

수정된 AUMDF 알고리즘을 이용한

음향 반향 제거

채상훈*, 천영호*, 백홍기**

*전북대학교 공과대학 전자공학과

**전북대학교 공과대학 전자정보공학부

An Acoustic Echo Cancelling using Modified AUMDF Algorithm

S. H. Chae*, Y. H. Choen*, H. K. Baik**

Dept. of Electronic Engineering, Chonbuk Nat'l Univ.

Division of Electronic & Information Engineering Chonbuk Nat'l Univ

E-mail : chaessh@hanmail.net

요 약

일반적으로 음향 반향 제거에서 반향의 임펄스 응답이 큰 경우 주파수 영역의 알고리즘은 시간 영역 알고리즘에 비해 긴 임펄스 응답에 따른 많은 계산량과 입력신호의 통계적 특성에 의한 영향을 줄일 수 있다. 그러나 주파수 영역 알고리즘에서는 시간 영역의 신호를 주파수 영역으로 변환시킬 때 필터 차수의 2배의 FFT 연산이 필요하게 되어, 긴 차수로 인한 실행 시간 지연이 발생하고 많은 메모리가 필요하다. 이러한 문제점을 감소시키고 수렴성을 향상시키기 위한 MDF 알고리즘이 제안되었으나 계산량이 많은 단점이 있고, UMDF와 AUMDF 알고리즘은 계산량은 감소되나 수렴성이 저하되는 문제점이 있다.

본 논문에서는 기존의 MDF 알고리즘과 거의 동일한 수렴 성능을 유지하면서 연산량과 메모리를 줄일 수 있는 수정된 AUMDF 알고리즘을 제안하였으며, 모의 실험을 통해 결과를 확인하였다.

1. 서 론

원거리 회의 시스템이나 이동통신 단말기에서 핸드프리(hands-free) 통화시 발생하는 음향 반향(acoustic echo)은 기존의 유선망 반향에 비해 매우 길고 시간에 따라 빠르게 변화하는 특징을 가지고 있어 통화 품질의 저하를 초래한다. 이러한 음향 반향은 스피커와 마이크 사이에 발생하는 반향 경로(echo path)를 적응 필터로 추정하여 제거하는 방법이 일반적으로 활용되고 있다. 가장 널리 사용되는 반향 제거 알고

리즘은 NLMS 알고리즘이다. 그러나 LMS 계열 알고리즘은 음성과 같이 고유치 분포율이 큰 신호가 입력될 경우 수렴 성능이 저하되고 긴 임펄스 응답 때문에 연산량이 증가하여 실시간 처리가 어려워진다. 이러한 문제점을 해결하기 위해 주파수 영역 알고리즘을 이용하면 입력 신호의 자기 상관도가 거의 없어져 입력 신호의 통계적 특성에 의한 영향을 줄일 수 있다 [2][3]. 그리고 시간 영역에서의 콘볼루션은 주파수 영역에서 간단한 곱셈으로 이루어지므로 연산량이 감소된다. 주파수 영역 알고리즘 중 가장 기본적인 형태인 FLMS(fast least mean square) 알고리즘은 음향 제거에 사용되는 필터 탭 수가 크고 시간 영역의 신호를 주파수 영역으로 변환시킬 때 overlap-save 방법을 이용하기 때문에 필터 차수의 2배의 FFT 연산이 필요하게 된다. 따라서 긴 반향 임펄스 응답이 발생하는 경우 블록 처리에 따른 실행 시간 지연이 발생하게 되고 연산을 수행하는데 많은 메모리가 필요하다.

MDF(multidelay block frequency domain) 알고리즘은 블록 벡터의 크기를 여러 블록으로 나누어 처리함으로써 이러한 문제점을 줄이고 탭 갱신 횟수를 증가시켜 수렴 성능이 향상되지만, FLMS보다 계산량이 많아지는 단점이 있다.

UMDF(unconstrained multidelay block frequency domain) 알고리즘은 MDF 구조에서 2번의 FFT 연산과정을 생략함으로써 연산량을 감소시키지만 수렴 성능이 현저하게 낮고, 이를 보완한 AUMDF(Alternative unconstrained multidelay block frequency domain) 알고리즘도 수렴 성능이 많이 개선되지는 않는다. 본 논문에서는 기존의 MDF와 동일한 수렴 성능을 유지하고 연산량과 메모리를 줄이는 수정된 AUMDF 알고리즘을 제안하였다.

2절에서는 주파수 영역 알고리즘, 3절에서는 제안한 방법

의 모의 실험 결과를 기술하였고 4절에서 결론을 맺었다.

2. 주파수 영역 알고리즘

2.1 FLMS 알고리즘

FLMS는 직렬로 들어오는 신호를 블록 단위로 나누어 FFT를 사용하여 주파수 영역에서 탭 계수를 주파수 영역에서 갱신하는 구조이다 [3]. 주파수 영역의 출력은 IFFT를 사용하여 시간 영역으로 바꾸어서 오차 값을 구하고 여기서 구한 오차를 FFT 연산을 통해 다시 주파수 영역으로 변환하여 계수를 갱신하게 된다. 그림 1은 FLMS 알고리즘의 기본 구조를 나타낸 블록도이다.

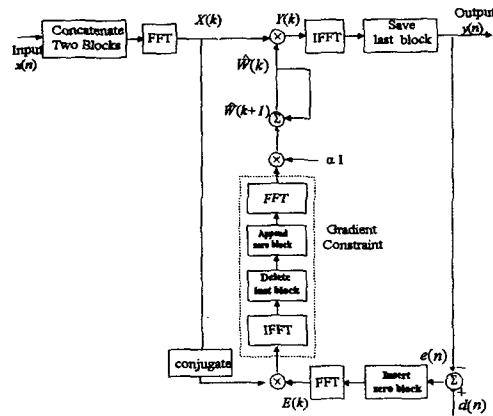


그림 1. FLMS 알고리즘의 블록도

FLMS에서는 선형 컨볼루션과 동일한 출력을 얻기 위하여 overlap-save 방법을 이용하기 때문에 FFT 차수는 시간 영역 필터 차수의 2배를 취하게 된다 [2][4]. 이로 인해 음향 반향과 같이 임펄스 응답이 긴 경우 출력의 지연이 발생하게 되어 실시간 처리가 힘들어진다. 그림 1에서 점선으로 표시된 부분은 주파수 영역으로 변환된 입력과 오차를 곱한 값을 IFFT 과정을 거쳐 마지막 블록에 0을 삽입한 후 FFT 연산을 함으로써 계수 갱신을 위한 정확한 경사 값 즉 선형 상관 값을 얻는 과정을 나타낸다 [3].

2.2 MDF 계열의 알고리즘

MDF 계열의 알고리즘으로는 UMDF, AUMDF가 있다. MDF 알고리즘은 블록의 차수를 줄이기 위해 여러 블록으로 나누는 후 지연을 두어 FFT 차수를 줄이는 알고리즘이다. MDF 알고리즘은 지연 차수에 따라 FFT 차수가 정해지는데 계수의 총 차수를 M 이라 하고 지연 블록 차수를 B 라 할 때 FFT의 차수 N 는 $2M/B$ 로 줄게 된다. 따라서 긴 반향 임펄스 응답에 따른 시간 지연 문제가 감소되고 지연 블록 차수만큼 계수

갱신 횟수가 증가되어 수렴 성능이 향상된다 [1]. 하지만 그에 따른 계산량이 증가하게 된다.

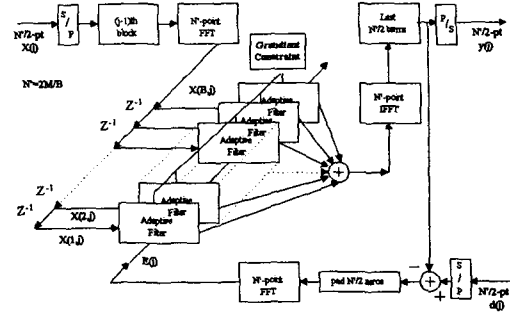


그림 2. MDF 알고리즘의 블록도

MDF에서 주파수 영역의 입력은 식 (1)과 같이 현재 블록과 과거 블록의 값을 함께 이용한다. 지연 부분의 입력 벡터는 식 (2)와 같은 방식으로 메모리에 저장된다.

$$X(B, j) = \text{diag}\{ \text{FFT}[x_0(j-1), \dots, x_{N/2-1}(j-1), x_0(j), \dots, x_{N/2-1}(j)] \} \quad (1)$$

$$X(b, j) = X(b+1, j-1), \quad b=1, 2, \dots, B-1 \quad (2)$$

주파수 영역의 출력은 식 (3)과 같이 나타난다.

$$Y(j) = \sum_{b=1}^B X(b, j) W(b, j) \quad (3)$$

시간영역의 출력 값은 식 (4)와 같이 나타내며 overlap-save 방법을 이용하기 때문에 마지막 M 요소 값만 시간 영역 출력 값으로 사용한다. 시간 영역의 오차 값은 식 (5)와 같다.

$$y(j) = \text{last } N/2 \text{ term} [\text{IFFT} \sum_{b=1}^B X(b, j) W(b, j)] \quad (4)$$

$$E(j) = \text{FFT}\{0, 0, \dots, 0, [d(j) - y(j)]^T\}^T \quad (5)$$

스텝 크기 벡터 값은 식 (6)~식 (9)에서의 같이 주파수 영역 입력 벡터의 전력 값에 의해 정규화되기 때문에 전력 스펙트럼을 평탄하게 할 수 있다 [6].

$$\mu(j) = [\mu_0(j), \mu_1(j), \dots, \mu_k(j), \dots, \mu_{N-1}(j)]^T \quad (6)$$

$$\mu_k(j) = B \mu_B / Z_k(j), \quad k=0, 1, \dots, N-1 \quad (7)$$

$$Z_k(j) = \beta Z_k(j-1) + (1-\beta) \left[\sum_{b=1}^B P_k(b, j) \right] \quad (8)$$

$$P_k(B, j) = X_k^*(B, j) X_k(B, j) \quad (9)$$

$$P_k(b, j) = P_k(b+1, j-1), \quad b=1, 2, \dots, B-1 \quad (10)$$

탭 계수 벡터 갱신을 하기 위해 식 (11)과 같이 상호 상관 벡터 값을 구하고 그림 2와 같이 전체 지연 블록에서 gradient constraint 과정을 거치게 되며 식 (12)와 같이 계수 벡터 값이 갱신된다.

$$\phi(b, j) = \text{IFFT}[\mu(j) X^*(b, j) E(j)] \quad (11)$$

$$\hat{W}(b, j+1) = \hat{W}(b, j) + \alpha \text{FFT} \left[\begin{matrix} \phi(b, j) \\ 0 \end{matrix} \right] \quad (12)$$

UMDF 알고리즘은 그림 2에서 gradient constraint 과정을 생략한 구조이다. 따라서 MDF보다 2번의 FFT 연산이 줄기 때문에 연산량은 감소하지만 정확한 선형 상관 값을 추정하지 못하므로 수렴 성능이 현저하게 낮아진다. UMDF 알고리즘의 계수 갱신은 식 (13)과 같이 나타난다.

$$\hat{W}(k+1) = \hat{W}(k) + \alpha X^*(k) E(k) \quad (13)$$

AUMDF 알고리즘은 이를 보완하기 위해 지연 블록 중 하나의 블록만 gradient constraint 과정을 수행하는데 수렴 성능이 많이 향상되지는 않는다. gradient constraint 과정을 수행하는 블록의 계수 갱신은 식 (12)와 같이 이루어지며, 나머지 블록의 계수 갱신은 식 (13)과 같이 행해진다.

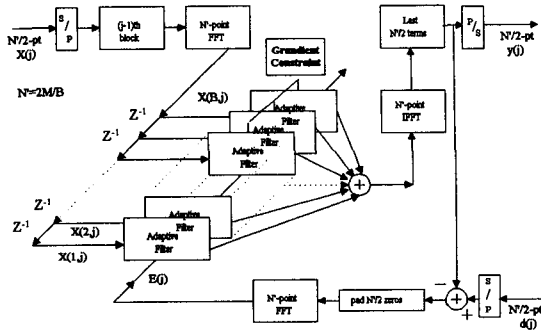


그림 3. 수정된 AUMDF 알고리즘의 블록도

2.3 수정된 AUMDF 알고리즘

본 논문에서 제안한 수정된 AUMDF 알고리즘은 기본 구조는 기존의 AUMDF와 유사하나 gradient constraint 과정을 여러 블록에 걸쳐 수행하며 계수 갱신 식은 식 (12)와 같다. 나머지 블록들의 계수 갱신 시에는 식 (13)에서와 같이 고정된 스텝 크기 값을 사용하지 않고 식 (6)~식 (10)과 같이 입력 벡터의 전력 값에 의해 정규화된 스텝 벡터를 사용함으로써 수렴 성능을 향상시켰다. unconstrained 블록의 계수 갱신 식은 식 (14)와 같다.

$$\hat{W}(k+1) = \hat{W}(k) + \mu(j) X^*(k) E(k) \quad (14)$$

표 1에는 알고리즘에 따른 곱셈 연산량을 비교하였다. 여기서 M 은 필터 차수, B 는 지연 차수를, B_c 는 constraint 블록 수를 나타낸다. 시간 영역 NLMS 알고리즘과 비교하였을 때 필터 차수가 높을수록 MDF와 수정된 AUMDF 알고리즘이 연산량이 감소함을 알 수가 있다. 수정된 AUMDF 알고리즘은 전체 지연 블록 중 50%가 constraint 블록일 때 연산량이 MDF에 비해 30%정도 감소된다.

표 2에는 알고리즘 수행에 필요한 메모리 포인트 수를 나타내는데 FLMS에 비해 MDF 알고리즘이 현저하게 감소되며

수정된 AUMDF 알고리즘이 가장 적은 메모리를 필요로 한다.

표 1. 알고리즘에 따른 곱셈 연산량

알고리즘	단위출력당 곱셈연산량
NLMS	$3M$
FLMS	$10 \log_2 M + 8$
MDF	$[(4B+6) \log_2 M + 8B - (4B+6) \log_2 B]$
수정된 AUMDF	$[(6+4B_c) \log_2 M + 8B - (6+4B_c) \log_2 B]$

표 2. 알고리즘에 따른 메모리 포인트 수 비교

알고리즘	메모리 포인트 수
FLMS	$14M$
MDF	$6M + 8M/B$
수정된 AUMDF	$4M + 8M/B + 2MB_c/B$

3. 모의 실험 및 결과

본 절에서는 입력 신호로 유색 잡음과 실제 음성 신호를 사용하여 NLMS 알고리즘과 MDF, 수정된 AUMDF 알고리즘의 성능을 비교하였다. MDF 알고리즘은 계산량 대비 수렴 성능이 가장 좋은 8차의 지연 차수를 사용하였다 [1]. 수정된 AUMDF 알고리즘도 MDF와 같은 8차의 지연 차수를 사용하고 constraint 블록 수가 4개일 때의 성능을 비교하였다. 유색 잡음은 백색 가우시안 잡음을 식 (15)와 같은 AR 모델을 통과시켜 분산이 1이 되도록 조정하여 얻었다.

$$A(z) = \frac{0.1}{1 - 0.9z^{-1}} \quad (15)$$

반향 경로는 512개의 샘플을 가지는 실측한 임펄스 응답을 사용하였고 이상 응답 신호는 신호 대 잡음비가 30dB가 되도록 백색 가우시안 잡음을 첨가하여 얻었다. 성능 평가는 유색 잡음은 식 (16)과 같은 MSE(mean square error) 곡선으로, 음성 신호는 식 (17)과 같은 ERLE(echo return loss enhancement) 곡선으로 평가를 하였다.

$$\text{MSE}(n) = 10 \log [E[e^2(n)]] \quad (16)$$

$$\text{ERLE}(n) = 10 \log \left[\frac{\sum_{i=0}^{K-1} [d^2(n-i)]}{\sum_{i=0}^{K-1} [e^2(n-i)]} \right] \quad (17)$$

그림 4은 수정된 AUMDF에서 constraint 블록 수 B_c 가 전

체 8개의 블록 중 4개인 경우를 기존의 AUMDF와 비교한 것으로 AUMDF보다 성능이 우수함을 알 수 있다. 그림 5는 NLMS와 MDF, 수정된 AUMDF 알고리즘의 MSE 곡선을 비교한 것으로 수정된 AUMDF가 MDF와 거의 동일한 수렴 성능을 나타내었고 NLMS보다 수렴 성능이 우수하였다. 그림 6은 입력이 음성 신호일 경우의 ERLE 곡선을 나타낸 것으로 MSE 곡선에서와 같이 수정된 AUMDF가 MDF와 거의 동일한 성능을 나타냄을 알 수 있었다.

4. 결론

본 논문에서는 시간 영역이 아닌 주파수 영역 알고리즘인 AUMDF 알고리즘을 음향 반향 제거에 도입하였다. 수정된 AUMDF 알고리즘은 시간 영역의 NLMS 알고리즘과 주파수 영역의 FLMS 알고리즘보다 수렴 성능이 우수하고, FLMS 알고리즘에서 문제가 되었던 출력 시간 지연을 감소시키는 등, MDF 알고리즘의 장점을 그대로 유지함을 알 수 있었다. 그리고 계산량과 알고리즘을 실행하는 데 필요한 메모리를 감소시킴으로써 하드웨어의 부담을 줄이고 긴 차수의 음향 반향이 발생하는 경우에도 실시간 처리가 가능해진다.

참고문헌

- [1] J. S. Soo and K. K. Chang "Multidelay block frequency domain adaptive filter," *IEEE Trans. Signal Processing*, vol. 38, no. 2, pp. Feb. 1990.
- [2] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1996.
- [3] J. J. Shynk. "Frequency-domain and multirate adaptive filtering," *IEEE Signal Processing Mag.*, vol. 9, no. 1, pp. 14-37, Jan. 1992.
- [4] E. R. Fera, "Fast implementation of LMS adaptive filter," *IEEE Trans. Signal Processing*, vol. 28, no. 6, pp. 464-475, June 1980.
- [5] G. A. Clark, S. R. Parker, "A unified Approach time-domain and frequency-domain realization of FIR adaptive digital filters," *IEEE Trans. Signal Processing*, vol. 31, no. 5, pp. 2075-2087, Oct. 1993.
- [6] J. C. LEE and C. K. Un "Performance of transform-domain LMS adaptive digital filter," *IEEE Trans. Signal Processing*, vol. 34, no. 6, pp. 499-510, June 1986.

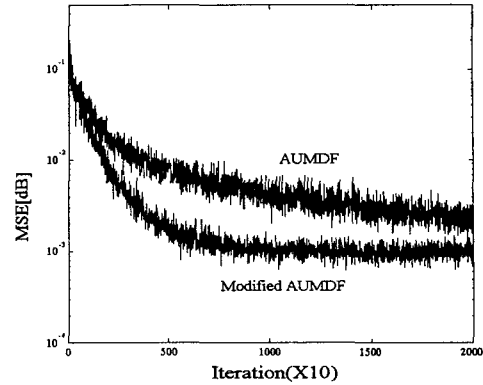


그림 4. AUMDF와 수정된 AUMDF의 MSE 곡선

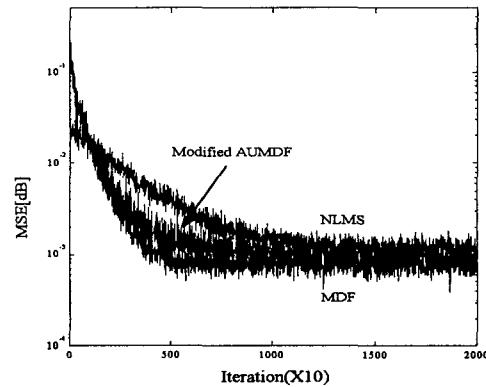


그림 5. NLMS와 MDF, 수정된 AUMDF의 MSE 곡선

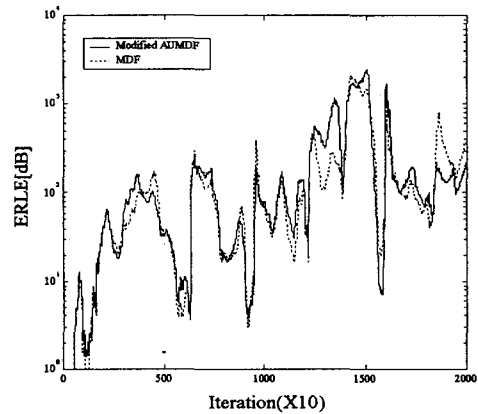


그림 6. MDF와 수정된 AUMDF의 ERLE 곡선