

디지털 제어기용 적응 신경망 필터의 설계 및 성능평가

김진선*, 신우철+, 홍준희**

Design and Performance Evaluation of a Neural Network based Adaptive Filter for Application of Digital Controller

Jin-Sun Kim*, Woo-cheol Shin+, Jun-hee Hong**

Abstract

This Paper describes a nonlinear adaptive noise filter using neural network for digital controller system. Back-Propagation Learning Algorithm based MLP (Multi Layer Perceptron) is used as adaptive filters. In this paper, it is assumed that the noise of primary input in the adaptive noise canceller is not the same characteristic as that of the reference input. Experimental results show that the neural network based noise canceller outperforms the linear noise canceller.

Especially to make noise cancel close to realtime, Primary input is divided by unit and each divided part is processed for very short time than all the processed data are unified to whole data.

Key Words : Multi Layer Perceptron : 다층 신경망 구조
Back-Propagation Learning Algorithm : 오차 역전파 학습 알고리즘

1. 서론

공작기계는 “고속화”, “고정밀화” 라는 두가지 요구조건에 맞추기 위해 디지털 컨트롤러의 이용이 필수적이다. 특히 제어부에 사용되는 디지털 하드웨어의 눈부신 발전에 힘입어 공작기계 컨트롤러는 제어 이외에도 센서 및 제어 신호의 아날로그 필터링을 디지털 기법을 이용하여 소프트웨어적으로 처리할 수 있는 여유를 갖게 되었고 실제 대부분의 제어프로그램에는 목적에 맞는 필터가 내장되어 있다.

공작기계가 고정밀화 되어가면서 일반적인 컨트롤러에 내장되어 있는 대역통과 필터로는 제거하기 어려운 반면 제어 및 센서 신호에 그 영향이 작아 소홀이 다루어졌던 시변(time-varying) 특성 노이즈가 상대적으로 컨트롤러의 정밀도에 큰 영향을 미치게 되었다. 따라서 현대의 공작기계 컨트롤러에 내장되는 소프트웨어 필터는 이런 시변 특성 노이즈에 대한 필터링 능력이 요구된다.

주저자, 충남대학교 기계공학과 대학원 (finalguard@hanmail.net)

주소: 305-764 대전광역시 유성구 궁동 220 충남대학교 기계공학과

+ 충남대학교 기계공학과 대학원

++ 충남대학교 기계공학과

이런 시변특성 노이즈를 제거하기 위한 적응 디지털 필터는 이미 여러 종류가 제안 되었는데 대표적으로는 Widrow가 제안한 LMS(Least Mean Square) 알고리즘을 이용한 선형구조의 FIR(Finity Impulse Response) 적응필터가 있다. LMS 알고리즘은 계산이 간단하고 쉽게 구현할 수 있고 처리하고자 하는 신호가 정적인 경우에는 우수한 성능을 보여주는 장점이 있지만 신호가 시변특성이 있는 경우에 대해서는 필터링 성능이 떨어지는 단점이 있다.

본 논문에서는 이와 같은 시변 노이즈에 대한 기존의 선형 적응 필터의 성능한계를 극복하기 위한 비선형 적응필터로 BP (Back-Propogation) 알고리즘 적용 MLP(Multi Layer Perceptron)구조의 리커런트 신경망을 이용한 적응 노이즈 필터를 제시하고 있다.

신경망에 의한 적응필터의 성능평가를 위해서 일반적으로 쓰이는 적응필터인 LMS 알고리즘 적용의 FIR 적응필터와 그 필터링결과를 상호비교하고 특성을 고찰하였다. 사용된 원신호는 음성신호를 이용하였으며, 기준이 되는 노이즈를 전송경로의 모델로 사용된 비선형 함수를 통과하여 생성된 노이즈신호를 원신호에 추가하여 실험하였다. 실험결과에 의하면 신경망을 이용한 비선형 적응필터의 노이즈제거 성능이 LMS 알고리즘을 이용한 FIR 선형적응필터보다 우수하였다.

2. 적응 노이즈 필터링 시스템

2.1 적응 노이즈 필터링 시스템의 구성도

일반적으로 적응필터는 오염된 주입력의 노이즈와 유사한 특성을 갖는 또 다른 신호를 이용하여 필터링을 한다. 본 연구에서는 노이즈를 전송경로 함수를 통과시켜 원 신호를 오염시킴으로서 전송경로를 통과하기 전과는 상호 다른 특성을 갖도록 하였다. 필터는 전송경로를 통과하기전의 노이즈를 입력으로 하여 원신호에 포함된 다른 특성의 노이즈를 제거하는 역할을 한다.

Fig. 1은 적응노이즈필터의 기본구성도이다. 원신호 $s(k)$ 에 잡음 전송경로를 통과한 노이즈 $n'(k)$ 가 섞인 신호 $x(k) (= s(k) + n'(k))$ 가 주입력 (Primary input)으로 사용되며 기준입력 (Referance input)인 노이즈 $n(k)$ 는 $n'(k)$ 와 상관관계($f(\cdot)$)가 있는 신호로서 적응필터의 입력으로 사용된다.

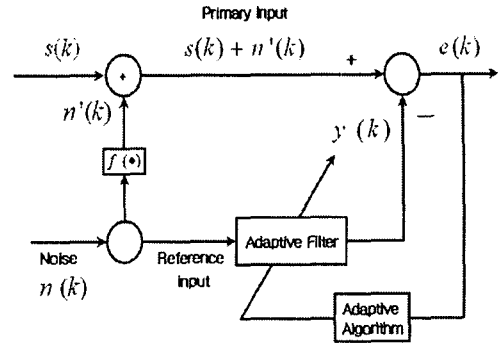


Fig. 1 Adaptive noise canceller

기본 동작구조는 시스템 출력을 적응필터로 귀환시켜 적응 알고리즘을 통해 전체 시스템의 출력전력을 최소화 하도록 필터 파라미터(신경망 시스템의 각 연결 가중치, 바이어스)를 계속 변화시킴으로써 주입력에 포함된 노이즈의 영향을 줄이는 것이다.

시스템 출력은 다음과 같은 식으로 정의된다.

$$e(k) = x(k) - y(k) = \{s(k) + n'(k)\} - y(k) \quad (1)$$

식(1)을 제공한 후 각 항의 시스템에 대한 영향치 기대값을 취하면 다음 식 (2)와 같다.

$$E[e^2(k)] = E[s^2(k)] + 2E[s(k)\{n'(k) - y(k)\}] + E[\{n'(k) - y(k)\}^2] \quad (2)$$

식 (2)의 두 번째항 $2E[s(k)\{n'(k) - y(k)\}]$ 에서 $s(k)$ 는 $n'(k)$ 및 $y(k)$ 와 uncorrelated 하므로 전체 기대치는 orthogonal property를 가지므로 "0" 이 된다. 따라서 식 (2)는 다음과 같이 정리 된다.

$$E[e^2(k)] = E[s^2(k)] + E[\{n'(k) - y(k)\}^2] \quad (3)$$

여기에서 원신호만을 추출하는 것이 목적이다. 원신호의 파워 $E[s^2(k)]$ 은 필터의 계수와는 무관하므로 필터계수를 조절하여 오차 $e(k)$ 의 파워 $E[e^2(k)]$ 를 최소화 하는

것은 식(3)의 두 번째 항 $E\{[n'(k) - y(k)]^2\}$ 을 최소화 하는 것이다. 이를 정리하면 식(4)와 같다.

$$E_{\min} [e^2(k)] = E [s^2(k)] + E_{\min} \{[n'(k) - y(k)]^2\} \quad (4)$$

식(4)에서 기대치를 제거하고 정리하면 $n'(k) - y(k)$ 가 최소 값을 가질 때 $e(k) = s(k)$ 가 된다. 따라서 필터는 선 입력 값인 $n(k)$ 값을 입력으로 하여 필터 출력 $y(k)$ 가 현재 입력 값이 $n'(k)$ 값을 추종하여 $n'(k) - y(k) = 0$ 으로 함으로써 원신호 $s(k)$ 를 추출 할 수 있다.

2.2 적응 FIR 필터

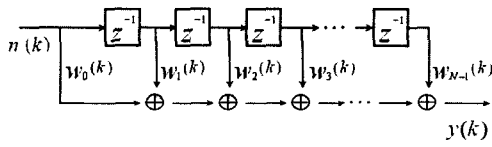


Fig. 2 Adaptive filter of FIR structure

Fig. 2는 널리 사용되는 FIR구조의 선형적응 필터이다. 위 블록선도에서 $e(k) \approx s(k)$ 가 되기위한 최적의 필터 계수는 LMS 알고리즘에 의한 시스템 출력이 최소화 하도록 필터의 임펄스 응답을 변화시키는데, 이 방법은 조정된 필터계수에 의한 필터출력과 주 입력과의 차이를 이용하여 매 샘플 당 필터 계수를 다음과 같이 계속 갱신함으로써 입력 신호에 포함된 시변 노이즈를 제거시킨다.

$$w(k+1) = w(k) + 2\mu E(k)n(k) \quad (5)$$

여기서 μ 는 수렴율을 나타낸다.

FIR 구조의 적응필터는 일반적으로 비선형성이 강하거나 시변특성이 있는 노이즈의 제거에는 성능의 저하되는 단점이 있다.

2.3 적응 신경망 필터

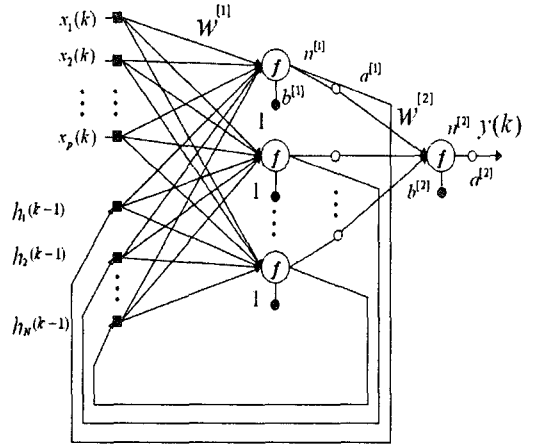


Fig. 3 Recurrent Neural Network

사용된 MLP(Multi Layer Perceptron)구조의 신경망은 은닉층의 값 전부를 입력층으로 회귀시키는 RNN(Recurrent Neural Network)의 엘만(Elman)모델⁽⁹⁾로 입력층은 문맥유닛(context unit)인 $h_n(k-1)$ 과 입력유닛(input unit)인 $x_p(k)$ 이라는 두 부분으로 나뉘어 구성되며, 문맥유닛은 은닉층의 값을 하나의 시간 단계 뒤의 입력으로 전달하는 역할을 하며 은닉유닛들의 갯수와 동일하다. 즉, 시간 k 에서의 문맥 유닛들은 순환 연결을 통해 바로 전단계의 시간 $k-1$ 에서의 은닉층 유닛들의 활성화 값을 입력으로 받아 시간 k 에 입력으로 되돌림 한다. 이처럼 문맥층은 이전 시간의 내용을 기억하는 역할을 할 뿐이지만, 과거의 신경회로망의 상태를 현재 반영할 수 있기 때문에 신경회로망이 과거의 특성을 기억하는 동적 특성을 갖게 된다. 세부적으로 입력은 $x_p(k)$ 만큼의 입력을 동시에 받는 매트릭스 X 라 한다. W 는 각 뉴런간의 가중치 매트릭스를 나타낸다. 첨자 [1]은 입력층-은닉층간, [2]는 은닉층-출력층간의 연결을 나타내며 b 는 각층의 바이어스를 나타낸다. n 은 각층의 출력층 매트릭스로 a 는 활성화 함수를 거친 각층의 출력 매트릭스를 의미한다. 은닉층과 출력층의 활성화 함수(Active Function)는 Hyperbolic Tangent Sigmoide 함수로 한다. BP 알고리즘에 의해 각층의 연결계수 매트릭스 $W^{[1]}(k)$ 와 $W^{[2]}(k)$ 의 각 성분 $w^{[1]}, w^{[2]}$ 는 다음과 같이 갱신된다.

$$\Delta w^{[1]}(k+1) = \eta \frac{\partial E(k)}{\partial w^{[1]}(k)} + \alpha \Delta w^{[1]}(k-1) \quad (6)$$

$$\Delta w^{[2]}(k+1) = \eta \frac{\partial E(k)}{\partial w^{[2]}(k)} + \alpha \Delta w^{[2]}(k-1) \quad (7)$$

여기서 η 와 α 는 각각 학습율과 모멘텀을 나타내며, 오차함수 E 는 원하는 출력과 신경망의 출력과의 차의 제곱으로 정의되며 각 연결계수는 오차함수가 최소가 되도록 생성된다. 프로그래밍을 위한 매트릭스 구조는 다음 Fig.4와 같다.

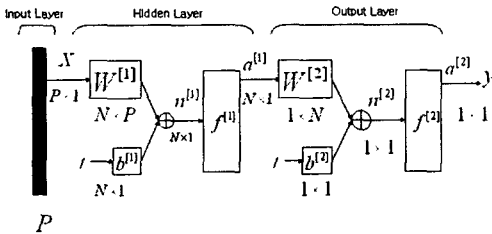


Fig. 4 Internal computation of Neural-Network filter

P-N-1의 구조로 입력층의 입력이 P개이고 X 는 $P \times 1$, 은닉층의 가중치 $W^{[1]}$ 은 $N \times P$ 의 구조, 은닉층의 바이어스와 활성화 함수를 거치기 전,후의 출력은 $N \times 1$ 의 구조를 갖고 출력층의 가중치 $W^{[2]}$ 은 $1 \times N$ 의 구조를 가지며, 마찬가지로 출력층의 바이어스와 활성화 함수를 거치기 전,후의 구조는 최종적으로 1×1 의 구조를 갖는다. 본 논문에서 사용한 MLP의 경우 필터링 전에 사전 파라미터 학습이 필요하다. 이 파라미터학습은 다음과 같은 과정으로 진행된다.

- ① 노이즈가 포함된 신호를 필터통과 시킨다.
- ② 이미 알고 있는 목적 신호 $s(k)$ 와 비교한다.
- ③ 오차 $e(k)$ 가 0이 되도록 각 가중치($W^{[1]}, W^{[2]}$)와 바이어스(b^1, b^2)를 학습 시킨다.
- ④ ③에서 각 가중치와 바이어스는 정해진 학습률에 따라 변화한다.

- ⑤ ①의 과정을 오차 $e(k)$ 가 원하는 수렴범위에 들 때 까지 반복한다.
- ⑥ 원하는 수렴범위에 들면 각 가중치와 바이어스를 고정한다.

학습이 완료되면 신경망은 노이즈 필터 전체로 보았을 때 오차 $e(k)$ 가 줄어드는 알고리즘을 내부적으로 갖게 된다. 초기의 학습율은 노이즈와 원신호가 사변해도 문맥유닛의 feedback에 의해 계속 오차가 줄어드는 알고리즘을 유지하게 된다.

3. 실험내용 및 결과

3.1 실험환경 설정

노이즈 필터의 성능을 평가하기 위한 시험신호로 실제 샘플링(sampling)된 음성신호를 사용하였다. 노이즈는 광대역의 백색노이즈(white noise)로서 이 신호를 비선형 전송 경로 함수 $f(\cdot)$ 을 거쳐서 발생된 신호를 주입력의 노이즈신호 $n'(k)$ 로 사용하였다.

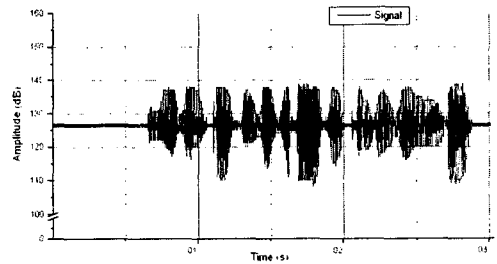


Fig. 5 Speech signal $s(k)$

Fig. 5.는 실험에 사용한 음성신호는 44.1kHz의 샘플링으로 “안녕하세요 생산실 김진선입니다.”라는 음성신호를 3초간 추출한 132300개의 데이터이다.

Fig. 6은 Fig. 5와 마찬가지로 샘플링된 가우시안 랜덤 노이즈이다. 노이즈 $n(k)$ 가 비선형 전송 경로 $f(\cdot)$ 에 의해 원신호 $s(k)$ 를 오염시킨 신호가 다음의 Fig. 7과 같다.

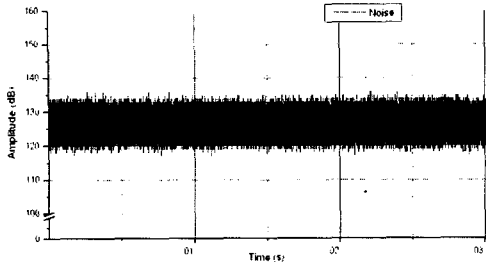


Fig. 6 Random noise (Reference Input) $n(k)$

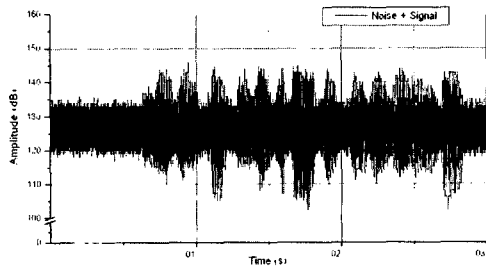


Fig. 7 Distorted speech signal (Primary Input) $s(k) + n'(k)$

Fig. 7은 원신호 $s(k)$ 와 노이즈 원으로부터 시지연 된 노이즈인 $n'(k)$ 의 합인 주입력의 파형을 나타낸다. 시지연 된 노이즈 $n'(k)$ 의 시지연은 데이터의 20샘플동안의 시간으로 0.4535ms이다. 이 때의 SNR(signal-to-noise ratio)는 3.23dB이다.

노이즈 제거 성능의 정량적인 상대비교를 위해 다음식과 같이 정의된 MSE(Mean Square Error, 평균자승오차)를 사용하며, 여기서 N은 학습 및 시험데이터의 구간을 나타내고 있다.

$$MSE = \frac{1}{N} \sum_{k=1}^N \{s(k) - e(k)\}^2 \quad (8)$$

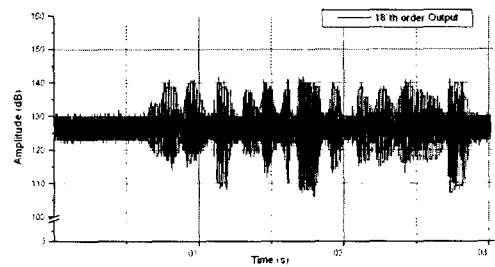
3.2 필터의 설계 및 결과

두 필터의 최종 설계변수는 다음과 같았다.

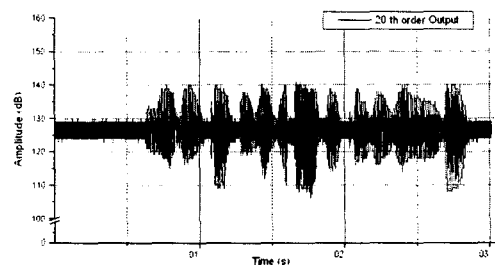
Table 1 Design Parameter

	FIR필터	신경망 필터
차수	20	-
P-N-1 노드	-	13-6-1 (입력7, 궤환6)
수렴율	0.0039	-
모멘텀율	-	0.007
학습율	-	0.1
튜닝 파라미터	21	91

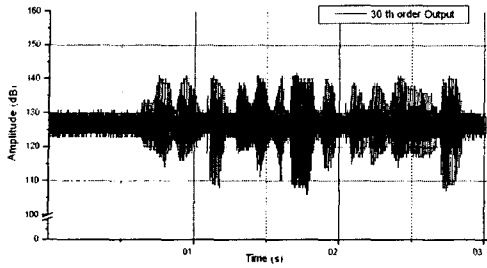
FIR필터는 반복 실험결과 학습상수를 0.01로 결정하였고 차수는 Fig. 8에서처럼 여러 차수로 FIR필터를 제작한 후 실험 결과를 비교하여 가장 우수한 성능을 갖는 차수의 필터를 선정하였다.



(a) 18'th order



(b) 20'th order

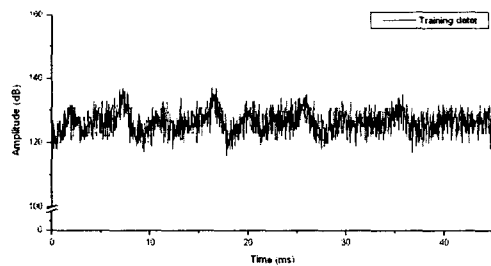


(c) 30'th order

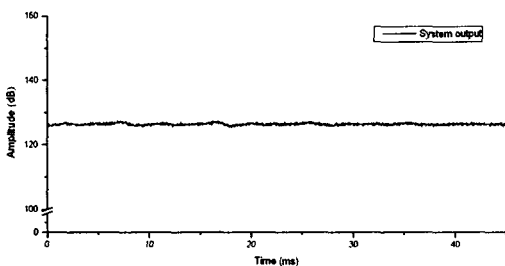
Fig. 8 System output of FIR filter

Fig. 8에서처럼 차수의 증가에 따라 필터링성능이 향상되지만 차수가 20차수 이상이 되면 도리어 성능이 저하되는 경향이보여 최적의 차수는 20으로 결정하였다. Fig. 8. (b)의 20차수 FIR필터의 SNR은 10.63 dB 이고 식 (8)에 의한 MSE는 0.0667이다.

신경망 필터의 경우 시행착오를 거쳐 P-N-1의 각 층의 노드수를 13-6-1로 선정하였으며 이 이상 입력층과 은닉층을 늘렸을 경우 FIR필터와 마찬가지로 SNR이 줄어드는 결과를 보였다.



(a) Training data



(b) System output with training data after BP training

Fig. 9 Learning data

신경망에서 BP알고리즘의 사전 파라미터학습을 위해 Fig. 9(a)와같이 1초(s)후부터 45.35ms동안의 2000샘플을 학습데이터로서 취한다. 학습데이터의 갯수가 적으면 학습이 이루어지지 않으며 많으면 계산량이 많아지기 때문에 적절한 학습데이터의 선정이 중요하며 경험 및 시행착오에 의존한다. Fig. 9.(b)는 학습을 마친 후 학습데이터에 의한 시스템 출력을 나타낸다.

전체 테스트 데이터를 입력한 경우의 시스템 출력은 Fig. 10과 같다.

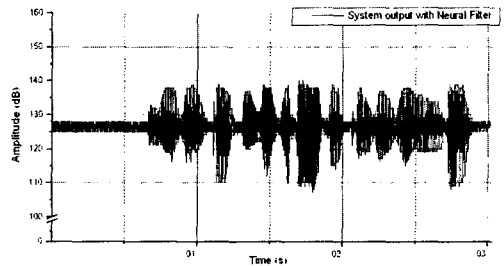


Fig. 10. System output with Neural Filter (13-6-1)

신경망 필터의 SNR은 21.11dB 이고 MSE는 0.00318이다.

3.3 성능평가

실험 결과에 따라 FIR 필터와 비교하여 신경망 필터의 성능을 평가 하면 다음 Table. 2와 같다.

Table. 2 Performance Evaluation

	FIR필터	신경망필터
MSE	0.0667	0.00318
SNR	10.63	21.11
튜닝파라메터	21	91

Table 2는 적응 노이즈필터에 LMS알고리즘 FIR 적응필터와 신경망필터를 선정하여 각각 실험한 결과이다. 성능비교를 위한 지표는 SNR과 MSE를 선택하였다. 추가로 튜닝

해야 할 파라미터의 개수로서 필터의 복잡도를 고려한다.

SNR면에서 신경망필터가 훨씬 우수한 필터링 효과를 보이고 있다. 수치상으론 두 배가 약간 넘지만 실제 Amplitude(dB)그래프 상에서는 눈에 필만큼 신경망필터의 노이즈제거 성능이 뛰어나다는 것을 알 수 있다. 이는 MSE의 비교에서도 약 21배나 우수하다는 것을 보면 알 수 있다. 그러나 프로그램화 된 디지털 필터의 계산량은 신경망필터가 4배이상 많다는 단점도 보였다.

4. 결 론

본 논문에서는 시변특성 노이즈 제거에 성능한계가 있는 선형구조의 FIR 적응필터를 대신하기 위한 비선형 적응필터로서 신경망을 사용한 적응 노이즈 필터를 제안하고 있다.

FIR필터는 LMS 알고리즘을 이용해 필터의 가중치를 튜닝하는 반면, 신경망 필터는 MLP의 구조를 BP 학습 알고리즘을 통하여 각 층의 가중치와 바이어스를 학습한다. 성능을 확인하기 위하여 비선형성이 강한 광대역노이즈가 섞인 음성신호에 대해 FIR 적응필터와 신경망 필터를 비교하여 실험하였다.

실험결과에 의하면 제안된 신경망 노이즈 필터는 비선형 노이즈 전송 경로에 대해 광대역 시변 노이즈를 효과적으로 제거하는것이 확인되었으며 기존의 LMS 알고리즘에 의한 FIR 선형 노이즈 필터에 비해 매우 우수한 필터링 성능을 보였다. 비록 필터를 구현함에 있어서 계산이 더 복잡하다는 단점도 보였으나 수렴속도나 효율면에서 볼 때 더 좋은 성능을 보여주고 있다.

참 고 문 헌

- (1) B. Widrow and S. D. Stearns, 1985 *Adaptive signal Processing* Printice-Hall, Inc., Englewood Cliffs, pp. 84~166
- (2) B. Kosko, 1992, *Neural Networks for Signal processing*. Prentice-Hall International
- (3) S. Haykin, 1991, *Adaptive Filter Theory*. Prentice-Hall, Inc., Englewood Cliffs, pp.29-48
- (4) Hagan, Demuth, Beasle, 1996 *Neural Network Design*, PWS Publishing Company.
- (5) B. Widrow 1975 "Adaptive noise cancelling : principle and applications" Proc. IEEE, Vol. 63, pp. 1692-1716
- (6) W.G.Kencht, M.E. Schenkel, 1995"Neural Network Filters for Speech Enhancement" IEEE Trans. on Speech And Audio Processing, Vol. 3, No. 6, 433
- (7) Woo-Hyuk Kwak, Won-Woo Park, Han-Go Choi 1998 Design of an Adoptive Noise Canceller using Neural Networks K.N.U.T Research Journal Vol.19, pp. 141~149
- (8) Joong-Woo Kim, Seong-Gon Kong, 2001 Speech Enhancement under Noisy Enviroment using Neuro-Fuzzy Filter
- (9) Jeffrey L. Elman, 1991 "Distributed representation, simple recurrent networks, and grammatical structure," Machine Learning, pp.195-225