

기울기 평균 벡터를 사용한 가변 스텝 최소 평균 사승을 사용한 음향 채널 추정기

An acoustic channel estimation using least mean fourth with an average gradient vector and a self-adjusted step size

임준석[†]

(Jun-Seok Lim^{1†})

¹세종대학교 전자정보통신공학과

(Received March 9, 2018; revised April 16, 2018; accepted May 30, 2018)

초 록: LMF(Least Mean Fourth) 알고리즘은 특히 비정규 잡음 상황에서 안정성 및 빠른 수렴성을 나타낼 뿐만 아니라 추정 오차도 낮은 것으로 잘 알려져 있다. 최근 LMS (Least Mean Square) 알고리즘 분야에서는 가변 스텝 크기를 적용한 알고리즘들에 대한 관심이 증대되어 왔다. 그 이유는 가변 스텝 크기 LMS가 다양한 환경에서 고정 스텝 크기 LMS보다 우수한 결과를 내기 때문이다. 본 논문에선 LMF에 대한 가변 스텝 크기의 한 방법으로 기울기 평균 벡터를 사용한 가변 스텝 크기를 사용하는 LMF 알고리즘을 제안한다. 제안된 방법은 가변 스텝 크기 LMS와 마찬가지로 고정 스텝 크기 LMF보다 우수할 것이 예상된다. 본 논문은 그 우수성을 시불변 채널과 시변 채널 각각의 채널 환경 하에서 시뮬레이션을 통하여 보인다.

핵심용어: 음향통신, LMF (Least Mean Fourth), 음향 채널 추정기, 가변 스텝 조절

ABSTRACT: The LMF (Least Mean Fourth) algorithm is well known for its fast convergence and low steady-state error especially in non-Gaussian noise environments. Recently, there has been increasing interest in the LMS (Least Mean Square) algorithms with self-adjusted step size. It is because the self-adjusted step-size LMS algorithms have shown to outperform the conventional fixed step-size LMS in the various situations. In this paper, a self-adjusted step-size LMF algorithm is proposed, which adopts an averaged gradient based step size as a self-adjusted step size. It is expected that the proposed algorithm also outperforms the conventional fixed step-size LMF. The superiority of the proposed algorithm is confirmed by the simulations in the time invariant and time variant channels.

Keywords: Acoustic communication, LMF (Least Mean Fourth), Acoustic channel estimation, Self-adjusted step-size

PACS numbers: 43.60.Dh, 43.60.Mn

1. 서 론

음향 통신과 음향 신호처리에서 필터를 설계하여 잡음을 동반한 데이터로부터 유용한 정보를 추출하고 있다. 이를 위한 필터 최적화 문제를 푸는 해결책 중 하나가 1949에 발표된 Wiener 필터 설계 방법이다.^[1] 그러나 Wiener 필터 설계는 처리될 신호들이 정

재 상태이어야 하고 또한 이들 신호들의 통계적 특성을 충분히 이용할 수 있어야 한다. 더욱이 최적 필터 계수를 구하기 위해 일명 Wiener-Hopf 방정식이라고 불리는 많은 계산량을 필요로 하는 선형 행렬 방정식을 풀어야 한다.^[1,2] 그러나 적응 필터는 신호의 통계적 특성을 완전하게 알지 못하더라도 최적의 필터 계수에 근접할 수 있다. 즉, 적절하게 선택된 통계적 기준에 따라 계속해서 최적 필터 계수가 조정되고 입력 신호와 목표 신호 사이의 오차로부터 점

[†]Corresponding author: Jun-Seok Lim (jslim@sejong.ac.kr)
Department of Electrical Engineering, Sejong University, 209,
Neungdong-ro, Guang-jin-gu, Seoul 05006, Republic of Korea
(Tel: 82-2-3408-3299, Fax: 82-2-3408-4329)

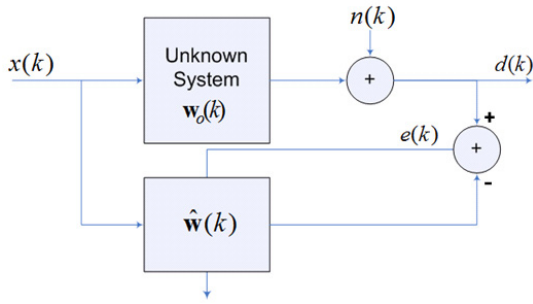


Fig. 1. A structure of adaptive filtering system.

차적으로 학습하면서 최적 필터에 근접해 간다. 이런 특징을 이용하고자 많은 연구자들이 적응 필터 방법 연구에 관심을 집중하여 왔다. 현재 적응 알고리즘은 시스템 인식, 잡음 제거, 에코 상쇄, 채널 등화 등 다양한 분야에 응용되고 있다.^[2] 디지털 적응 필터 시스템은 Fig. 1과 같은 블록 다이어그램으로 나타낼 수 있다. 즉 필터 출력은 입력 신호를 이용하여 목표 신호를 적응적으로 추정하는 문제로 생각할 수 있다.

Fig. 1로부터 오차 신호는 다음 Eq. (1)과 같다.

$$e(k) = d(k) - y(k) = d(k) - \hat{\mathbf{w}}^H(k)\mathbf{x}(k), \quad (1)$$

여기서 $\mathbf{x}(k)$ 는 입력신호 $x(k)$ 의 현재부터 $N-1$ 번째까지 N 개의 샘플을 사용한 벡터이고, $\mathbf{x}(k) = [x(k), x(k-1), \dots, x(k-N+1)]^T$, $\hat{\mathbf{w}}(k)$ 은 차수가 N 인 적응 디지털 필터 계수 벡터이며, $\hat{\mathbf{w}}(k) = [\hat{w}_0(k), \hat{w}_1(k), \dots, \hat{w}_{N-1}(k)]^T$ 는 전치 행렬을 나타낸다. 그 외에 부가 잡음 $n(k)$ 는 입력에 통계적으로 독립이고, 평균 '0'을 중심으로 대칭 분포를 이룬다고 가정한다.

Fig. 1과 같은 구조의 적응 필터를 구하는 방법들 중에 하나인 최소평균자승(Least Mean Square, LMS) 알고리즘은 실제 값과 추정 값 사이의 오차의 에너지 평균, $E(e(k)^2)$,를 최소화 하는 변수를 추정한다. 또 다른 방법으로는 오차의 네제곱, 즉 $E(e(k)^4)$,를 최소화하는 경우도 있다.^[3] 이를 최소평균사승(Least Mean Fourth, LMF) 알고리즘이라고 한다.^[3] LMS와 LMF 모두 최소화하는 변수를 얻기 위해서 급경사법(steepest descent)을 사용한다. 알려진 바와 같이 급경사법을 위해서는 추정 변수 값을 갱신하는 속도를

조절하기 위해서 스텝 크기를 사용한다.

LMF는 측정 신호가 비정규 분포 부가 잡음에 노출되었을 때 LMS 방법보다 우수하다고 알려져 있다. 특히 수렴 후 결과를 보면 LMF가 LMS보다 더 정확한 추정 결과를 생산한다.^[4] 그리고 LMS와 마찬가지로 LMF 알고리즘의 수렴 속도와 오차 성능이 스텝의 크기에 따라서 달라진다고 알려져 있다.^[2] 즉, 빠른 수렴 속도를 위해서는 큰 크기의 스텝이 필요하나 정확도가 떨어지고, 정확도 향상을 위해서 작은 크기의 스텝을 택한 경우 수렴 속도가 떨어진다.^[2] 그러나 수렴 속도와 정확도 둘 다를 모두 만족하는 하나의 스텝 크기를 고정적으로 정하는 것은 불가능하다. 이와 관련하여 스텝 크기를 상황에 따라 변화시켜서 수렴 성능 향상을 꾀하는 가변 스텝 크기 알고리즘이 LMS 알고리즘과 관련하여 많은 결과가 발표되었다.^[5-8] 그러나 이에 반하여 LMF 분야에서는 가변 스텝 크기에 관한 연구가 상대적으로 적다.^[9]

본 논문에서는 LMF를 위한 새로운 가변 스텝 크기 알고리즘을 제안한다. 제안하는 방법은 Malenovsky^[10] 등이 LMS 알고리즘을 위해서 제안한 지수함수적으로 평균화된 기울기 벡터를 사용하는 가변 스텝 사이즈 계산법을 LMF 알고리즘에 맞게 수정하여 제안한다.

제안된 알고리즘의 성능을 보이기 위해서 기존 LMF 알고리즘과 함께 시불변 시스템과 시변 시스템에 대한 시스템 인식 실험을 하고 그 수렴 성능을 비교한다.

II. LMF

본장에서는 LMF 알고리즘을 요약한다. LMF는 오차의 네제곱 즉 $E(e(k)^4)$ 를 최소화하는 하는 것이라고 서론에서 기술한 바 있다. 단 LMF에서는 오차의 네제곱 값의 평균 $E(e(k)^4)$ 을 사용하는 대신, 오차의 네제곱 순시값 $e(k)^4$ 를 사용한다. 좀 더 구체적 말하면 다음과 같다.^[2,3]

Fig. 1에서와 같이 $\hat{\mathbf{w}}^T(k) = [\hat{w}_0(k), \hat{w}_1(k), \dots, \hat{w}_{N-1}(k)]$ 과 $\mathbf{w}_o^T(k) = [w_0(k), w_1(k), \dots, w_{N-1}(k)]$ 를 각각 추정 시스템 계수 벡터와 참 시스템 계수 벡터라고 하

자. 시간 k 에서 순시 오차는 다음 식과 같다.

$$\begin{aligned} e(k) &= d(k) - \hat{\mathbf{w}}^H(k) \mathbf{x}(k) \\ &= \mathbf{w}_0^H(k) \mathbf{u}(k) + n(k) - \hat{\mathbf{w}}^H(k) \mathbf{x}(k), \\ &= n(k) - \mathbf{c}^H(k) \mathbf{x}(k), \end{aligned} \quad (2)$$

여기서 $\mathbf{c}(k) = \hat{\mathbf{w}}(k) - \mathbf{w}_0(k)$. 순시 오차 $e(k)$ 의 4제곱 한 후 $\hat{\mathbf{w}}(k)$ 에 대한 미분을 적용하면 다음과 같은 식을 얻을 수 있다. H는 허미트 전치(hermitian transpose) 행렬이다.

$$\nabla(e(k)^4) = -4e(k)^3 \mathbf{x}(k). \quad (3)$$

위 순시 미분 값을 사용하여 다음과 같은 추정 시스템 계수 갱신 식이 유도된다.

$$\begin{aligned} \hat{\mathbf{w}}(k+1) &= \hat{\mathbf{w}}(k) - \hat{\mu} \nabla(e(k)^4) \\ &= \hat{\mathbf{w}}(k) + \hat{\mu} 4e(k)^3 \mathbf{x}(k), \\ &= \hat{\mathbf{w}}(k) + \mu e(k)^3 \mathbf{x}(k), \end{aligned} \quad (4)$$

여기서 μ 는 스텝 크기다.

III. 평균 기울기 벡터를 사용한 가변 스텝 크기 LMF

가변 스텝 크기를 얻기 위해서 Eq. (5)와 같은 목적 함수를 고려한다.

$$J = \frac{1}{4} \sum_{k=1}^n \lambda^{n-k} e(k)^4, \quad (5)$$

여기서 $e(k) = d(k) - \mathbf{w}^H \mathbf{x}(k)$ 이다. 그리고 Eq. (5)를 이용한 추정 계수 벡터 갱신 식은 다음 Eq. (6)과 같다.

$$\begin{aligned} \mathbf{w}(n) &= \mathbf{w}(n-1) + \mu_n \nabla_{\mathbf{w}} J|_{\mathbf{w}=\mathbf{w}(n-1)} \\ &= \mathbf{w}(n-1) + \mu_n \mathbf{g}_{n|n-1}, \end{aligned} \quad (6)$$

그리고 목적함수의 미분을 좀 더 자세히 풀면 Eq. (7)과 같음을 알 수 있다.

$$\begin{aligned} \nabla_{\mathbf{w}} J &= \sum_{k=1}^n \lambda^{n-k} e(k)^3 (-\mathbf{x}(k)) \\ &= \sum_{k=1}^n \lambda^{n-k} e(k)^2 (-\mathbf{x}(k)) (d(k)^* - \mathbf{x}^H(k) \mathbf{w}) \\ &= (\lambda \mathbf{R}_{n-1} + e(n)^2 \mathbf{x}(n) \mathbf{x}^H(n)) \mathbf{w} \\ &\quad - (\lambda \mathbf{r}_{n-1} + e(n)^2 \mathbf{x}(n) d(n)^*) \\ &= \mathbf{R}_n \mathbf{w} - \mathbf{r}_n \end{aligned} \quad (7)$$

여기서 행렬 \mathbf{R}_n 에 대한 갱신식을 다음과 다시 쓸 수 있다.

$$\begin{aligned} \mathbf{R}_n &= \lambda \mathbf{R}_{n-1} + e(n)^2 \mathbf{x}(n) \mathbf{x}^H(n). \\ &= \lambda \mathbf{R}_{n-1} + (e(n) \mathbf{x}(n)) (e(n) \mathbf{x}(n))^H. \end{aligned} \quad (8)$$

Eq. (8)에서 보면 행렬 \mathbf{R}_n 은 새로운 행렬 $e(n) \mathbf{x}(n)$ 의 지수함수 가중 자기 상관 행렬이라고 할 수 있다. 그리고 벡터 \mathbf{r}_n 에 대한 갱신식도 다음과 다시 쓸 수 있다.

$$\begin{aligned} \mathbf{r}_n &= \lambda \mathbf{r}_{n-1} + e(n)^2 \mathbf{x}(n) d(n)^*. \\ &= \lambda \mathbf{r}_{n-1} + (e(n) \mathbf{x}(n)) (e(n) d(n))^*. \end{aligned} \quad (9)$$

Eq. (9)에서 보면 벡터 \mathbf{r}_n 는 행렬 $e(n) \mathbf{x}(n)$ 과 스칼라 양 $(e(n) d(n))$ 사이의 지수함수 가중 상호 상관 벡터임을 알 수 있다. 그리고 Eq. (7) 목적 함수의 순시 기울기에 대해서 일종의 이동 평균을 취하는 형식을 가진다. 이로부터 순간순간 변동성을 줄이는 효과를 낸다. 그리고 Eq. (7)을 정리하여 다음과 같은 기울기 평균 벡터 갱신식을 얻는다.

$$\nabla_{\mathbf{w}} J|_{\mathbf{w}=\mathbf{w}(n-1)} = \mathbf{R}_n \mathbf{w}(n-1) - \mathbf{r}_n = \mathbf{g}_{n|n-1}. \quad (10)$$

또 가변 스텝 크기는 다음과 목적함수를 스텝 사이즈에 대해서 최소화할 함으로써 다음과 같이 얻을 수 있다.

$$\nabla_{\mu} J = 0, \quad (11)$$

위 미분 시에 나오는 상수는 무의미하여 생략하고, $\mathbf{w}(n) = \mathbf{w}(n-1) + \mu_n \mathbf{g}_{n|n-1}$ 과 $e(k) = d(k) - \mathbf{w}^H(n)$

$\mathbf{x}(k)$ 도 함께 고려한다. 그 결과를 자세하게 풀어 쓰면 Eq. (12)와 같다. Eq. (12)의 각 항을 따로 묶어 정리하면 Eq. (13)이 된다. Eq. (14)는 Eq. (13)을 행렬과 벡터를 이용하여 정리한 것이다.

$$\begin{aligned} \nabla_{\mu} J &= \sum_{k=1}^n \lambda^{n-k} e(k)^2 (-\mathbf{g}_{n|n-1}^H \mathbf{x}(k)) (d(k)^* - \mathbf{x}^H(k) \mathbf{w}(n)) \\ &= \sum_{k=1}^n \lambda^{n-k} e(k)^2 (-\mathbf{g}_{n|n-1}^H \mathbf{x}(k) d(k)^* \\ &\quad + \mathbf{g}_{n|n-1}^H \mathbf{x}(k) \mathbf{x}^H(k) (\mathbf{w}(n-1) + \mu_n \mathbf{g}_{n|n-1})) \end{aligned} \quad (12)$$

$$\begin{aligned} \nabla_{\mu} J &= -\mathbf{g}_{n|n-1}^H \left\{ \sum_{k=1}^n \lambda^{n-k} e(k)^2 \mathbf{x}(k) d(k)^* \right\} \\ &\quad + \mathbf{g}_{n|n-1}^H \left\{ \sum_{k=1}^n \lambda^{n-k} e(k)^2 \mathbf{x}(k) \mathbf{x}^H(k) \right\} \mathbf{w}(n-1) \\ &\quad + \mathbf{g}_{n|n-1}^H \left\{ \sum_{k=1}^n \lambda^{n-k} e(k)^2 \mathbf{x}(k) \mathbf{x}^H(k) \right\} \mathbf{g}_{n|n-1} \mu_n. \end{aligned} \quad (13)$$

$$\begin{aligned} \nabla_{\mu} J &= -\mathbf{g}_{n|n-1}^H \mathbf{r}_n + \mathbf{g}_{n|n-1}^H \mathbf{R}_n \mathbf{w}(n-1) \\ &\quad + \mathbf{g}_{n|n-1}^H \mathbf{R}_n \mathbf{g}_{n|n-1} \mu_n. \end{aligned} \quad (14)$$

위 Eqs. (14)와 (10)으로부터 다음과 같은 가변 스텝 크기가 유도 된다.

$$\mu_n = \frac{-\mathbf{g}_{n|n-1}^H (\mathbf{R}_n \mathbf{w}(n-1) - \mathbf{r}_n)}{\mathbf{g}_{n|n-1}^H \mathbf{R}_n \mathbf{g}_{n|n-1}} = \frac{-\mathbf{g}_{n|n-1}^H \mathbf{g}_{n|n-1}}{\mathbf{g}_{n|n-1}^H \mathbf{R}_n \mathbf{g}_{n|n-1}}. \quad (15)$$

위 Eq. (5)부터 Eq. (15)까지를 정리하여 새로운 가변 스텝 크기 LMF를 정리하면 다음 Table 1과 같이

Table 1. Summary of the proposed algorithm.

$e(n) = d(n) - \mathbf{w}(n-1)^H \mathbf{x}(n)$
$\mathbf{R}_n = \lambda \mathbf{R}_{n-1} + e(n)^2 \mathbf{x}(n) \mathbf{x}^H(n)$
$\mathbf{r}_n = \lambda \mathbf{r}_{n-1} + e(n)^2 \mathbf{x}(n) d(n)^*$
$\mathbf{g}_{n n-1} = \mathbf{R}_n \mathbf{w}_{n-1} - \mathbf{r}_n$
$\mu_n = \frac{-\mathbf{g}_{n n-1}^H \mathbf{g}_{n n-1}}{\mathbf{g}_{n n-1}^H \mathbf{R}_n \mathbf{g}_{n n-1}}$
$\mathbf{w}(n) = \mathbf{w}(n-1) + \mu_n \mathbf{g}_{n n-1}$
Total number of multiplication and division: $3N^2 + 8N + 3$ (N is the dimension of $\mathbf{w}(n)$).

요약 된다.

Table 1에 요약한 알고리즘의 복잡도를 알고리즘 전체에 사용된 곱셈과 나눗셈의 전체 횟수로 할 때 Table 1의 알고리즘은 $O(N^2)$ 수준으로써 일반적인 LMF의 $O(N)$ 수준에 비해 무척 많은 계산량을 요구하는 것을 알 수 있다. 본 논문에서는 계산량을 줄이기 위한 한 예로 다음과 같은 단순화된 알고리즘을 함께 제안한다.

단순화 알고리즘을 위해서 Eqs. (8)과 (9)에 나온 \mathbf{R}_n 과 \mathbf{r}_n 에 대한 갱신식을 다음과 같이 수정한다.

$$\begin{aligned} \mathbf{R}_n \mathbf{w}(n-1) &= \lambda \mathbf{R}_{n-1} \mathbf{w}(n-1) + e(n)^2 \mathbf{x}(n) \mathbf{x}^H(n) \mathbf{w}(n-1) \\ &= \lambda \mathbf{R}_{n-1} \mathbf{w}(n-1) + e(n)^2 \mathbf{x}(n) y(n)^* \end{aligned} \quad (16)$$

여기서 $y(n) = \mathbf{w}(n-1)^H \mathbf{x}(n)$. 또 Eq. (6)에 \mathbf{R}_n 을 양변에 곱하여 다음과 같이 수정한다.

$$\mathbf{R}_n \mathbf{w}(n) = \mathbf{R}_n \mathbf{w}(n-1) + \mu_n \mathbf{R}_n \mathbf{g}_{n|n-1}. \quad (17)$$

Eqs. (16)과 (17)은 모두 벡터 갱신식이 되었다. 갱신식을 간단하게 보이기 위해서 다음과 같은 새로운 벡터를 정의한다.

$$\mathbf{R} \mathbf{w}_{pre}(n) \triangleq \mathbf{R}_n \mathbf{w}(n-1), \mathbf{R} \mathbf{w}_{post}(n) \triangleq \mathbf{R}_n \mathbf{w}(n). \quad (18)$$

Eq. (18)를 이용하여 Eqs. (16)과 (17)을 다시 쓰면 다음과 같다.

$$\mathbf{R} \mathbf{w}_{pre}(n) = \lambda \mathbf{R} \mathbf{w}_{post}(n-1) + e(n)^2 \mathbf{x}(n) y(n)^*. \quad (19)$$

$$\mathbf{R} \mathbf{w}_{post}(n) = \mathbf{R} \mathbf{w}_{pre}(n) + \mu_n \mathbf{R}_n \mathbf{g}_{n|n-1}. \quad (20)$$

그리고 \mathbf{R}_n 의 주대각 요소만으로 벡터를 만들어서 이를 $\mathbf{r}_{diag,n}$ 이라고 하자. 주대각 요소만 사용하는 이유는 자기상관 행렬에서 주대각으로부터 멀어지면 요소값이 급격히 작아지는 성질이 있기 때문이다. 특히 채널 추정이나 시스템 인식 문제를 다룰 때 입력 신호를 충분히 불규칙하게 만들어 쓰는 경우가 대부분이라서 이런 간략화 방법이 의미를 가진다고

Table 2. Simplification of the proposed algorithm.

$$\begin{aligned}
y(n) &= \mathbf{w}(n-1)^H \mathbf{x}(n) \\
e(n) &= d(n) - y(n) \\
\mathbf{Rw}_{pre}(n) &= \lambda \mathbf{Rw}_{post}(n-1) + e(n)^2 \mathbf{x}(n) \mathbf{y}(n)^* \\
\mathbf{r}_n &= \lambda \mathbf{r}_{n-1} + e(n)^2 \mathbf{x}(n) d(n)^* \\
\mathbf{g}_{n|n-1} &= \mathbf{Rw}_{pre}(n) - \mathbf{r}_n \\
\mathbf{r}_{diag,n} &= \lambda \mathbf{r}_{diag,n-1} + \mathbf{x}(n) \odot \mathbf{x}(n)^* \\
\mathbf{rg}_{diag,n} &= \mathbf{r}_{diag,n} \odot \mathbf{g}_{n|n-1} \\
\mu_n &= \frac{-\mathbf{g}_{n|n-1}^H \mathbf{g}_{n|n-1}}{\mathbf{g}_{n|n-1}^H \mathbf{rg}_{diag,n}} \\
\mathbf{w}(n) &= \mathbf{w}(n-1) + \mu_n \mathbf{g}_{n|n-1} \\
\mathbf{Rw}_{post}(n) &= \mathbf{Rw}_{pre}(n) + \mu_n \mathbf{rg}_{diag,n}
\end{aligned}$$

Total number of multiplication and division: $11N+7$ (N is the dimension of $\mathbf{w}(n)$).

할 수 있다. 이때 $\mathbf{r}_{diag,n}$ 은 다음과 같이 갱신할 수 있다.

$$\mathbf{r}_{diag,n} = \lambda \mathbf{r}_{diag,n-1} + \mathbf{x}(n) \odot \mathbf{x}(n)^* \quad (21)$$

여기서 \odot 다음 식은 요소별 곱을 의미한다. 다음은 위 $\mathbf{r}_{diag,n}$ 을 사용하여 Eq. 20의 두 번째 항을 간략화하는 근사식이다.

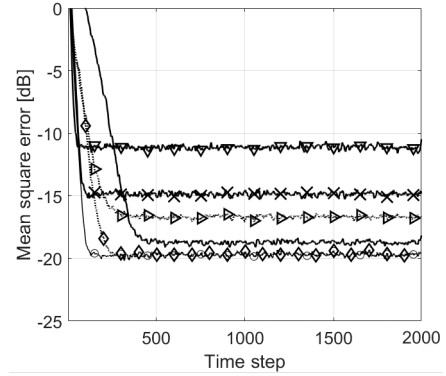
$$\mathbf{rg}_{diag,n} \triangleq \mathbf{R}_n \mathbf{g}_{n|n-1} \cong \mathbf{r}_{diag,n} \odot \mathbf{g}_{n|n-1} \quad (22)$$

위 같은 간략화 과정을 Table 1에 적용하여 간략화된 알고리즘을 구성하면 Table 2와 같다.

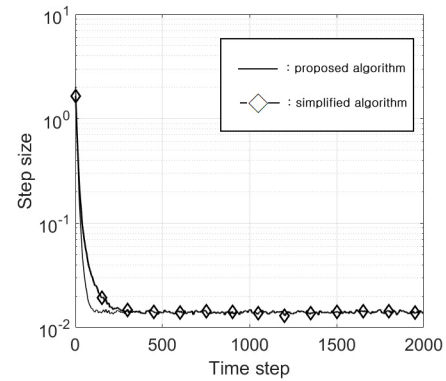
Table 2의 알고리즘의 곱셈과 나눗셈 전체 횟수는 원래 알고리즘에 대해서 훨씬 줄어들어 그 수준이 $O(N)$ 수준에 도달함을 알 수 있다.

IV. 시뮬레이션

본 장에서는 시뮬레이션을 통해서 제안한 알고리즘의 수렴 성능을 알아본다. 이를 위해서 시간에 상관없이 특성이 일정한 시불변 시스템에 대해서 일반 LMF와 3장에서 제안한 두 가지 가변 스텝 LMF간의 수렴 성능을 비교한다. 그리고 실험 중간에 채널이 갑자기 바뀌는 시변 시스템 시변 시스템에 대해서 각각 수렴 성능을 비교한다.



(a) Mean square error comparison



(b) A step-size curve for the proposed

Fig. 2. Convergence performance comparison in the time invariant channel (solid line: LMF with $\mu = 0.01$, $-x-$: LMF with $\mu = 0.05$, $-v-$: LMF with $\mu = 0.1$, $-o-$: LMS with $\mu = 0.01$, $-o-$: proposed algorithm, $-diamond-$: simplified proposed algorithm).

4.1 시불변 시스템에서 수렴 성능 비교

제안된 알고리즘의 일반적인 수렴 성능을 보이기 위해서 시불변 시스템 환경 하에서 수렴 실험을 하였다. 실험에 사용된 환경을 구체적으로 기술하면, 입력 신호는 -0.5 와 0.5 사이의 균일 분포를 이루는 불규칙 신호를 사용하였다. 또 부가 잡음도 같은 균일 분포 불규칙 잡음 사용하였다. 신호 대 잡음비는 10 dB로 하였다. 이런 신호를 사용하여 시스템의 임펄스 응답이 $\mathbf{w}_0 = [-0.3, -0.9, 0.8, -0.7, 0.6]$ 인 시불변 채널 상에서 제안된 알고리즘과 전통적인 고정 스텝 크기 LMF의 수렴 성능을 각각 비교하였다. 이때 사용한 고정 스텝 크기 값은 0.01 , 0.05 및 0.1 로 설정하였다. 그리고 LMF와 LMS 사이 성능 차이의 경향을 보이기 위하여 고정 스텝 크기 값은 0.01 인 경우도 함께 도시 하였다. 그 외에 제안한 두 알고리즘의 스텝 사

이즈 변화를 함께 도시하였다.

Fig. 2를 보면 제안한 두 가변 스텝 LMF와 세 개의 서로 다른 고정 스텝 크기 사용하는 LMF를 서로 비교하였다. 결과로부터 고정 스텝 크기 LMF의 경우 작은 크기 스텝 크기를 ($\mu = 0.01$)에서 ($\mu = 0.1$)로 크게 사용할 때는 MSE(Mean Square Error)가 작은 경우로부터 점점 커지는 것을 관찰 할 수 있다. 그리고 LMS의 경우 같은 스텝 사이즈 LMF에 비해서 나쁜 MSE를 보이고 있다. 수렴 속도는 고정 스텝 크기 ($\mu = 0.01$)를 사용할 경우 LMF는 수렴 속도가 가장 느리고 고정 스텝 크기 ($\mu = 0.1$)인 경우는 수렴 속도가 가장 빠르다. 따라서 고정 스텝 사이즈만으로는 수렴 속도와 MSE를 모두 좋게 유지할 수 없음을 알 수 있다.

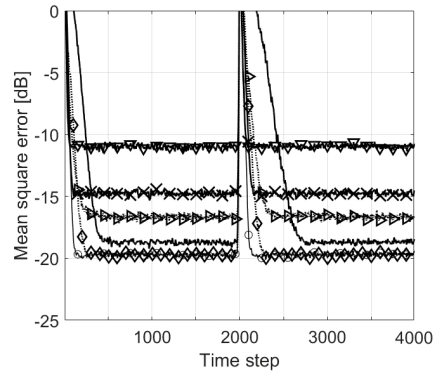
제안한 두 가변 스텝 LMF는 제안한 데로 스텝 크기를 LMF의 추정 상황에 알맞게 스스로 조절하여 수렴 초기에는 빨리 수렴하고 충분히 수렴 후에는 작은 오차를 유지하도록 하고 있음을 알 수 있다. 특히 Table 2에 요약한 간편화된 알고리즘이 Table 1에 요약한 원래 알고리즘에 비해 수렴 속도는 늦으나 수렴 경향은 거의 같은 것을 관찰 할 수 있다. 이는 간략화 과정에서 오는 영향이 존재하나 가변성을 크게 해치지 않는다는 것으로 생각할 수 있다.

4.2 급격하게 변하는 시변 시스템에서

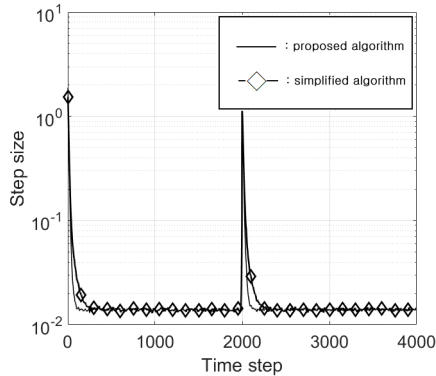
수렴 성능 비교

제안된 두 알고리즘의 성능을 보기 위한 또 다른 경우로써 급격히 변하는 시변 시스템 환경 하에서 수렴 실험을 하였다.

실험에 사용된 환경을 구체적으로 기술하면, 입력 신호와 부가 잡음은 이전과 같고, 시변 시스템을 모사하기 위해서 처음부터 2000번 샘플까지는 시스템의 임펄스 응답이 $\mathbf{w}_{01} = [-0.3, -0.9, 0.8, -0.7, 0.6]$ 이다가 2001번 샘플부터 임펄스 응답이 $\mathbf{w}_{02} = [0.6, -0.7, 0.8, -0.9, -0.3]$ 로 갑자기 변하는 시변 채널 상에서 제안된 알고리즘과 전통적인 고정 스텝 크기 LMF의 수렴 성능을 각각 비교하였다. 이때 사용한 고정 스텝 크기 값도 0.01, 0.05 및 0.1로 설정하였다. 그리고 LMF와 LMS 사이 성능의 경향을 보이기 위한 고정 스텝 크기 값은 0.01인 경우와 제안한 두 알고리즘의 스텝 사이즈 변화도 함께 도시 하였다.



(a) Mean square error comparison



(b) A step-size curve for the proposed

Fig. 3. Convergence performance comparison in the abruptly-changing channel (solid line: LMF with $\mu = 0.01$, $-\times-$: LMF with $\mu = 0.05$, $-\nabla-$: LMF with $\mu = 0.1$, $-\triangleright-$: LMS with $\mu = 0.01$, $-\circ-$: proposed algorithm, $-\diamond-$: simplified proposed algorithm).

Fig. 3을 보면 수렴 성능은 시불변의 경우와 유사하며, 제안한 가변 스텝 LMF가 2001 샘플 이전에 수렴이 되어 시불변 특성을 유지하다 갑자기 새로운 채널로 바뀌는 경우에도 변화된 채널 상황에 맞춰서 채널 추정기가 다시 신속하게 적응하고 있는 것을 알 수 있다. 시변 시스템의 경우도 Table 2에 요약한 간편화된 알고리즘이 Table 1에 요약한 원래 알고리즘에 비해 수렴 속도는 조금 늦으나 수렴 경향은 거의 같음을 확인할 수 있다.

V. 결론

본 논문에선 고정 스텝 LMF의 수렴 성능을 향상시키기 위한 한 방법으로 가변 스텝 LMF를 제안하였다. 제안한 방법은 순시 오차 기울기 대신 이동 평균 기울기를 사용한 가변 스텝을 사용하도록 만들었

다. 이를 통해서 기울기 상태에 따라 알맞은 가변 스텝 크기를 제공하여 수렴 속도를 향상시키고 채널 추정 성능의 변동을 줄이는 효과를 얻었다.

References

1. B. Farhang-Boroujeny, *Adaptive Filters: Theory and Applications, 2nd Edition* (John Wiley & Sons, Inc. Hoboken, NJ, 2013), pp. 62.
2. A. H. Sayed, *Adaptive Filters* (IEEE Press, Wiley-Interscience, New York, 2008), pp.167.
3. E. Walach and B. Widrow, "The least mean fourth (LMF) adaptive algorithm and its family," *IEEE Trans. IT.* **30**, 275-283 (1984).
4. P. I. Hubscher and J. C. Bermudez, "An improved statistical analysis of the least mean fourth (LMF) adaptive algorithm," *IEEE Trans. SP.* **51**, 664-671 (2003).
5. B. Farhang-Boroujeny, "Variable-step-size LMS algorithm: new developments and experiments," *IEE Proc. -Vis. Image Signal Process.* **141**, 311-317 (1994).
6. R. H. Kwong and E. W. Johnston, "A variable step size LMS algorithm," *IEEE Trans. SP.* **40**, 1633-1642 (1992).
7. J. S. Lim, "A constant modulus algorithm based on an orthogonal projection" (in Korean), *J. Acoust. Soc. Kr.* **28**, 640-645 (2009).
8. J. S. Lim and Y. K. Pyeon, "A variable step sized constant modulus algorithm based on an averaged gradient," *KICS.* **35**, 79-82 (2010).
9. S. M. Asad, A. Zerguine, and M. Moinuddin, "On the convergence analysis of a variable step-size LMF algorithm of the quotient form," *Proc. ICASSP 2010*, 3722-3725 (2010).
10. V. Malenovsky, Z. Smekal, and I. Koula, "Optimal step-size LMS algorithm using exponentially averaged gradient vector," *Proc. EUROCON 2005*, 1554-1557 (2005).

저자 약력

▶ 임 준 석 (Jun-Seok Lim)



1986년 서울대학교 전자공학과 학사 졸업
 1988년 서울대학교 전자공학과 석사 졸업
 1996년 서울대학교 전자공학과 박사 졸업
 1996년7월 ~ 1997년10월 LG종합기술원
 현재 세종대학교 전자정보통신공학과 교수