

고속통신에서의 잡음제거용 적응필터의 수렴성능 분석 및 설계

조삼호*, 권승탁, 서광석**

서남대학교 정보통신학과*, 수학과**

전화 : (0671) 620-0041 / 팩스 : (0671) 620-0040

Convergence Analysis and Design of Adaptive Filter for Noise Cancel over High Speed Communication System

Sam-Ho Cho*, Seung-Tag Kwon, Kwang-Suk Suh**

Dept. of Information Communication*, Dept. of Mathematic**, Seonam University

E-mail : jsh@tiger.seonam.ac.kr, stkwon@tiger.seonam.ac.kr

Abstract

Echo and near-end crosstalk(NEXT) can be generated in two-wire duplex transmission. In this paper investigates how to cancel echoes of high speed communication. A pipeline algorithm is used to remove the echoes that high speed communication. It is least mean squared(LMS) algorithm based on the relaxed look-ahead concept, it is focused on the pipelined LMS, and its performance is compared to that of the serial LMS algorithm. And we design pipelined adaptive filtering. In advanced of the hardware implementation with VHDL code the performance of pipelined LMS algorithm is verified by the computer simulation.

I. 서론

현재의 반향제거기(echo canceller)는 진화음성만의 질을 향상시키기 위하여 사용되고 있으나 고속의 전송시스템에 사용하기에는 한계가 있으므로 고속시스템의 반향신호를 제거하기 위한 기능이 추가되어야 한다. 그래서 고속 신호처리에 사용되는 파이프라인 알고리즘에 관한 연구들이 진행되고 있다. 고속 디지털 신호처리 방법에는 파이프라인 방법과 병렬처리 방법이 이용되고 있다[1-2]. 반향신호와 근단누화를 제거하기 위한 기술들은 주로 적응 알고리즘을 이용하여 적응필터를 설계하여

이용한다 [3-4]. 적응신호처리(adaptive signal processing)는 신호처리 과정 중 그 필요에 따라 시스템의 특성을 변화시키는 기능을 갖춘 신호처리를 말하며, 시스템의 특성을 변화시키는 방법을 적응 알고리즘(adaptive algorithm)이라 한다. 적응 알고리즘은 주로 수식이 간단하여 하드웨어 구현이 쉬운 최소자승 평균 알고리즘(LMS:Least Mean Square)을 이용한다[5]. 파이프라인 알고리즘은 적응 LMS알고리즘에 예측(Look-ahead)기술을 적용한 기술로서 저전력 고속 디지털 신호처리에 적합한 구조를 가진다. 예측기술은 입출력 특성이 변화하지 않는 직렬 알고리즘으로 시간 재설정(retiming)을 통하여 처리속도를 증가시키는 기술이다. 또한 이 알고리즘은 하드웨어 증가를 줄이기 위하여 완화 예측(relaxed-look-ahead) 기술을 발전시켜서 사용한다.

본 논문은 고속전이중 전송시스템에서 반향신호와 근단누화를 제거하기 위한 파이프라인 알고리즘의 성능을 평가하고 그에 맞는 적응필터를 설계한다. 본 논문의 구성은 2장 최소자승 평균 알고리즘적용 LMS의 알고리즘, 3장 파이프라인 알고리즘 및 성능분석, 4장 파이프라인 적응 및 필터설계, 5장 결론으로 구성된다.

II. 적응 알고리즘

입력신호의 사전정보를 완전히 모르는 경우 신호처리과정에서 신호처리시스템을 어떠한 기준아래에서 최적이 되도록 축차 수정해주는 기능을 갖춘 시스템이 필요하다. 신호처리 과정에서 그 필요에 따라 시스템의 특성을 변화시키는 기능을 갖춘 시스

템을 적응 필터(adaptive filter)라 하고 시스템의 특성을 변화시키는 알고리즘을 적응 알고리즘이라 한다. 본 장에서는 적응 알고리즘과 적응 필터의 구조에 관하여 설명한다.

2.1 최소자승평균 알고리즘

적응 신호처리 시스템의 구조는 그림 1.과 같이 입력신호 $x(n)$ 에 대한 응답을 $y(n)$ 이라 하면, 원하는 응답 $d(n)$ 과의 차를 이용하여 처리 시스템의 필터계수 $W_n(M)$ 을 자동적으로 조정하여 오차신호가 자승평균오차의 의미에서 최소가 되도록 하는 시스템이다. 이때 사용하는 알고리즘이 최소자승평균(LMS: Least Mean Square)알고리즘이다

최소자승 알고리즘은 상관행렬 R 을 모르는 경우 기울기만 이용하여 다음과 같이 반복계산을 다음과 같이 행하는 방법이다.

$$W(n) = W(n-1) + 2\mu e(n)X(n) \quad (1)$$

이 방법은 계수조정 파라미터(step size) μ 의 값에 반복계산을 하면 계수 값이 바뀌는 것을 볼 수 있다. 여기서, $W(n)$ 은 시간에서 필터의 계수, $e(n)$ 은 추정오차신호 $X(n)$ 은 필터 입력신호이고 μ 는 적응속도와 안정성을 결정하고 상수로 너무 큰 값으로 설정하면 추정오차가 발산하게되고 또 너무 작게 잡으면 수렴속도는 늦는 반면 수렴후 진동 폭이 작다. 일반적으로 최소자승평균알고리즘이 안정적으로 수렴하기 위한 μ 값의 상한선은 입력신호의 상관행렬의 고유치(eigenvalues)에 의해서 결정된다. 즉 식(2)을 만족하는 μ 를 설정한다. 여기서 λ_{max} 는 입력신

$$\frac{1}{\lambda_{max}} > \mu > 0 \quad (2)$$

호의 상관행렬의 고유값 중 최대값이다. 필터 응답 $y(n)$ 은 입력신호를 $x(n)$, 필터의 계수를 $W(k)(k=0,1,\dots,M)$ 로 하면 식(3)과 같다

$$y = \sum_{k=0}^{M-1} W(k)x(n-k) \quad (3)$$

$$e(n) = d(n) - y(n) \quad (4)$$

여기서 필터계수 $W(k)$ 는 시스템 특성을 규정하는 파라미터이고 출력 $y(n)$ 은 이상응답 $d(n)$ 에 접근하도록 파라메타 값을 조정한다. 파라메타 조정을 위해서 이상응답 $d(n)$ 과 실제응답 $y(n)$ 의 차를 이용한다. 여기서 오차신호 $e(n)$ 은 오차신호의 자승평균치 (MSE : mean Square error) 식(5)와 같이 가 최소가 되도록 필터의 계수를 결정한다.

$$MSE = E[e^2(n)] \quad (5)$$

여기서 $E(x)$ 는 x 의 기대치이다.

식(5)에 식(4)를 대입하면

$$\begin{aligned} MSE &= E[e^2(n)] \\ &= E[d^2(n)] - 2E[d(n)y(n)] + E[y(n)^2] \\ &= E[d^2(n)] - 2 \sum_{k=0}^{M-1} h(k)E[d(n)x(n-k)] \\ &\quad + \sum_{k=0}^{M-1} \sum_{m=0}^{M-1} h(k)h(m)E[x(n-k)x(n-m)] \end{aligned} \quad (6)$$

로 된다. 식 (6)의 1항은 이상응답의 자승평균치(전력)와 같다. 이것을 P_d 라 하면

$$P_d = E[d^2(n)] \quad (7)$$

제2항의 기대치는 이상응답과 필터 입력신호의 상호 상관함수이다.

이것을 $P_{dx}(k) (k=0, 1, \dots, M-1)$ 로 표시한다.

$$R_{dx}(k) = E[d(n)x(n-k)] \quad (8)$$

제 3항의 기대치는 입력신호 $x(n)$ 의 자기 상관함수이다. 이것을

$$R_{xx}(m-k) = E[x(n-k)x(n-m)] \quad (9)$$

로 하여 식(6)을 정리하면

$$\begin{aligned} MSE &= P_d - 2 \sum_{k=0}^{M-1} h(k)R_{dx}(k) \\ &\quad + \sum_{k=0}^{M-1} \sum_{m=0}^{M-1} h(k)h(m)R_{xx}(k-m) \end{aligned} \quad (10)$$

로 된다.

2.2 적응 필터의 구조

최적 처리하는 적응 필터의 구조는 그림1.에 일반적인 적응 필터블럭(F-BLOCK)과 가중치계산블럭(W-BLOCK)으로 구성되고 일반적인 디지털 필터와 같이 가산기, 곱셈기 지연소자들을 이용하여 하드웨어로 구성할 수 있다. 그림2.의 적응필터블럭은 입력데이터 $x(n)$ 와 적응계수 $w(n)$ 와 컨볼루션으로 FIR필터와 같은 동작을 한다. 가중치계산블럭은 최소자승평균 알고리즘을 이용하여 가중치를 계산한다. 이 적응시스템을 하드웨어로 구현해보면 하드웨어로 구성해보면 각 소자들의 지연시간 때문에 입력신호 $x(n)$ 와 적응신호가 계산된 계수 $w(n)$ 와 동시에 동작하려면 가중치계산 시간만큼의 지연이 발생한다. 따라서 이 문제를 해결하기 위한 방법이 필요하다.

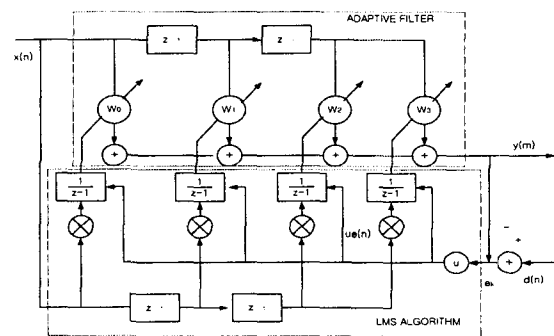


그림 1. 적응필터의 구조

III. 파이프라인 알고리즘

파이프라인은 중첩(folding) 기술을 이용하여 고속 전송과 저 전력시스템을 개발하는데 적합한 기술이다. 비파이프라인 시스템은 하드웨어 동작시간에만 동작하기 때문에 입력된 데이터 흐름이 완전히 처리될 때까지 다음의 데이터를 처리 할 수 없

다. 그러나 M 단계로 파이프라인 된 경우 비파이프라인 시스템보다 M배만큼 빨라진다[6-7]. 그러나 파이프라인 처리능력의 향상으로 인하여 하드웨어가 복잡해진다. 이를 해결하기 위하여 순환필터를 활용한 예측(look-ahead) 기술을 도입하였다. 파이프라인 알고리즘은 최소자승 알고리즘 안의 계수를 갱신하는 부분에 완화예측 (relaxed look-ahead) 기술을 적용하여 유도해 낼 수 있다. 본 장에서는 최소자승평균 알고리즘을 이용하여 파이프라인 알고리즘을 구현하는 과정을 설명한다.

3.1 완화예측기술

완화예측기술은 매우 높은 샘플 율과 전체적인 하드웨어를 최소화 할 수 있다.

$$W(n) = W(n-1) + e(n)x(n) \quad (11)$$

식 (11)의 계산시간은 하나의 더하는 시간보다 작다. 다음 시간 영역에 M-단계의 예측을 적용하면 식(12)을 얻을 수 있다.

$$W(n) = W(n-M) + \sum_{i=0}^{M-1} e(n-i)x(n-i) \quad (12)$$

이러한 변화는 순환 루프에 M래치를 포함한다. 이것은 더하는 동작의 M-단계의 파이프라이닝을 얻을 수 있도록 시간제정의 (retime)할 수 있다. 이 변화에서는 입출력 특성이 변하지 않는다. 이러한 불변성은 입출력 특성에 관하여 식(2)의 두 번째 항의 예측에 의하여 얻어진다. 완화 예측기술은 식(2)에 의하여 구도가 이루어지고 예측된 기술을 통하여 동작한다.

지연완화는 식(2)에 지연입력 $x(n-D_1)$ 과 지연계수 $e(n-D_1)$ 를 포함시켜 사용한다. 만일 $e(n)x(n)$ 에서 산출된 평균값이 D_1 샘플들보다 크거나 작으면, 식(2)은 다음과 같이 쓸 수 있다.

$$W(n) = W(n-M) + \sum_{i=0}^{M-1} e(n-D_1-i)x(n-D_1-i) \quad (13)$$

이것은 지연된 최소자승평균과 같은 결과이고 이 알고리즘은 전형적인 최소자승평균 알고리즘을 적용하였다.

일반적으로, 이것은 식(13)에 다양한 산출값 $e(n)x(n)$ 을 정지시키거나 늦게 하기 위한 것이다. 합완화(sum relaxation)의 응용은 식(13)에 LA 항은 포함하여 다음 식을 얻는다.

$$W(n) = W(n-M) + \frac{M}{LA} \sum_{i=0}^{LA-1} e(n-i)x(n-i) \quad (14)$$

이 완화는 산출되는 $x(n)$ 의 평균값이 느리게 변화한다면 정당화 될 수 있고 최소자승 평균필터를 위한 시뮬레이션을 좋은 결과를 나타낸다. 지연과 합 완화는 파이프라인 최소자승 평균 알고리즘을 수용하였다. 직렬 최소자승평균(SLMS)알고리즘 필터는 다음 방정식에 의하여 설명된다.

$$\begin{aligned} W(n) &= W(n-1) + \mu e(n)x(n) \\ e(n) &= d(n) - W^T e(n-1)x(n) \end{aligned} \quad (15)$$

D_2 는 합완화를 통한 지연이다. D_1 과 D_2 지연들은 파이프라인 하드웨어동작들에서 실제로 구현할 수 있다.

3.2 파이프라인 구조

파이프라인 알고리즘의 클럭점은 순차구조의 시간보다 적은 $\frac{1}{M}$ 시간이다. 여기서 M은 파이프라이닝의 단계를 나타낸다.

파이프라인 된 적응 필터를 동작하는 것은 순차적인 최소자승 평균식에 예측을 적용하였다. 그리고 나서, 완화 예측기법을 포함하여 파이프라인 알고리즘으로 발전시킨다.

순차 최소자승알고리즘은 식(1)에서 설명하였다.

W-BLOCK안의 D_2 지연소자를 만 $W(n)$ 드는 것은, D_2 단계의 예측으로 식(1)과 식(2)에 적용시켜 표현된 항을 $W(n-D_2)$ 로 표현한다. 식(1)을 다시쓰면

$$W(n) = W(n-D_2) + \mu \sum_{i=0}^{D_2-1} e(n-i)u(n-i) \quad (16)$$

예측을 정확하게 적용하기 위하여, $e(n)$ 은 $W(n-D_2)$ 의 함수와 같이 식(16)으로 표현하였다. 하지만, 이것은 믿을 수 없을 (incredibly) 만큼 복잡하고 하드웨어가 매우 복잡하다. 왜냐하면 식(1)과 식(2)의 쌍을 이루는 방정식이다. 따라서 식(16)은 완화를 적용한 방정식을 적용하였다.

지연은 완화는 (EFP:Error Feedback Path) 식(16)에 D_1 지연들이 포함하여 다음 식을 얻을 수 있다.

$$W(n) = W(n-D_2) + \mu \sum_{i=0}^{D_2-1} e(n-D_1-i)x(n-D_1-i) \quad (17)$$

식(16)의 두 번째 항은 과거와 현재 추정 변화도의 합이다. 완화의 유효성은 추정 변화도가 D_1 샘플들 상에서 변하지 않는다는 것을 기본으로 한다. 식(17)의 하드웨어 전체는 $N(D_2-1)$ 가산기들이다. 때문에 N과 D_2 의 값은, 전반적으로 받아들일 수 없다. 여기서 우리는 더 나아가서 식(17)에 LA항을 포함하여 합완화(sum relaxation)을 적용하여 다음 식을 얻는다.

$$W(n) = W(n-D_2) + \mu \sum_{i=0}^{LA-1} e(n-D_1-i)u(n-D_1-i) \quad (18)$$

여기서 LA는 1에서 D_2 까지 변할 수 있다. 끝으로 우리는 $W(n-1)$ 을 대신하여 식(18) 식(2)에 대입하면 다음 식을 얻는다.

$$\begin{aligned} e(n) &= d(n) - W^T(n-1)x(n) \\ &= d(n) - [W(n-D_2-1) \\ &\quad + \mu \sum_{i=0}^{LA-1} e(n-D_1-i-1)x(n-D_1-i-1)]x(n) \end{aligned} \quad (19)$$

μ 추정은 충분히 작고 $W(n-D_2)$ 에 의하여 $W(n-D_2-1)$ 에 의하여 다시 쓴다고, 우리는 식(19)을 다음과 같이 쓸 수 있다.

$$e(n) = d(n) - W^T(n-D_2)x(n) \quad (20)$$

파이프라인 최소자승평균 알고리즘은 식(18)와 (11)에 의하여 완벽하게 설명하였고 전체 하드웨어는 $N(LA-1)$ 개의 가산기들이다.

IV. 파이프라인 알고리즘 성능평가

파이프라인 알고리즘은 최소자승평균 알고리즘에 완화예측 기

술을 적용하여 구현한다. 본 장에서는 최소자승평균 알고리즘과 파이프라인 알고리즘의 오류계산과 오차의 자승평균치(MSE)에 시뮬레이션 결과에 관하여 설명한다.

4.1 적응최소자승평균 알고리즘 구현

적응최소자승평균 알고리즘의 시뮬레이션 결과는 그림2.과 같다. 그림에 나타나 있듯이 반복횟수가 증가함에 따라 오류 값이 감소하는 것을 확인할 수 있다.

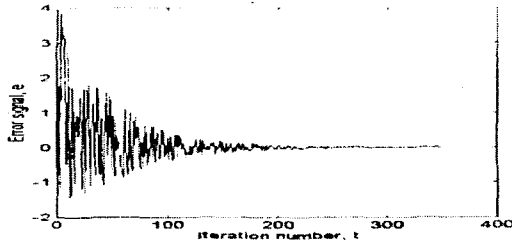


그림2. 적응최소자승평균 알고리즘 오류값

그림3은 식(2)을 이용하여 MSE를 계산한 결과로 반복횟수의 증가에 따라 MSE가 수렴함을 확인할 수 있다.

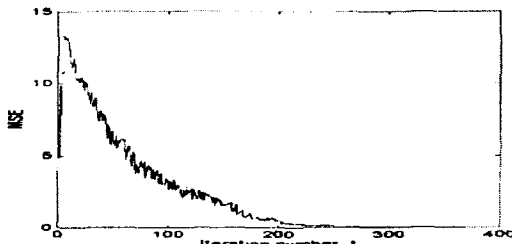


그림3. 적응최소자승평균의 오차 자승 평균치

그림4.는 직렬 적용필터와 파이프라인 알고리즘의 오차의 자승 평균치를 비교한 결과이다. 여기서 파이프라인 알고리즘은 LA=2, D₁=51, D₂=4를 갖는다. 시뮬레이션 결과를 보면 직렬최소자승 평균 알고리즘보다 지연이 되어 나타나고 있지만 수렴속도가 빠름을 볼 수 있다.

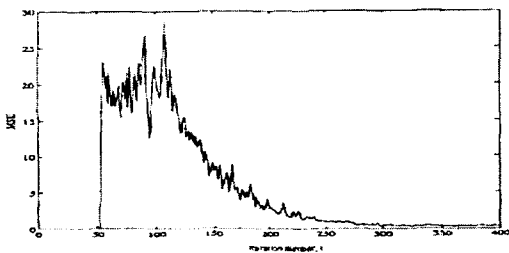


그림4. 파이프라인 알고리즘의 오차 자승 평균치

4.2 파이프라인 적응필터

파이프라인 적응필터는 FIR필터 심볼간의 간섭을 최소화하기 위하여 계수 값을 반복적으로 갱신하는 구조로 구현한다. 그림 5는 5탭을 가지는 구조로 설계하였으나 실제 적용시스템에서는 탭을 늘려서 사용해야 한다. 입력신호를 200HZ을 사용하였을

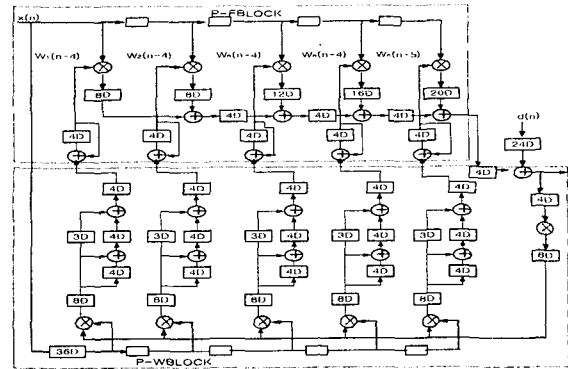


그림5 파이프라인 적응필터구조(5탭)

때 17탭이 필요하다. 그림5.은 직렬 최소자승 평균필터보다 4배의 수렴 속도를 가지는 파이프라인 적응 필터의 구조이다 여기서 램치들의 모임은 각 가산기의 곱셈기 출력이고, 실제 파이프라이닝을 구현할 때 사용한다.

V. 결론

본 논문에서는 전이중 통신을 하는 고속통신시스템의 반향이 근단누화를 제거하는 방법에 관하여 연구하였다. 적응 파이프라인 알고리즘은 고속통신시스템의 잡음을 제거하는데 필요한 기술이라 할 수 있다. 컴퓨터 시뮬레이션 결과 파이프라인 알고리즘은 직렬 최소자승평균 알고리즘 보다 처음에는 지연이 되나 빠르게 수렴하는 것을 확인하였고 필터로 구현할 경우 5 탭으로 구성한 경우 파이프라인 적응필터가 48배의 속도증가를 얻을 수 있음을 확인하였다. 이는 시간 재설정 방법을 이용하여 가능하였다.

본 논문에서 연구한 파이프라인 적응필터는 성능에 맞게 적용하면 고속모뎀이나 고속 디지털 신호처리에 유용하게 사용될 것이다.

참고문헌

- [1] K.K Parhi, "Algorithm transformation techniques for concurrent processors," *Proceeding of the IEEE*, Vol.77 PP.1879-1895, Dec, 1989
- [2] K.K Parhi and D.G. Messersschmitt, "Pipeline interleaving and parallelism in recursive digital filters-Part I: Pipelining using scattered look-ahead and decomposition," *IEEE Trans. on Acoustics, Speech and Signal Proc.*, Vol. 37 PP. 1099-1117, July 1989.
- [3] William A. thrison, Jae S. Lim, and Elliot Stinger, "A new application of adaptive noise cancellation," *IEEE Trans, ASSP-34*, No.1, Feb, PP. 21-27, 1986.
- [4] R.B. Wallace and R.A. Goubran, "Improved tracking adaptive noise canceler for nonstationary environments," *IEEE Trans, SP Vol 40*, No 3pp. 700 - 703, Mar, 1992.
- [5] N.R. Shanbhag and K.K. Parhi, *Pipelined Adaptive Digital Filters*. Boston, MA :Kluwer, 1994.
- [6] N.R. Shanbhag and K.K. Parhi, "Relaxed look-ahead pipelined LMS adaptive filters and their application to ADPCM coder," *IEEE Trans. on Circuits and Systems* Dec, 1993
- [7] G.h. Im, N.R. Shanbhag "Pipelined VLSI NEXT Cancellation for Premises Applications," *IEEE*, 1994