Phase-2 Radial Basis Neural Modeling

복잡한 비선형 시스템은 1단계 모델링만 이용하여 정확하게 시스템을 표현할수 없는 경우가 많다. 연결부를 시간에 따라 수정하여도 변하는 복잡한 신호들의 유형을 완벽하게 파악하기는 어렵다. 즉 전체모델링에서 얻어지는 모델링 1단계 예측에러는 완전하게 White-noise를 보장할 수가 없는 것이다. 이에 다시 전체모델링 1단계 예측에러들을 입력벡터, $\bar{e}_t = (e_t, e_{t-1}, \dots, e_{t-9})$ 로 하여 다음시점의 예측에러, e_{t+1} 를 재예측한 \hat{e}_{t+1} 를 출력으로 하는 세부모델링 과정을 반복한다. 그림 4.2는

세부모델링의 입출력 과정을 보여준다. 1단계에서 해당 모델링 예측에러를 계산하여 시간지연 모델링 예측에러를 2단계 입력벡터로 사용된다. 각단계별로 입출력의 관계는 그림 4.1과 4.2에서 보여주듯이 서로 종속적인 관계를 가지고 있다.

2단계 모델링과정은 확장하여 K단계 모델링을 설계할수 있다. 그림 4.3은 2단계 모델링방법을 이용한 전체적인 뉴럴네트워크의 입력과 출력을 보여주고 있으며 확장된 K단계 모델링을 이용한것이 그림 4.4에 나타나 있다. 최적의 단계수를 결정하는것은 신호의 복잡성과 시스템의 운영속도등에 따라 결정되어야 한다. 다단계로 증가할수록 학습시간은 증가하나 시스템 복잡성에 따라 다단계 모델링 설계는 응용 된다.

다단계 모델링 과정은 기존에 존재하는 Werbos et. al,[13]의 연쇄적인 예측방법과 연관되어 설명된다. 즉 다단계 예측을 위한 신경망의 구조는 주어진 신경망을 이용하여 입력과 출력들이 서로 연쇄적인 관계를 갖고 설계되어 전단계의 신경망의 일부 출력이 다음단계 신경망의 일부 입력요소로 사용되어 지는 과정이 반복적으로 이루어 진다(그림 4.5 참조). RHNN에 의한 다단계 모델링은 전단계의 신경망의 출력(\hat{y}_{t+1})을 이용하여 예측에러(e_{t+1})를 계산하여 다음단계 신경망의 입력요소(\bar{e}_{t+1})로 사용하여 다음단계의 예측에러(\hat{e}_{t+1})를 출력으로 예측한다. RHNN에 의한 방법은 전단계 출력이 직접 연쇄적으로 다음단계의 입력으로 사용되어 지는것이 아니라 전단계 출력을 사용하여 예측에러를 계산하여 입력벡터의 요소로 사용하고 결국은 모델링의 예측에러를 다시 재예측하여 최종

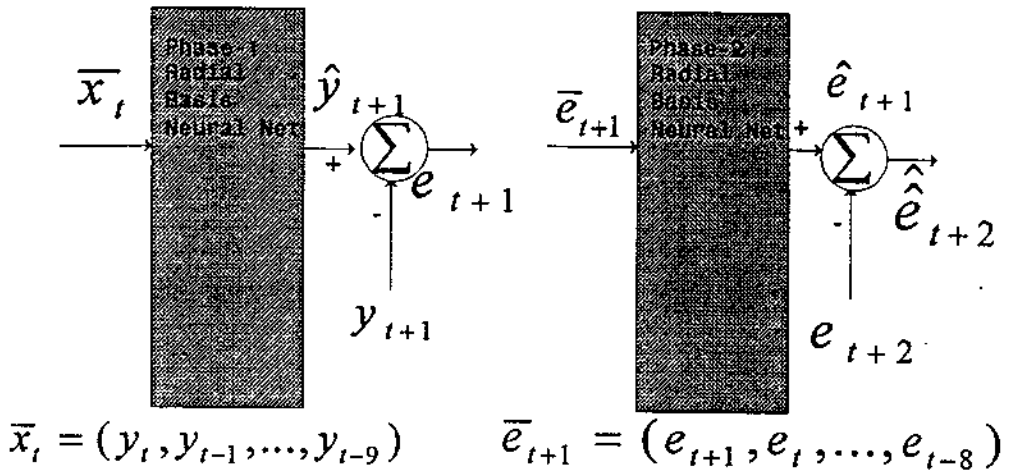


Figure 4.3 2-phase Radial Basis Neural Network

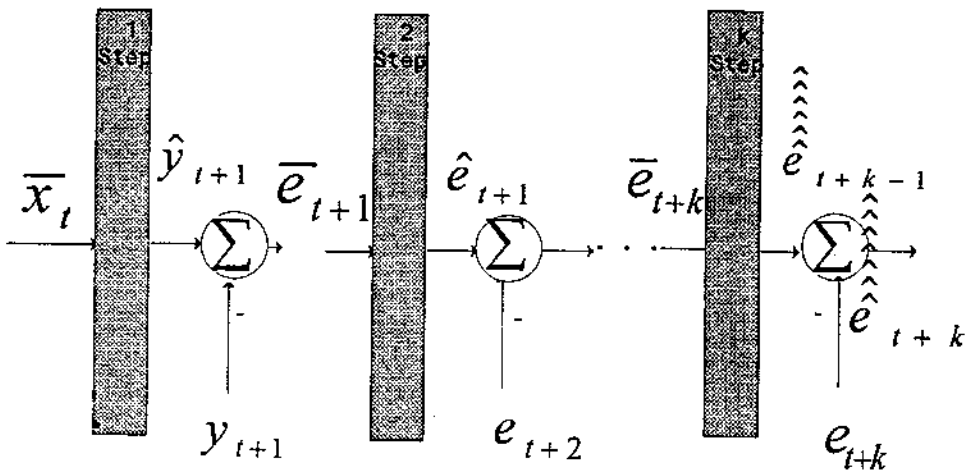


Figure 4.4. K-Phase Radial Basis Neural Network

적으로 모델링 예측에러를 최소화 하도록 반복적인 모델링 방법을 사용하고 있다.

Werbos et. al.[13]의 연구는 시스템의 다단계 이후를 예측하는 것이라면 제한된 K단계 RHNN은 모델링의 정확도를 높이기 위하여 반복적으로 생성되는 단계별 모델링의 에러를 이용하여 재모델링과정이라 설명된다.

중간층의 Radial 함수를 지닌 Node의 수는 Radial 함수 신경망의 구조 결정시 주요한 변수로써 이는 각 Node에 존재하는 각각의 중심벡터와 분산변수에 서로 종속적으로 영향을 받는다. 주어진 학습데이터에 대하여 많은 중간층의 Node수는 인접한 중심벡터간의 거리를 줄여 Radial 함수내의 작은 분산변수

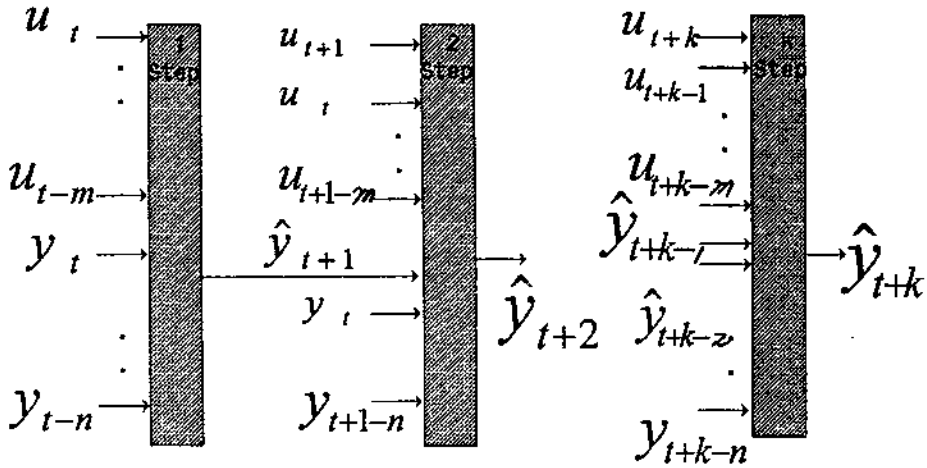


Figure 4.5. Forward Chaining of the Same Network, $k>n>m$ [Werbos et. al 1992]

를 갖게되고 상대적으로 중간층의 Node수를 작게 설계하면 큰 분산변수들을 갖게 될 것이다. 최적의 Node수를 결정하는 것은 매우 복잡한 비선형 최적화문제이고 휴리스틱한 Clustering방법으로 그 판계를 살펴 볼 수 있겠다. 그림 4.1-그림 4.2는 Radial 함수의 2단계 신경망 모델링방법을 나타낸 것이다. 1단계 신경망에서 중간층의 Node의 Radial 함수의 중심벡터들과 분산변수는 2단계 신경망의 Radial 함수의 중심벡터와 분산변수보다 크게 나타난다.

4.2. 신경망 학습

학습과정은 2단계의 신경망에 대하여 각각 2단계 학습(Two-phase learning)과정으로 구성된다. 첫째로 Off-line학습으로 주어진 학습 데이터에서 중간층의 Node수와 각 Node에 있는 중심벡터와 분산변수 그리고 초기 연결부들을 결정하는 단계이고 이는 초기에 신경망 구조설계시 진행되나 한번 구조가 학습으로 정해지면 Off-line학습은 중단된다. 두번째

학습과정은 온라인으로 계속하여 연결부를 입력되는 신호들을 이용하여 학습하는 과정으로 변하는 신호의 특성을 추적하기 위하여 계속 진행된다. 그러므로 본연구에서 사용된 학습과정은 신경망구조 결정의 Off-line학습과 연결부를 수정하는 On-line 학습으로 나뉘어 진다. Off-line학습으로는 Kohonen Learning Algorithm, K-means Clustering, 그리고 ART 2가 사용될수 있으며 연결부를 수집되는 신호에 따라 온라인으로 학습하는 단계에서는 Kalman Filter의 변환인 Recursive least square방법으로 신경망의 연결부를 계속 수정한다. 각 정해진 신경망 구조의 연결부를 온라인으로 학습하는 Recursive least square에 의한 학습과정은 다음과 같이 서술 된다.

$$y_{t+1} = \sum_{j=1}^M w_j(t) f_j(\bar{x}_t) = \phi^T(t) W(t) \tag{3}$$

$$W(t+1) = W(t) + \frac{P(t-1) \phi(t)}{1 + \phi^T(t) P(t-1) \phi(t)} \times [\mathcal{N}_{t+1} - \phi^T(t) W(t)] \tag{4}$$

$$\phi(t) = [f_1(\bar{x}_t), f_2(\bar{x}_t), \dots, f_M(\bar{x}_t)] \quad (5)$$

$$P(t) = P(t-1) + \frac{P(t-1)\Phi(t)\Phi(t)^T P(t-1)}{r + \Phi(t)^T P(t-1)\Phi(t)} \quad (6)$$

Covariance행렬 $P(t)$ 는 역시 순환적으로 다음과 같이 수정되며 여기서 r 은 측정치 Noise의 분산이다. \hat{y}_{t+1} 는 신경망의 출력을 나타내며 $W(t) = (w_1(t), w_2(t), \dots, w_M(t))$ 는 시간 t 에서의 신경망 연결부, $\bar{x}(t) = (y_t, y_{t-1}, y_{t-2}, \dots, y_{t-9})$ 는 시간 t 에서의 입력벡터, y_{t+1} 는 시간 $t+1$ 의 실제관측치 그리고 $f_j(\bar{x}_t)$ 는 중간층의 Node j 의 출력을 의미한다.

Recursive least square의 장점은 최적의 변수를 추정하는데 과거로부터의 필요한 모든 신호를 각 Iteration마다 전부 이용하지 않고도 새로 들어온 신호에 대해서만 순환적으로 계산하여도 최적의 변수를 구할수 있어 실시간으로 수집되는 신호에 대하여 실시간 모델링에 적합하다 [1]. Recursive least square 학습은 Off-line 학습에서 신경망의 구조와 중심벡터와 분산변수들이 정해진후 각 신경망 연결부를 새로 입력되는 신호에 대하여 학습하는 단계로 전체모델링과 세부모델링의 학습알고리즘으로 사용 된다.

Local modeling에 사용된 두번째 신경망의 학습과정도 1차 모델링을 경유(Filtering)해서 나온 모델링에러(Innovation)들을 다시 2차 신경망의 입력신호로 사용하여 동일한 학습 과정을 거친다. 학습의 알고리즘으로는 첫번째 신경망과 동일한 Recursive least square에 의한다.

4.3. 변화탐지과정

변화탐지과정은 신호의 특성이 변화하여

다른 특성을 갖을때 변화되는 시점을 빨리 찾아 내는데 목적이 있다. 이 과정은 기계나 설비의 상태가 정상상태에서 이상상태로 변화할때 변화되는 시점을 온라인으로 빨리 탐지하는 과정이다. 또한 뉴럴네트워크를 이용하여 시스템상태를 인식 하기위한 모델링시 서로 다른 상이한 시스템을 분리시켜 각각 고유의 시스템에 적절한 모델구조와 모델변수를 설정하여 정확도를 높이는데 있다. 이 과정은 신경망에 의한 모델링시 적용할 수 있다. 학습을 통하여 시스템 모델의 주요정보가 연결부에 얻어지게 되는바 학습시킬때 서로 상이한 시스템을 분리시켜 각각의 고유 특성에 적절한 신경망과 연결부를 구한다. 이는 학습시 발생하는 정보의 충돌을 막아 학습효과를 높이고 모델링의 정확도를 높이는데 있다. 만약 시스템의 변화가 탐지되면 변화가 탐지된 이후부터 기존에 가지고 있는 신경망과 분리하여 다시 변화된 새로운 시스템을 새로운 신경망에 변화시점 이후부터 학습한다. 이러한 과정은 두개의 상이한 시스템을 모델링하는데 서로다른 뉴럴네트워크를 분리시켜 설계하여 전문화된 모듈의 신경망을 구축한다.

본 논문에서 사용된 변화탐지과정은 연속적인 두개의 Reference Block과 Test Block을 시간에 따라 추적하면서 시스템 변화를 탐지하는 방법에 근거하고 있다. Test Block은 현재의 주어진 시간 t 를 기준으로 구간 $[t-L+1, t]$ 사이의 시스템의 모델링에의한 일단계 예측에러(Innovation)들로 구성되어 있다. Reference Block은 Test Block의 바로 인접된 이전 Block으로 $[t-2L+1, t-L]$ 구간의 동일한 모델링에 의한 Innovation들로 구성되어 있다.

변화 탐지과정의 기본방법은 Test Block과 Reference Block의 Innovation Series를 계속하여 증가하는 시간에 대하여 움직이면서 (Sliding) 비교하는 방법이다. 여기서 Innovation은 1단계와 2단계의 모델링을 거쳐 산출되는 모델링에러, \hat{e}_i ($i=1,2,\dots,t,\dots,L,\dots$)를 나타낸다. 만약에 Test Block과 Reference Block의 Innovation들이 동일한 특성을 갖지 아니하면 Test Block내의 어느시점 이후부터 시스템 변화가 발생한다고 가정하는 것이다.

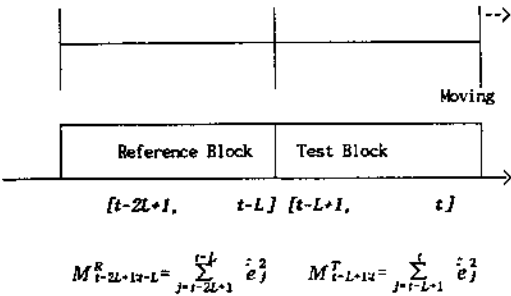


Figure 4.6. Change Detection Procedure

위 그림 4.6 에서 \hat{e}_i 는 시간 t 에서의 본 논문에서 제시한 2단계 모델링에 의한 일단계 예측에러이다. 모델링이 이론적으로 정확하다면 에러들은 White Noise특성을 갖게 되어 Reference Block과 Test Block의 White-Noise test도 가능하다. 또한 Reference Block과 Test Block의 예측에러의 척도인 $M_{t-2L+1:t-L}^R$ 과 $M_{t-L+1:t}^T$ 은 이론적으로 각각 자유도가 L 인 Chi-Square분포를 따르게 되어 통계적인 분석도 가능하다. 그러나 일반적으로 제시한 모델링의 예측에러가 White-Noise가 아닌 경우에는 Reference Block과 Test Block의 각각 계량치의 크기 비교로서 경험적인 방법에 의존된다. 본 연구에서는 두 계량치의 차이의 절대값이

지정된 값(Threshold)보다 크면 변화로 판정하였다.

5. 실험결과

본 연구의 실험을 위하여 그림 5와 같은 회전하는 모형기계를 제작하였고 센서로는 진동측정기를 사용하였으며 A/D Converter 데이터 수집장치를 IBM PC-486시스템을 이용하여 실시간으로 1 KHZ의 샘플링 비율로 신호를 받아 분석하였다. 우선 (1) 정상상태로 제작된 Sample설비를 가동하여 신호를 수집하여 모델링하고 인위적으로 베어링의 각 부분에 상처를 내어 신호를 수집하였고 (2) 외륜에 이상이 있을때와 (3) 내륜에 이상이 있을때 (4) 베어링 볼에 이상이 있을때 (5) 축에 풀림이 있을때 (6) 밸런스가 불량일 경우 각각에 대하여 수집한 신호를 이용하여 본 연구에서 제시한 2단계 신경망 모델링방법을 적용하였다. 외륜 그리고 내륜에 이상이 있을때는 인위적으로 베어링의 해당 부위에 인식가능한 상처를 내었다. 상처의 크기는 육안으로 식별가능한 최소의 상처를 베어링에 내었고 밸런스가 불량한 경우는 전동 회전체를 조립할때에 사용된 조립나사를 완전히 조이지 않은 상태를 의미한다.

신경망의 입력으로는 시간 t 를 기준으로 과거의 10개의 신호를 지닌 입력벡터(\bar{x}_t)를 사용하여 1단계 모델링한후 1단계 모델링에 의한 과거의 10개의 예측에러를 입력벡터로 사용하여 2단계 모델링을 하였다. 1단계 모델링에 사용된 뉴럴네트워크의 출력으로는 시간 $t+1$ 의 신호 y_{t+1} 를 예측하는 신호 \hat{y}_{t+1} 로 하였다. 1단계 신경망의 입출력은 다음식에

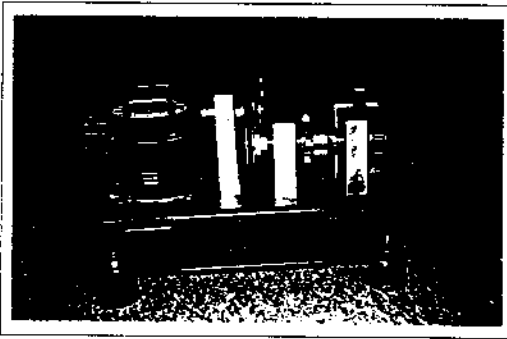


Figure 5. The Prototype Machine for Implementation

의해 표시된다. M 은 중간층의 총 노드수를 나타내며 이는 가능한 전체 입력의 Cluster수를 의미한다. 본 실험에서는 M 은 Off-line으로 기존의 Clustering 알고리즘[6]을 사용하여 15개로 결정하였다. 1단계와 2단계의 모델링은 이전의 그림 4.1과 그림 4.2에 설명하고 있으며 이를 수식으로 각단계별 입출력을 표현하면 다음과 같다.

$$\hat{y}_{t+1} = \sum_{j=1}^M w_j f_j(\bar{x}_t)$$

$\bar{x}(t) = (y_t, y_{t-1}, y_{t-2}, \dots, y_{t-9})$ 는 1단계 모델링의 입력이고 \hat{y}_{t+1} 는 출력이다.

동일한 방법으로 2단계 신경망의 입력벡터 $\bar{e}(t) = (e_t, e_{t-1}, e_{t-2}, \dots, e_{t-9})$ 에 대한 출력 \hat{e}_{t+1} 은 다음과 같다.

$$\hat{e}_{t+1} = \sum_{j=1}^M w'_j f'_j(\bar{e}_t)$$

여기서 \hat{e}_{t+1} 는 시간 $t+1$ 의 1단계 예측에러 e_{t+1} 를 다시 예측하는 값이다. 본 논문의 관점중의 하나는 1단계 예측에러들이 1단계와 2단계 모델링을 통하여 전체모델링의 예측에러를 줄일수 있는 가이다. 즉 1단계 모델링

에 사용된 신경망을 사용하여 생성된 에러들을 이용하여 모델 정밀화 과정인 2단계 신경망을 통하여 다시 재모델링 하여 전체적인 모델링의 에러를 줄이는데 있다. 그러므로 시간 $t+1$ 에서의 전체적인 모델링의 에러 \hat{e}_{t+1} 는 2단계 모델링 에러로 표시된다.

$$\hat{e}_{t+1} = e_{t+1} - \hat{e}_{t+1}$$

제안된 2단계 모델링을 평가하기 위하여 모의 실험적인 Data와 그림 5에서 보여준 모형기계에서 얻은 실제 Data를 이용하여 모델링에러를 분석하였다. 실험에 사용된 모의 Data는 AR(4), ARMA(2,1) [13], 그리고 비선형 모델(Chen et. al., 1992)[4]을 이용하여 Data를 생성하였다.

각각의 경우에 대하여 모델링의 성능을 평가하기 위하여 처음 1024개의 신호들을 신경망 구조를 Off-line으로 학습하여 신경망 중간층내의 Node수와 각 Node내의 변수를 결정하였고 나머지 150개에 대하여 On-line으로 학습하면서 일단계 예측에러(One-step prediction error)를 분석하였다. 표 1은 이론적인 모델에서 모의 Data를 이용하여 모델링의 성능을 평가한 결과이다. 모델의 구조와 변수들은 다음과 같이 주어 진다.

AR(4):

$$y_t = -1.878y_{t-1} - 1.781y_{t-2} - 1.201y_{t-3} - 0.373y_{t-4} + e_t$$

ARMA(2,1):

$$y_t - 1.490y_{t-1} + 0.650y_{t-2} = e_t + 0.5e_{t-1}$$

비선형모델:

$$y_t = 0.757y_{t-1} + 0.389u_{t-1} - 0.037y_{t-1}^2 + 0.379y_{t-1}u_{t-1} + 0.068u_{t-1}^2 + 0.121y_{t-1}u_{t-1}^2 + 0.063u_{t-1}^3 - 0.739e_{t-1} - 0.368u_{t-1}e_{t-1} + e_t$$

위 모델에서 u_t 와 e_t 는 정규분포 $N(0,1)$ 를 따른다고 가정하여 실험 Data를 생성하였다. 표 1.1-표 1.3은 1024개의 학습Data중 마지막 150개의 신호에대한 1 단계 예측에러(One-step prediction error)에 대한분석이며 Test Data는 학습 Data와의 새로운 신호 150개에 대한 1 단계 예측에러에 대한 평균과 분산, MAD 그리고 SSE(sum of square errors)값이다. 여기서 사용된 MAD는 예측에러의 절대값에 대한 평균이다. 표 1.1의 Box-Jenkins방법으로 이론적인 모델구조와 모델변수를 사용하여 1단계 예측에러를 계산하여 예측에러에 대한 평균, 분산, MAD, 그리고 SSE를 Training Data와 Test Data에 대하여 계산한 것을 보여 주고 있다. 표 1.2는 2단계 모델링의 첫번째 단계인 RHNN에 의한 1단계 모델링을 동일한 Data를 사용하여 역시 1단계 예측에러에 대한 분석을 보여 주고 있고 표 1.3은 RHNN의 2단계모델링에 대한 예측에러의 분석을 보여준다. 표 1.1 - 표 1.3은 제안된 RHNN에 의한 2단계 모델링 방법이 Box-Jenkins방법과 동등한 정확도를 보였주고 있다. RHNN에 의한 1단계 모델링보다 RHNN에 의한 2단계 모델링의 정확도가 뛰어나 제안된 2단계모델링의 정확성을 보여주고 있다.

표 1.2와 표 1.3에서 보여준 결과는 1단계 전체모델링을 통하여 모델링의 정확도는 떨어지나 2단계 세부모델링에서 3가지의 서로 다른 모델에 대하여 향상된것은 2단계 모델링방법의 타당성을 보여 준다.

실제 Data를 사용한 경우는 우선 Box-Jenkins 방법에 의한 ARMA의 모델링 방법을 이용 모델의 차수와 변수를 구한것이 표 2에 나타나 있다. 모델의 차수는 여러번의 반복적인 방법에 의하여 기존하는 SAS(Statistical Analysis System Inc.) 소프트웨어를 사용하여 가능한 가장 모델링이 정확하도록 Off-line분석에 의하여 구하였다. 여기서 제시되는 모델의 차수가 최적성은 보이기 어려운 문제이나 여러가지 반복적인 실험방법에 의하여 구하여 얻은 것임을 밝힌다. Box-Jenkins의 모델링 방법은 많은 시간과 노력이 들며 또한 데이터의 특성을 분석할수 있는 전문가가 필요하다는 점이 단점이며 반면에 RHNN의 모델링 방법이 비교적 쉬운 방법이라 할수 있다. RHNN모델링에서는 모델의 차수를 결정하는 대신 Clustering방법에 의하여 중간층의 변수를 결정하고 이를 이용하여 계속적으로 신경망 연결부를 적응적으로 수정하여 나가는 방법을 이용하고 있는 것이다.

Table 1.1 Statistical Analysis of One-step Prediction Errors Box-Jenkins Method

Model Structure	Training Data				Test Data			
	Mean	Standard-Deviation	MAD	SSE	Mean	Standard-Deviation	MAD	SSE
AR(4)	-0.0028	0.9892	0.635	150.1	-0.0056	1.0520	0.754	170.2
ARMA(2,1)	-0.3085	0.9623	1.124	304.2	-0.3164	1.4764	1.152	308.2
Nonlinear	0.0532	1.5837	2.419	1321.4	0.6450	1.8443	2.963	1421.1

Table 1.2 Statistical Analysis of One-step Prediction Errors 1-Phase RHNN Modeling

Model Structure	Training Data				Test Data			
	Mean	Standard-Deviation	MAD	SSE	Mean	Standard-Deviation	MAD	SSE
AR(4)	0.0131	0.9921	0.712	172.5	0.1235	1.0421	0.823	185.1
ARMA(2,1)	0.4472	1.1212	1.511	321.3	-0.3621	1.6331	1.836	401.1
Nonlinear	0.0724	1.9211	2.732	1432.2	0.7260	1.9342	3.001	1501.2

Table 1.3 Statistical Analysis of One-step Prediction Errors 2-Phase RHNN Modeling

Model Structure	Training Data				Test Data			
	Mean	Standard-Deviation	MAD	SSE	Mean	Standard-Deviation	MAD	SSE
AR(4)	-0.0112	0.9432	0.687	158.2	0.1056	1.0419	0.712	178.3
ARMA(2,1)	-0.3133	0.9812	1.171	296.2	-0.3421	1.4821	1.412	302.1
Nonlinear	0.0467	1.4911	2.476	1123.3	0.3185	1.9231	2.912	1321.5

전문가의 많은 노력과 시간이 필요한 시스템 모델링을 비교적 Black Box Approach에 의한 손쉽게 할수 있다는 점에서 큰 장점이 있는 것이다.

모델링의 정확도 척도인 일단계 예측에러의 평균과 분산, 그리고 MAD(Mean Absolute Deviation)의 결과가 표 3에 수록되어 있다. 표 3은 RHNN(Radial Basis Hybrid Neural Network)의 모델링에 의한 예측에러의 통계적분석이 Box-Jenkins의 ARMA 모델링에 의한 예측에러보다 적은 것을 보여주고 있다. 이는 기존의 ARMA의 모델링의 방법보다 제시한 RHNN이 정확도에 있어 우수함을 나타낸다고 볼수 있다.

그림 6과 그림 7은 기계의 정상상태에서의 ARMA의 예측과 RHNN의 2단계 모델링에 의한 1단계 예측에러를 시간별로 나타나 있다. 그림에서 보여 주듯이 RHNN은신호의 변화

를 Box-Jenkins방법보다 잘 추적하고 있다. 외륜에 이상이 있는 경우의 각방법의 예측결과가 그림 8과 그림 9에 나타나 있다. 외륜에 이상이 있는 경우에는 정상상태보다 모델링의 정확도가 감소하나 RHNN의 방법이 더 우수한것으로 보여준다.

그림 10은 모의로 이론치 Data AR(4)를 사용하여 생성시킨 Data를 제시된 2단계 RHNN과 이론적인 모델구조와 변수를 사용하여 1단계 예측에러에 대한 그림이다. 정확한 이론적인 모델 AR(4)를 사용한 경우가 예측에러가 RHNN 예측에러보다 적은것으로 보여 졌다. 이는 AR(4)모델이 비교적 에러에 영향이 적고 이론적으로 정확하게 표현되기 때문이다. 그러나 그림 11과 그림 12에서는 AR(4)보다 복잡한 모델인 ARMA(2,1)과 비선형 모델에서는 2단계 RHNN의 모델링의 1단계 예측에러가 비슷하거나 더 작게 나타

Table 2. ARMA Prediction Model

Machine Condition	ARMA Model Structure and Parameters (B denotes Backshift Operator)
Normal Condition	$(1-0.91B^{**}(1)-0.59B^{**}(2)+0.84B^{**}(3)-0.15B^{**}(4)) y_t$ $= (1-0.861B^{**}(1)-0.711B^{**}(2)+0.777B^{**}(3)) e_t$
Outer Race	$(1 - 0.14094B^{**}(1) - 0.7498B^{**}(2) - 0.10917B^{**}(3)) y_t$ $= (1 - 0.21667B^{**}(1) - 0.77942B^{**}(2)) e_t$
Inner Race	$(1 + 0.218B^{**}(1)-0.522B^{**}(2)-0.0335B^{**}(3)) y_t$ $= (1 + 0.19239B^{**}(1) - 0.58471B^{**}(2)) e_t$
Ball	$(1-0.135B^{**}(1)-0.804B^{**}(2)-0.060B^{**}(3)) y_t$ $= (1-0.143B^{**}(1)-0.853B^{**}(2)) e_t$
Looseness	$(1-0.298B^{**}(1)-0.537B^{**}(2)+0.075B^{**}(3)) y_t$ $= (1-0.183B^{**}(1)-0.50931B^{**}(2)) e_t$
Unbalance	$(1+0.27B^{**}(1)-0.99B^{**}(2)-0.30B^{**}(3))+0.13B^{**}(4)) y_t$ $= (1+0.345B^{**}(1)-0.911B^{**}(2)-0.355B^{**}(3)) e_t$

Example: $(1 + 0.2B^{**}(1) + 1.4B^{**}(2))y_t = y_t + 0.2y_{t-1} + 1.4y_{t-2}$

Model order and parameters are obtained by PC-SAS(Statistical Analysis System Inc.) ver 6.4 using iterative procedure.

Table 3. Statistical Analysis of One-step Prediction Errors

Machine Condition	Test Data					
	RHNN Modeling			ARMA Modeling		
	Mean	Standard-Deviation	MAD	Mean	Standard-Deviation	MAD
Normal	0.0021	0.0292	0.141	-0.0056	0.0520	0.153
Outer Race	-0.0185	0.8123	0.533	0.0764	5.4764	1.309
Inner Race	0.01328	0.5837	0.419	0.1450	3.3443	1.019
Ball	-0.0621	1.3994	0.674	0.1527	8.0023	1.655
Looseness	-0.0105	0.0975	0.207	0.0497	0.5607	0.471
Unbalance	-0.0171	0.0404	0.127	-0.0179	0.2800	0.314

나 모델링의 정확도가 복잡한 모델일수록 뛰 어남을 보였다. 그림 13-그림 15는 RHNN의 단계별 모델링의 정확도를 비교하였다. 모델 정밀화 과정인 2단계 모델링을 거친후에는 전체 모델링과정인 1단계 모델링의 예측에러

와 비교할때 점차 줄어들고 있음을 보여 주 었다.

변화탐지과정의 성능을 평가하기위하여 각 각의 이상상태의 1024개의 신호를 오프라인 으로 받아 두개의 서로 다른 시스템을 연결

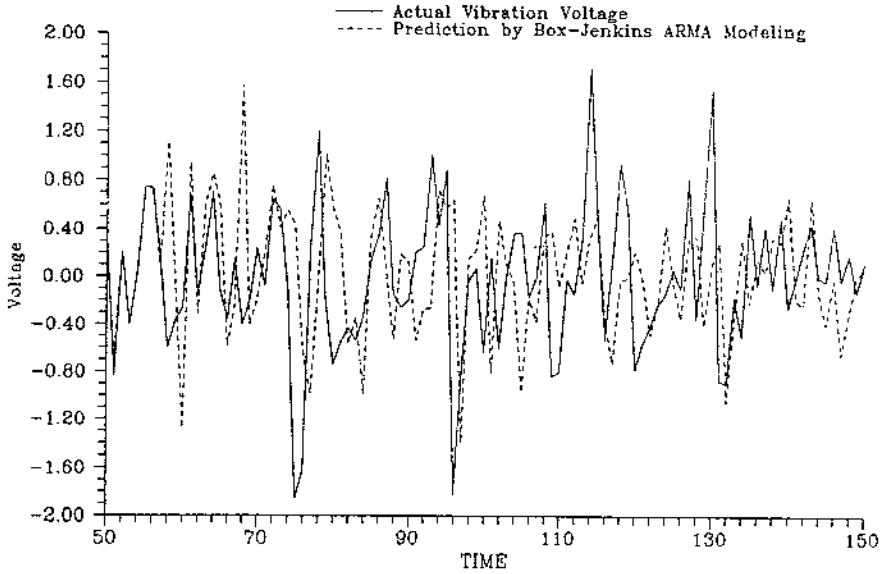


Figure 6. BOX-JENKINS ONE-STEP PREDICTION (Normal Condition)

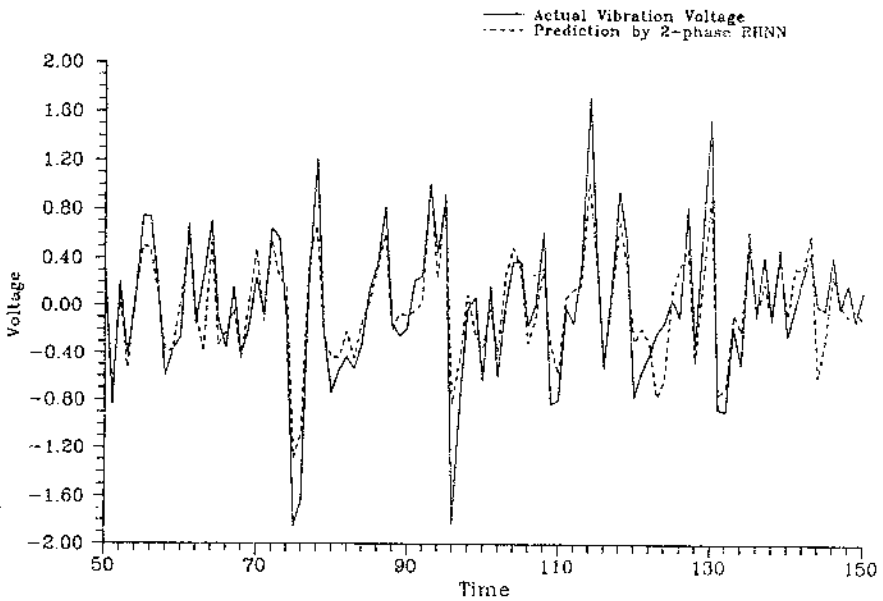


Figure 7. ONE-STEP PREDICTION BY TWO-PHASE RHNN (Normal Condition)

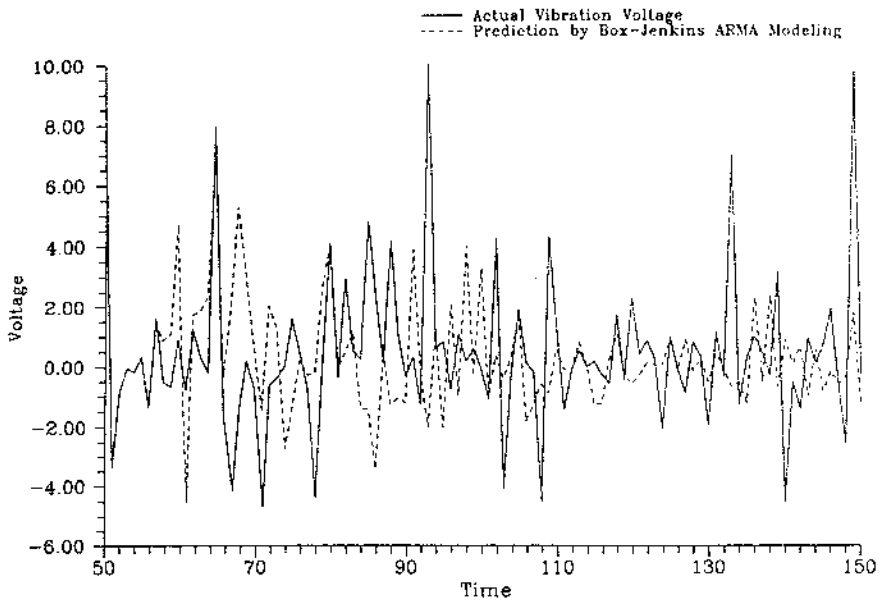


Figure 8. BOX-JENKINS ONE-STEP PREDICTION (Abnormal Condition of Outer Race)

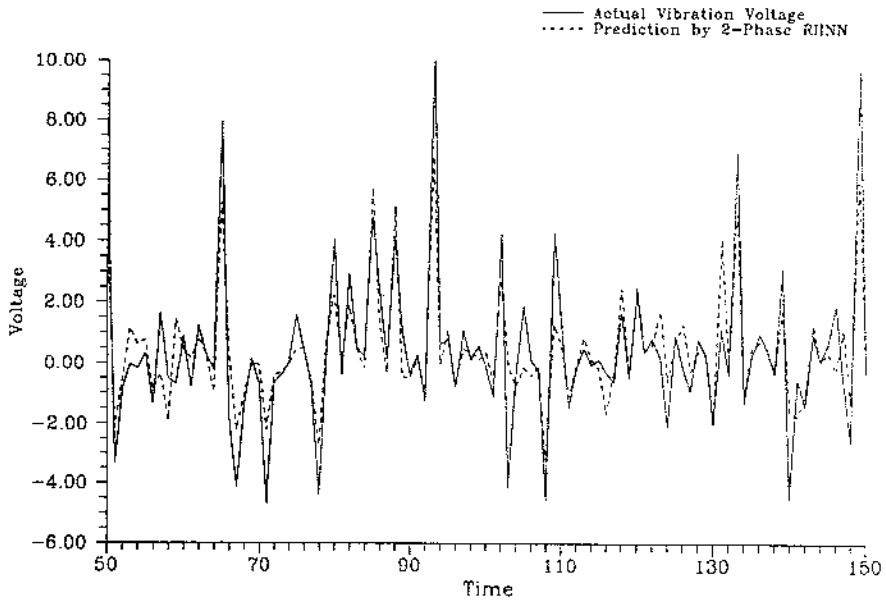


Figure 9. ONE-STEP PREDICTION BY TWO-PHASE RHNN (Abnormal Condition of Outer Race)

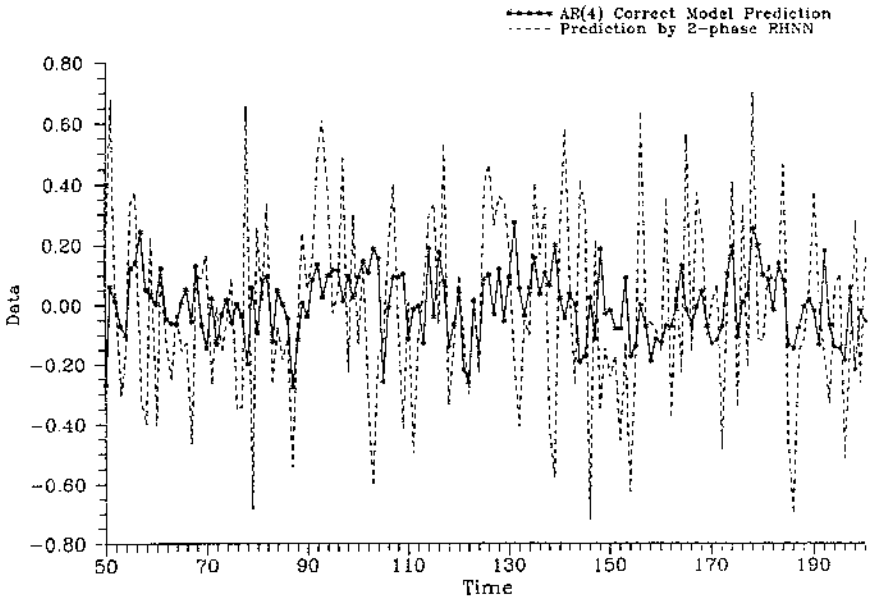


Figure 10. ONE-STEP PREDICTION ERRORS (AR(4) Simulated Series)

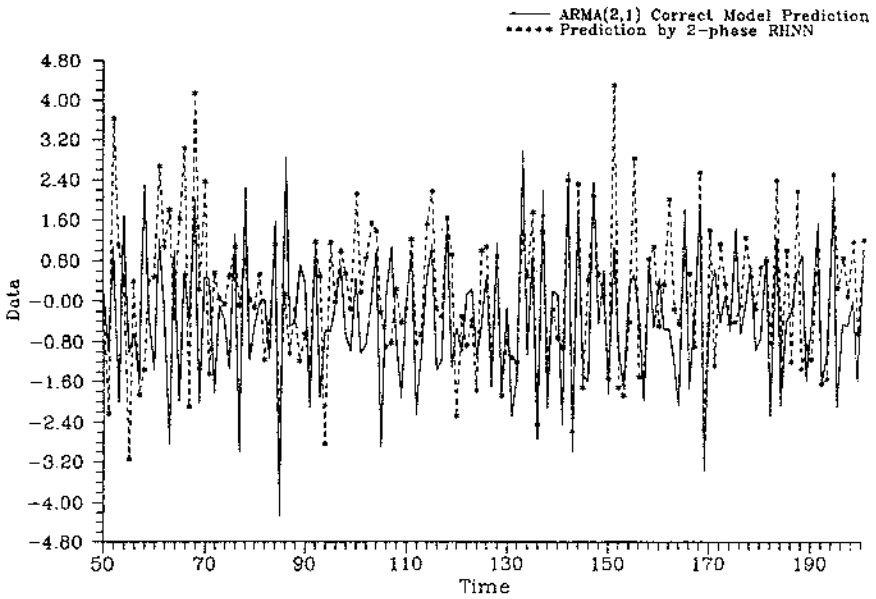


Figure 11. ONE-STEP PREDICTION ERRORS (ARMA(2,1) Simulated Series)

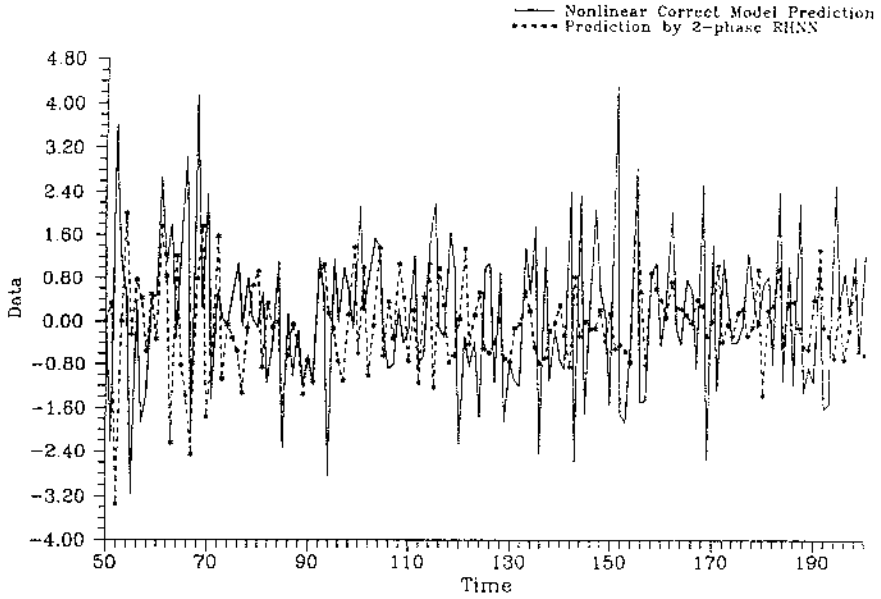


Figure 12. ONE-STEP PREDICTION ERRORS (Nonlinear Simulated Series)

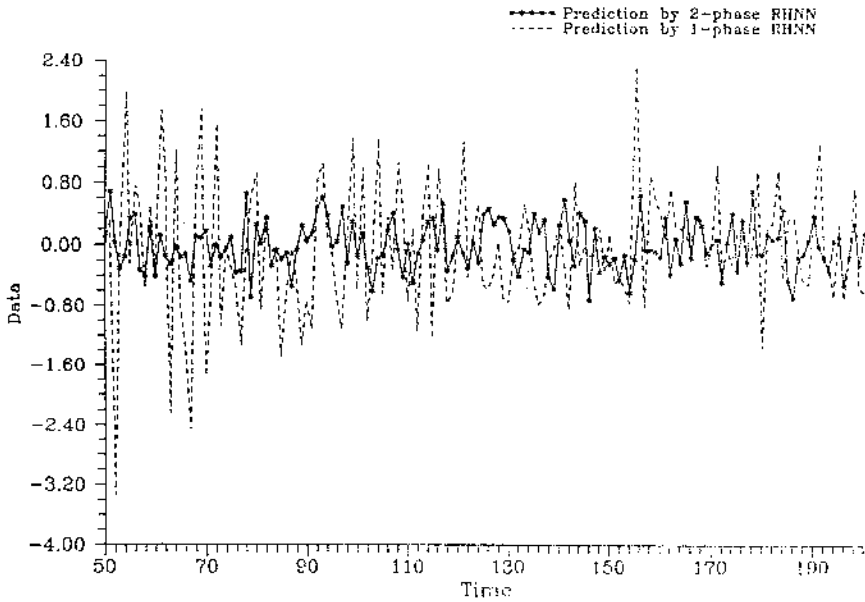


Figure 13. ONE-STEP PREDICTION ERRORS (AR(4) Simulated Series)

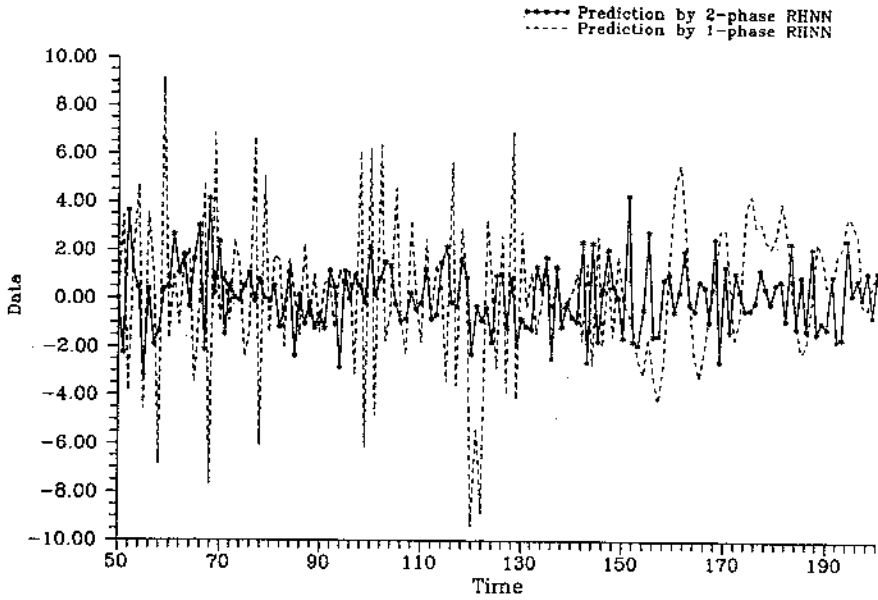


Figure 14. ONE-STEP PREDICTION ERRORS (ARMA(2,1) Simulated Series)

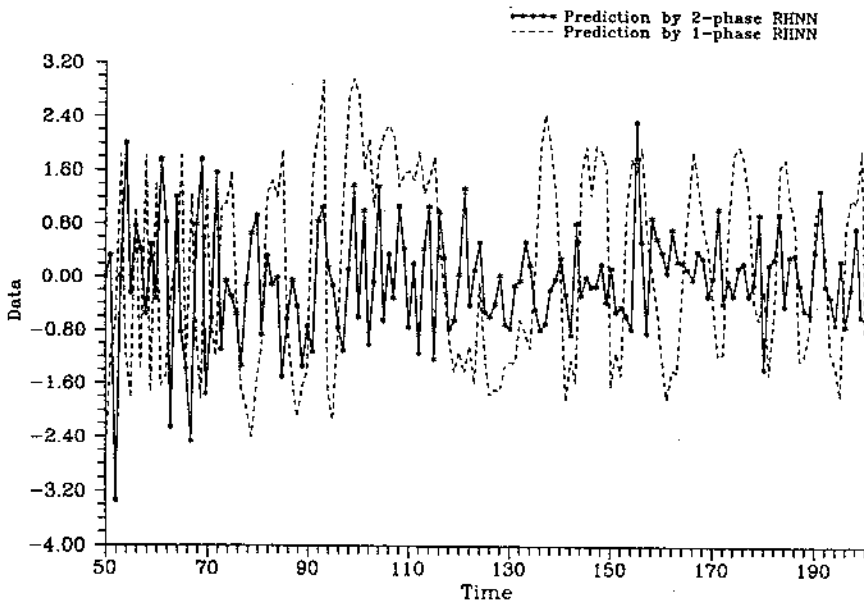


Figure 15. ONE-STEP PREDICTION ERRORS (Nonlinear Simulated Series)

Table 4. Detection Performance

Machine Condition Change	Detection Time	Change Time	Detection Delay
Normal→ Outer Race	1026	1025	1
Normal→ Inner Race	1027	1025	2
Normal→ Ball	1025	1025	0
Normal→ Looseness	1027	1025	2
Normal→ Unbalance	1028	1025	3

한 2042개의 신호가 수록된 데이터 화일을 입력신호로 이용하여 평가하였다. 평가의 방법은 1025시간에서 변화가 있을때 얼마나 빠르게 변화를 탐지하는가와 변화가 없을때 허위 탐지가 얼마나 적은가에 대한 평가이다. 변화가 없을때 변화로 인식하는 허위 변화탐지율은 0%이었고 변화탐지의 시간은 변화후 5개의 신호샘플 이내에 변화를 탐지 하였다. 표 4는 실제 데이터를 이용한 변화 탐지과정을 보여 준다.

5. 결론

본 연구는 기계설비상태의 이상여부를 탐지하고 진단하는데 필요한 기본적인 신경망에 의한 온라인 모델링방법과 변화탐지과정을 제시하였다. 제시한 RHNN은 온라인 모델링 방법보다 우수함을 보였다. 또한 모델링 방법을 이용하여 변화탐지과정을 개발하였고 제시한 탐지과정은 효율적으로 시스템 변화를 탐지 하였다. 본 결과는 기계설비의 감시 및 진단시스템의 개발에 직접적으로 응용가능하며 또한 그러한 시스템 구현이 연구 진

행중이다. 일반적으로 동적이고 변하는 시스템의 모델링을 실시간으로 구현하는 것은 시스템 신호들의 특성이 알수 없는 상황에서는 어려운 과제이다. 본 논문은 이러한 동적인 시스템 모델링을 실시간으로 가능하게 할수 있다는 점에서 의의가 있으며 응용범위도 각 제조공정의 모델링및 제어시스템 설계, 다단계 예측 시스템 개발및 각종 시스템 인식등에 활용될 수 있다.

본 연구의 확장과제로는 제시된 2단계 신경망 모델링은 처음 신경망 구조와 변수를 결정하는데 Off-line 학습을 사용하고 신경망의 구조와 변수가 결정되면 On-line 학습방법으로 연결부를 수정하고 있다. Off-line 학습과정에서 중간층의 Node수와 Radial 함수의 변수들을 처음 학습데이터를 근거로 결정하고 그후에 계속해서 입력되는 온라인 학습데이터는 신경망 연결부만 수정하고 구조는 바뀌지 않는 방법에 의존하고 있는것이다. 신경망 구조를 결정할때에 보다 전체 입력환경에 적응적이고 전체 입력신호를 잘 나타내려면 온라인 학습에 의한 신경망 구조결정이 필요할 것이다. 2단계 모델링이 확장된 다단계 모델링에 대한 평가와 단계수를 결정하는 방법도 또 다른 확장과제이다.

참고 문헌

- [1] Astrom, K. J., and Wittenmark, B., *Computer-Controlled Systems Theory and Design*, Englewood-Cliffs, Prentice Hall, New Jersey, 1984.
- [2] Elanayar, S., and Shin, Y. C., "Tool Wear Estimation in Turning Operations based

- on Radial Basis Functions," in *Dagli, C. H., S. R. T. Kumara, and Y. C. Shin (Eds.). Intelligent Engineering Systems Through Artificial Neural Networks (Proceedings of the Artificial Neural Networks in Engineering)*, pp. 685-691, November 10-13, 1991.
- [3] Carpenter, Gail A., and Grossberg, Stephen, "ART 2: Self-organization of Stable Category Recognition Codes for Analog Input Patterns," *Applied Optics*, Vol. 26, No. 23, pp 4919-4930, December 1987.
- [4] Chen, C.F.N., Billings, S.A, and Grant P. M., "A Parallel Recursive Prediction error Algorithm for Training Layered Neural Networks," *Int. J. Control*, Vol. 51, pp. 1215-1228, 1990
- [5] Kil, R.M., and Choi, J.Y., "Time-Series Prediction Based on Global and Local Estimation", *World Congress on Neural Networks(WCNN)*, Oregon, July 11-15, Vol 4, pp 617-621, 1993.
- [6] Kohonen, T., *Self-organization and Associative Memory*, Springer-Verlag, New York, 1984.
- [7] Lapedes, A., and Farber, R., "Nonlinear Signal Processing using Neural Networks: Prediction and System Modelling," *Preprint LA-UR-87-2662*, 1987.
- [8] Lapedes, A., and Farber, R., "How Neural Nets Work," *Neural Information Processing Systems*, pp 442-456, 1988.
- [9] Ljung, L., and Soderstrom, T., *Theory and Practice of Recursive Identification*, MIT Press, Cambridge, MA., 1983.
- [10] Musavi, M.T., Hummels, D.M., and Ahmed, W., "Detection of Small Signals Using an Unsupervised Radial Basis Function Network", *World Congress on Neural Networks(WCNN)*, Oregon, July 11-15, Vol 4, pp 671-674, 1993.
- [11] Narendra, K. S., and Parthasarathy, K., "Identification and Control of Dynamical systems using Neural Networks," *I.E.E. E. Transactions on Neural Networks*, 1, pp. 4-27. 1990.
- [12] Neumerkel, D, Murray, Smith R., and Gollee, H., "Modelling Dynamic Processes with Clustered Time-Delay Neurons", *Proceeding of 1993 International Joint Conference on Neural Networks(IJCNN'93 Nagoya Japan)*, Oct. 25-29, pp 1765-1768, 1993.
- [13] Sastri, T. and Wang, G.N., "Neuro-identification of Time Series for On-line Prediction", *Modeling and Scientific Computing: Neural networks for Identification and Control of Dynamic System*, 1993.
- [14] Sastri, T. and Wang, G.N., "Adaptive Model Identification Method", *Neural Parallel and Scientific Computations*, 1993.
- [15] Sebestyen, G.S., "Pattern Recognition by an Adaptive Process of Sample Set Construction," *IRE Trans. Info. Theory*, . II-8, pp. 82-91, September 1962.
- [16] Werbos, P. J., "Beyond Regression: New

Tools for Prediction and Analysis in the Behavior Science," *Ph.D Dissertation, Harvard University, Cambridge, MA, Committee on Applied Mathematics, Nov. 1974. Also published as Report of the Cambridge Project, Cambridge, MA, 1975.*

[17] Werbos, P. J., McAvoy, T., and Su, T., *Neural Networks System Identification and Control in the Chemical Process Industries*, draft, to appear in White and Jofge (Eds.), *Handbook of Intelligent Control*, Van Nostrand, New York, 1992.