

적응 쌍선형 격자필터(II) - 최소자승 격자 알고리즘

(Adaptive Bilinear Lattice Filter(II)-Least Squares Lattice Algorithm)

白 興 基*

(Heung Ki Baik)

要 約

비선형 시스템을 쌍선형 필터로 모델링할 때 사용할 수 있는 2개의 고속 순환 최소자승 격자 알고리즘을 본 논문에서 제시하였다. 본 논문에서 제시한 격자 알고리즘은 좋은 수치적 특성을 갖으며 기존의 방법에 비해 계산량을 대폭 줄일 수 있는 장점이 있다. 첫 번째 방법인 식오차 알고리즘은 적응필터의 출력을 계산하기 위해 측정된 원하는 응답신호를 직접 이용한 것으로 개념적으로 간단하나 관측잡음이 있을 경우에 시스템의 추정이 바이어스(bias)되기 때문에 관측잡음이 작은 경우 유용하며, 두 번째 접근 방법인 출력오차 알고리즘은 현재의 출력을 얻기 위해 과거의 적응필터의 출력을 이용하는 것으로 준최적 최소자승 출력오차 적응필터를 실현할 수 있다. 컴퓨터 시뮬레이션을 통하여 적응 쌍선형 알고리즘의 특성을 보였으며 출력오차 알고리즘이 식오차 알고리즘에 비해 관측잡음의 영향을 덜 받음을 보였다.

Abstract

This paper presents two fast least-squares lattice algorithms for adaptive nonlinear filters equipped with bilinear system models. The lattice filters perform a Gram-Schmidt orthogonalization of the input data and have very good numerical properties. Furthermore, the computational complexity of the algorithms is an order of magnitude smaller than previously available methods. The first of the two approaches is an equation error algorithm that uses the measured desired response signal directly to compute the adaptive filter outputs. This method is conceptually very simple; however, it will result in biased system models in the presence of measurement noise. The second approach is an approximate least-squares output error solution. In this case, the past samples of the output of the adaptive system itself are used to produce the filter output at the current time. Results of several experiments that demonstrate and compare the properties of the adaptive bilinear filters are also presented in this paper. These results indicate that the output error algorithm is less sensitive to output measurement noise than the equation error method.

I. 서 론

일반적으로 비선형 시스템은 Volterra 급수에 의

해 잘 모델링된다. 이 중에서 절단된 Volterra 급수에 바탕을 둔 2차 Volterra 필터가 많이 사용되고 있으며 이에 대한 연구는 거의 다 트랜스버설(transversal) 필터에 대한 LMS 알고리즘이거나 이의 변형이다.^{[1]-[5]} 그러나 이런 필터는 수렴속도가 느리고 수렴속도가 입력신호에 의존한다는 단점이 있다. 참고문헌[6]에 격자 직교화를 이용하여 2차 적응

*正會員, 全北大學校 電子工學科
(Dept. of Elec. Eng., Chonbuk Nat'l Univ.)
接受日字: 1991年 10月 2日

Volterra 필터의 수렴속도를 높이는 알고리즘이 소개되어 있으나 이 방법은 입력신호가 가우시안이라고 가정하였으며 수렴속도 역시 그다지 크지 않다. 이런 방법과 다른 방법으로 최소 자승 기법을 비선형 적응 필터에 도입하여 수렴특성을 좋게한 최소자승 알고리즘에 대한 연구가 최근에 연구되었다.^{[7]-[10]}

한편 Volterra 필터는 비선형 시스템을 모델링하는데 많은 계수를 필요로 하며 이에 따라 많은 계산량을 필요로 한다. 이러한 단점을 극복하기 위하여 계환을 포함하는 모델로서 가장 간단한 모델인 쌍선형 시스템 모델을 생각할 수 있다. 이 모델의 입력과 출력사이의 관계는 다음과 같다.

$$y(n) = \sum_{i=0}^{N-1} a_i x(n-i) + \sum_{i=0}^{N-1} \sum_{j=1}^{N-1} b_{ij} x(n-i) y(n-j) + \sum_{j=1}^{N-1} c_j y(n-j) \quad (1)$$

쌍선형 시스템은 Volterra 시스템 모델에 비해 많은 장점을 갖고 있다. 그러나 이와 같은 시스템 모델의 장점에도 불구하고 비선형 계환 모델을 이용한 적응필터에 관한 연구는 아주 적다.^{[11]-[15]} 참고문헌 [11], [12], [14]는 직접형 (direct-form) 구조와 최소자승 알고리즘과 이의 변형에 대해 다루고 있으나 이것은 계산이 매우 복잡하고 이런 종류의 알고리즘은 거의 다 수치적인 문제점 (numerical problem)을 갖고 있다. 참고문헌 [15]는 미지의 파라미터가 오직 잡음 통계적일 때 적응 쌍선형 필터링에 대한 칼만 필터를 다루고 있다. 이 방법은 데이터를 그람-슈미트 (gram-schmidt) 직교화 하는 것으로 이 방법을 이용하여 식(1)의 구조를 적응필터로 실현하는 데는 매 회당 $O(N^4)$ 의 연산이 필요하다. 한편 격자구조는 수렴속도가 빠르고 수치적으로 안정한 알고리즘을 만들 수 있는 장점이 있다. 쌍선형 시스템의 격자구조에 대한 연구가 참고문헌 [16]-[18]에 나타나 있다.

본 논문은 2개의 논문으로 구성된 논문의 두 번째 논문으로서 쌍선형 격자필터의 최소자승 적응 격자 알고리즘에 관한 논문이다. 본 논문에서 제시한 두 알고리즘은 적응필터에서 출력계환이 이루어 지는 방법에 따라 다르다. 첫번째 방법인 식오차 (equation error) 적응필터는 적응필터 출력을 계산하기 위해 원하는 응답신호를 직접 사용하며, 두번째 방법인 출력오차 (output error) 알고리즘은 현재의 출력을 얻기 위해 과거의 적응필터 출력을 이용하며 준최적 (suboptimal) 최소자승 출력오차 적응필터를 실현할 수 있다. 첫번째 방법은 관측잡음 (measurement noise)이 있을 경우 시스템 추정이 바이어스 (bias) 되

기 때문에 관측잡음이 작을 경우에만 유용하다. 이와 달리 출력오차 알고리즘은 관측오차의 영향을 감소시킨다.

본 논문은 적응 쌍선형 격자필터에 관한 두 번째 논문으로서 다음과 같이 구성되어 있다. 쌍선형 격자필터에 대한 간단한 설명이 II장에 있고 III장에 쌍선형 격자 필터에 대한 순환 최소자승 알고리즘이 유도되어 있다. 이 알고리즘에 대한 여러가지 실험 결과가 IV장에 있으며 결론이 V장에 나타나 있다.

II. 쌍선형 격자필터

원하는 응답신호 $y(n)$ 을 입력신호 $x(n)$ 에 대해 쌍선형 시스템의 응답으로 적응시켜 추정하는 문제에 대해 생각하여 보자. 이 때 쌍선형 시스템의 파라미터는 다음과 같은 오차함수를 매 순간마다 최소화함으로써 순환적으로 얻을 수 있다.

$$\zeta_n(n) = \sum_{k=1}^n \lambda^{n-k} [y(y) - \hat{y}_n(k)]^T \quad (2a)$$

여기서 λ ($0 < \lambda \leq 1$)는 적응필터의 기억용량을 결정해주는 상수이고, $\hat{y}_n(k)$ 은 시간 n 일 때의 적응필터의 파라미터에 의한 $y(k)$ 의 추정치이다.

$\hat{y}_n(n)$ 을 추정하는 방법에 따라 위의 문제를 푸는데 두 가지 방법을 생각할 수 있다. 첫번째 방법은 식오차 방법 (equation error formulation)으로 적응필터의 출력이 다음과 같이 추정된다.

$$\hat{y}_n(n) = \sum_{i=0}^{N-1} a_i(n) x(n-i) + \sum_{i=0}^{N-1} \sum_{j=1}^{N-1} b_{ij}(n) x(n-i) y(n-j) + \sum_{j=1}^{N-1} c_j(n) y(n-j) \quad (2b)$$

여기서 $a_i(n)$, $b_{ij}(n)$, $c_j(n)$ 은 시간 n 에서의 적응 쌍선형필터의 계수이다. 두 번째 방법은 출력오차 방법 (output error formulation)으로 적응필터의 출력이 다음과 같이 추정된다.

$$\hat{y}_n(n) = \sum_{i=0}^{N-1} a_i(n) x(n-i) + \sum_{i=0}^{N-1} \sum_{j=1}^{N-1} b_{ij}(n) x(n-i) \hat{y}_{n-j}(n-j) + \sum_{j=1}^{N-1} c_j(n) \hat{y}_{n-j}(n-j) \quad (2c)$$

이 방법은 시간 n 일 때의 적응필터의 계수가 위의 식에서 볼 수 있는 과거의 추정치 $y_{n-j}(n-j)$; $j=1, 2, \dots, N-1$ 과 관계가 있다는 면에서 준최적 최소자승 (suboptimal least square) 오차의 해를 준다.

식오차 방법의 직접해 (direct solution)는 기존의 방법을 이용하여 구할 수 있다. 입력신호벡터 $U_n(n)$ (백터크기 $N^2 + N - 1$)과 계수벡터 $W_n(n)$ 을 다음과 같이 각각 정의하자.

$$U_N(n) = [x(n), \dots, x(n-N+1), x(n)y(n-1), \dots, x(n-N+1)y(n-N+1), y(n-1), \dots, y(n-N+1)]^T \quad (2d)$$

$$W_N(n) = [a_0(n), \dots, a_{N-1}(n), b_{0,1}(n), \dots, b_{N-1,N-1}(n), c_1(n), \dots, c_{N-1}(n)]^T \quad (2e)$$

여기서 $(\cdot)^T$ 는 전치행렬을 나타낸다. 식(2a)은 다음과 같이 간단히 나타낼 수 있다.

$$\zeta_N(n) = \sum_{k=1}^n \lambda^{n-k} [y(k) - W_N^T(n) U_N(k)]^2 \quad (2f)$$

최적계수벡터 $W_N^*(n)$ 는 $\zeta_N(n)$ 를 최소화함으로써 다음과 같이 구해진다.^[21]

$$W_N^*(n) = R_N^{-1}(n) P_N(n) \quad (2g)$$

여기서

$$R_N(n) = \sum_{k=1}^n \lambda^{n-k} U_N(k) U_N^T(k) \quad (2h)$$

$$P_N(n) = \sum_{k=1}^n \lambda^{n-k} U_N(k) y(k) \quad (2i)$$

이다.

출력오차 방법의 해는 오직 순환법에 의해서만 구해질 수 있다. 다시 말하면 매 시간마다 $W_N(n)$ 을 구할 필요가 있다. 해는 식(2d)에 $y(k)$ 대신 $\hat{y}_k(k)$ 를 대입하고, 매 시간마다 식(2h), (2i), (2j)을 순환시킴으로써 얻을 수 있다.

위에 설명한 두가지 해는 역행렬을 구하거나 이와 유사한 방법을 사용해야 하므로 많은 계산량을 필요로 한다. 참고문헌 [11], [14]에서 추구한 방법은 이러한 직접법의 변형이므로 매 순간마다 최소한 $O(N^4)$

의 연산을 수행해야 한다. 따라서 $W_N^*(n)$ 을 구하는 데 빠른 알고리즘이 사용되고 있으나 이 방법은 거의 다 수치적인 문제점 (numerical problem)을 가지고 있다. 이러한 문제점을 극복하기 위한 방법으로 쌍선형 필터의 격자구조가 참고문헌 [16]-[18]에 소개되어 있다. 쌍선형 필터의 격자구조가 그림 1에 완전한 격자식이 표1에 나타나 있다.

표 1. 쌍선형 격자필터
Table 1. Bilinear lattice filter.

<i>Initialization</i>	
$f_0(n) = b_0(n) = x(n)$	(T-1a)
$f_0^{(p)}(n) = [d(n-1), x(n-1)d(n-1), x(n)d(n-1)]^T, p=1$ $[x(n-p)d(n-1), x(n)d(n-p)]^T, p=2, 3, \dots, N-1$	(T-2b)
$e_0(n) = y(n)$	(T-1c)
<i>Iteration Procedure</i>	
DO (T-1d) - (T-1g), for $m=1, 2, \dots, N-1$	
$f_m(n) = \begin{bmatrix} f_{m-1}(n) - K_m^{\sigma T}(n) b_{m-1}(n-1) \\ f_{m-1}^{(m)}(n) - K_m^{\sigma m T}(n) b_{m-1}(n-1) \end{bmatrix}$	(T-1d)
$b_m(n) = \begin{bmatrix} b_{m-1}(n-1) - K_m^{b T}(n) f_{m-1}(n) \\ f_{m-1}^{(m)}(n) - K_m^{b m T}(n) f_{m-1}(n) \end{bmatrix}$	(T-1e)
DO (T-1f), for $p=m+1, m+2, \dots, N-1$	
$f_m^{(p)}(n) = f_{m-1}^{(p)}(n) - K_m^{\sigma p T}(n) b_{m-1}(n-1)$	(T-1f)
$e_m(n) = e_{m-1}(n) - k_m^{\sigma T}(n) b_{m-1}(n)$	(T-1g)
$e_N(n) = e_{N-1}(n) - k_N^{\sigma T}(n) b_{N-1}(n)$	(T-1h)
<i>For equation error formulation, $d(n) = y(n)$.</i>	
<i>For output error formulation, $d(n) = y(n) - e_N(n)$.</i>	

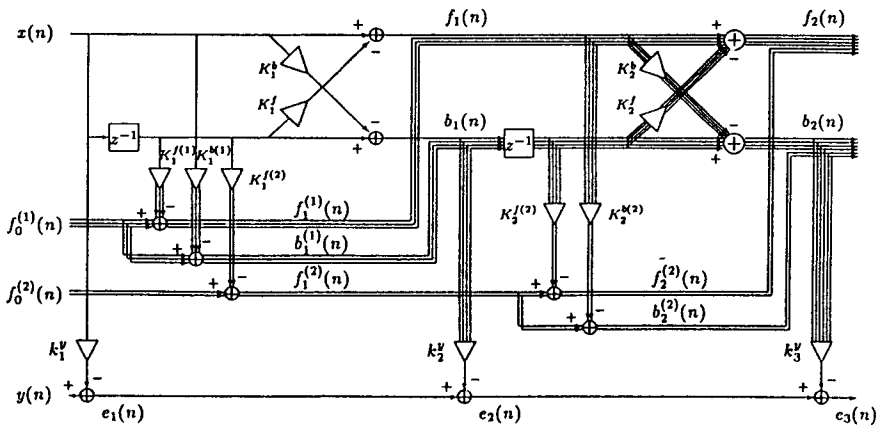


그림 1. N=3일 때의 쌍선형 격자구조의 블록도
Fig. 1. Block diagram of the bilinear lattice structure for N=3.

III. 순환 최소자승 적응 쌍선형 격자 알고리즘

그림1의 격자구조로 부터 적응 알고리즘을 유도하는 과정은 간단하며 참고문헌 [19]의 방법과 유사하다. $K_m^f(n)$ 을 갱신하는 과정에 대해 생각하여 보자. 다른 변수에 대한 유도과정은 비슷하다. 최소자승 적응 격자 쌍선형 알고리즘에 관한 완전한 순환식은 표2에 주어져 있다.

$K_m^f(n)$ 은 $f_{m-1}(n)$ 을 $b_{m-1}(n-1)$ 의 선형결합으로 추정할 때의 최적 계수벡터로서 다음과 같다.

$$K_m^f(n) = R_{m-1}^{-b}(n-1) \Delta_m(n) \quad (3a)$$

여기서

$$R_{m-1}^b(n) = \sum_{k=1}^n \lambda^{n-k} b_{m-1,n}(k) b_{m-1,n}^T(k) \quad (3b)$$

$$\Delta_m(n) = \sum_{k=1}^n \lambda^{n-k} b_{m-1,n}(k-1) f_{m-1,n}^T(k) \quad (3c)$$

는 각각 $b_{m-1,n}(n)$ 의 최소자승 자기상관행렬, $b_{m-1,n}(n-1)$ 과 $f_{m-1,n}(n)$ 의 최소자승 교차상관행렬이다. 위 식에서 예측오차벡터의 두번째 밀침차 n 은 예측오차가 시간 n 일 때의 최적계수를 이용하여 계산된 것을 나타낸다. 역방향 및 순방향 예측오차 $b_m(n)$ 과 $f_m(n)$ 은 위의 양과 다음과 같은 관계가 있다.

$$b_m(n) = b_{m,n}(n) \quad (3d)$$

$$f_m(n) = f_{m,n}(n) \quad (3e)$$

이제 앞으로 예측오차벡터에서 두번째 침차를 생략하기로 하자. 교차상관행렬과 자기상관행렬은 다음과 같이 갱신된다!^[20]

$$\Delta_m(n) = \lambda \Delta_m(n-1) + b_{m-1}(n-1) f_{m-1}^T(n) / \alpha_{m-1}(n-1) \quad (3f)$$

$$R_{m-1}^b(n) = \lambda R_{m-1}^b(n-1) + b_{m-1}(n) b_{m-1}^T(n) / \alpha_{m-1}(n) \quad (3g)$$

여기서

$$\alpha_m(n) = \alpha_{m-1}(n) - b_{m-1}^T(n) R_{m-1}^{-b}(n) b_{m-1}(n) \quad (3h)$$

이고, $\alpha_m(n)$ 은 m 차 단의 likelihood로서 다음과 같은 값을 갖는다. ^[20]

$$0 < \alpha_m(n) \leq 1 \quad (3i)$$

자기상관행렬의 역행렬은 행렬 역변환 공리를 이용하여 순환적으로 구할 수 있다!^[21]

$$R_m^{-b}(n) = \lambda^{-1} R_m^{-b}(n-1) - \frac{\lambda^{-2} R_m^{-b}(n-1) b_m(n) b_m^T(n) R_m^{-b}(n-1)}{\alpha_m(n) + \lambda^{-1} b_m^T(n) R_m^{-b}(n-1) b_m(n)} \quad (3j)$$

표 2. 순환 최소자승 적응 쌍선형 격자 알고리즘

Table 2. The recursive least squares adaptive bilinear lattice algorithm.

Initialization

DO (T-2a) - (T-2b), for $m=0, 1, \dots, N-1$

$$\alpha_m(0) = 1 \quad (T-2a)$$

$$R_m^f(0) = R_m^b(0) = \begin{cases} \delta, & \text{if } m=0 \\ \delta I_{2m+2}, & \text{otherwise} \end{cases} \quad (T-2b)$$

Iteration Procedure

DO (T-2c) - (T-2u), for $n=1, 2, \dots$

$$\alpha_0(n) = 1 \quad (T-2c)$$

$$R_0^f(n) = R_0^b(n) = \lambda R_0^f(n-1) + x(n)^2 \quad (T-2d)$$

$$f_0(n) = b_0(n) = x(n) \quad (T-2e)$$

$$f_0^{(p)}(n) = \begin{cases} [d(n-1), x(n-1)d(n-1), x(n)d(n-1)]^T, & p=1 \\ [x(n-p)d(n-1), x(n)d(n-p)]^T, & p=2, 3, \dots, N-1 \end{cases} \quad (T-2f)$$

$$e_0(n) = y(n) \quad (T-2g)$$

DO (T-2h) - (T-2s), for $m=1, 2, \dots, N-1$

$$\Delta_m(n) = \lambda \Delta_m(n-1) + b_{m-1}(n-1) f_{m-1}^T(n) / \alpha_{m-1}(n-1) \quad (T-2h)$$

$$\Delta_m^{f(m)}(n) = \lambda \Delta_m^{f(m)}(n-1) + b_{m-1}(n-1) f_{m-1}^{(m)T}(n) / \alpha_{m-1}(n-1) \quad (T-2i)$$

$$\Delta_m^{b(m)}(n) = \lambda \Delta_m^{b(m)}(n-1) + f_{m-1}(n) f_{m-1}^{(m)T}(n) / \alpha_{m-1}(n-1) \quad (T-2j)$$

$$f_m(n) = \begin{bmatrix} f_{m-1}(n) - \Delta_m^T(n) R_{m-1}^{-b}(n-1) b_{m-1}(n-1) \\ f_{m-1}^{(m)}(n) - \Delta_m^{f(m)T}(n) R_{m-1}^{-b}(n-1) b_{m-1}(n-1) \end{bmatrix} \quad (T-2k)$$

$$b_m(n) = \begin{bmatrix} b_{m-1}(n-1) - \Delta_m(n) R_{m-1}^{-f}(n) f_{m-1}(n) \\ f_{m-1}^{(m)}(n) - \Delta_m^{b(m)}(n) R_{m-1}^{-f}(n) f_{m-1}(n) \end{bmatrix} \quad (T-2l)$$

DO (T-2m) - (T-2n), for $p=m+1, m+2, \dots, N-1$

$$\Delta_m^{f(p)}(n) = \lambda \Delta_m^{f(p)}(n-1) + b_{m-1}(n-1) f_{m-1}^{(p)T}(n) / \alpha_{m-1}(n-1) \quad (T-2m)$$

$$f_m^{(p)}(n) = f_{m-1}^{(p)}(n) - \Delta_m^{f(p)T}(n) R_{m-1}^{-b}(n-1) b_{m-1}(n-1) \quad (T-2n)$$

$$\Delta_m^f(n) = \lambda \Delta_m^f(n-1) + b_{m-1}(n) e_{m-1}(n) / \alpha_{m-1}(n) \quad (T-2o)$$

$$e_m(n) = e_{m-1}(n) - \Delta_m^{fT}(n) b_{m-1}(n) \quad (T-2p)$$

$$\alpha_m(n) = \alpha_{m-1}(n) - b_{m-1}^T(n) R_{m-1}^{-b}(n) b_{m-1}(n) \quad (T-2q)$$

$$R_m^{-f}(n) = \lambda^{-1} R_m^{-f}(n-1) - \frac{\lambda^{-2} R_m^{-f}(n-1) f_m(n) f_m^T(n) R_m^{-f}(n-1)}{\alpha_m(n-1) + \lambda^{-1} f_m^T(n) R_m^{-f}(n-1) f_m(n)} \quad (T-2r)$$

$$R_m^{-b}(n) = \lambda^{-1} R_m^{-b}(n-1) - \frac{\lambda^{-2} R_m^{-b}(n-1) b_m^T(n) R_m^{-b}(n-1)}{\alpha_m(n) + \lambda^{-1} b_m^T(n) R_m^{-b}(n-1) b_m(n)} \quad (T-2s)$$

$$\Delta_n^f(n) = \lambda \Delta_n^f(n-1) + b_{n-1}(n) e_{n-1}(n) / \alpha_{n-1}(n) \quad (T-2t)$$

$$e_n(n) = e_{n-1}(n) - \Delta_n^{fT}(n) R_{n-1}^{-b}(n) b_{n-1}(n) \quad (T-2u)$$

† For equation error formulation, $d(n) = y(n)$.

For output error formulation, $d(n) = y(n) - e_n(n)$.

$K_m^b(n)$ 의 순환식은 $b_{m-1}(n-1)$ 이 $f_{m-1}(n)$ 의 선형 결합으로 추정될 때의 최적 계수벡터가 $K_m^b(n)$ 이라는 사실로 부터 쉽게 유도할 수 있다. 관련식이 표2의 (T-2h), (T-2l)의 윗부분, (T-2q)에 나타나 있다. 보조변수 $K_m^{(p)}(n)$ 과 $K_m^{(m)}(n)$ 은 비슷하게 갱신될 수 있다. $K_m^{(p)}(n)$ 은 $b_{m-1}(n-1)$ 을 사용하여 $f_{m-1}^{(p)}(n)$ 을 추정할 때의 최적 계수행렬이고, $K_m^{(m)}(n)$ 은 $f_{m-1}(n)$ 을 사용하여 $f_{m-1}(n)$ 을 사용하여 $f_{m-1}^{(m)}(n)$ 을 추정할 때의 최적 계수행렬이다.

접합과정(joint-process) 추정에 대한 순환식의 유도는 앞에서 설명한 유도와 유사하다. $b_m(n): m=0, 1, \dots, N-1$ 의 직교화는 $b_m(n)$ 을 사용하여 $y(n)$ 을 추정하는 것이 $e_{m-1}(n)$ ($b_0(n), b_1(n), \dots, b_{m-1}(n)$)을 사용하여 $y(n)$ 을 추정할 때의 오차)을 추정하는 것과 같은 결과를 준다. 표2의 식 (T-2p) - (T-2s), (T-2t), (T-2u)이 접합과정 추정기 계수벡터와 접합과정 추정오차 수열의 순환식임을 쉽게 알 수 있다.

각 순환식에 필요한 곱셈과 나눗셈의 연산횟수가 표3에 나타나 있다. $N \geq 2$ 일 때 각 iteration에 필요한 곱셈과 나눗셈의 연산수는 $46/3 N^3 + 10N^2 - 1/3N$

표 3. 적응 쌍선형 격자 알고리즘의 계산량
Table 3. Computational complexity of the adaptive bilinear lattice algorithm.

식 번호	곱셈과 나눗셈의 수	식 번호	곱셈과 나눗셈의 수
(T-2d)	2	(T-2p)	$n_m l_p$
(T-2g)	$2(N-1)$	(T-2q)	$2n_m + 1$
(T-2j)	l_m	‡	n_m^2
†	$2n_m l_m$	(T-2r)	n_m
(T-2k)	$2n_m l_m$	(T-2s)	n_m
(T-2l)	$n_m^2 + n_m l_m$	(T-2t) #	$2n_m m^2 + 3n_m m + 2$
(T-2m) ††	$2n_m^2 + n_m l_m$	(T-2u)	$2n_m m^2 + 3n_m m + 2$
(T-2n)	$2n_m^2 + n_m l_m$	(T-2v)	$2n_m + 1$
(T-2o)	$2n_m l_p + l_p$	(T-2v-v)	$n_m^2 + n_m$
$\sum_{m=1}^{N-1} (4n_{m+1} + 6n_{m+1} + 6n_m l_m + 6n_m^2 + 5n_m + l_m + 5)$ $+ \sum_{m=1}^{N-1} \sum_{p=m+1}^{N-1} (3n_m l_p + l_p) - n_m^2 + 2N - 1$ $= 46/3 N^3 + 10N^2 - 1/3N - 50; N \geq 2$			

$$n_m = \begin{cases} 1, & m=1 \\ 2m, & m=2, 3, \dots, N-1 \end{cases}$$

$$l_p = \begin{cases} 3, & p=1 \\ 2, & p=2, 3, \dots, N-1 \end{cases}$$

- † 식 (T-1i)와 (T-2j)에서의 $f_{m-1}^{(m)}(n)/\alpha_{m-1}(n-1)$ 의 계산
- †† $R_{m-1}^{-b}(n)b_{m-1}(n)$ 의 계산
- ‡ 식 (T-2k)과 (T-2n)에서의 $R_{m-1}^{-b}(n-1)b_{m-1}(n-1)$ 은 $R_{m-1}^{-b}(n)b_{m-1}(n)$ 의 과거치이다.
- # $m=N-1$ 일 때 식 (T-2r)의 계산은 불필요하다.

-50이다. 이 연산수는 $O(N^3)$ 으로 N 이 클 경우 이전의 알고리즘의 연산수 $O(N^4)$ 에 비해 작다.

이 알고리즘의 초기조건은 기존의 최소자승 격자 알고리즘[20]의 초기조건과 같으며 다음과 같다.

$$A_m(0) = A_m^{(p)}(0) = A_m^{(m)}(0) = \Delta_m^y(0) = 0 \quad (3k)$$

$$R_m^y(0) = R_m^b(0) = \begin{cases} \delta, & m=0 \\ \delta I_{2m+2}, & m=1, 2, \dots, N-1 \end{cases} \quad (3l)$$

$$\alpha_m(0) = 1 \quad (3m)$$

여기서 δ 는 작은 양의 상수이며 I_m 은 $m \times m$ 단위행렬이다. 정확한 초기화가 가능하나 이 논문에서는 다루지 않는다.

IV. 실험결과

두 알고리즘의 성능을 평가하기 위하여 미지의 쌍선형 시스템의 식별 실험을 하였다. 또한 $O(N^4)$ 의 계산량이 필요한 여러 알고리즘과의 성능 비교도 아울러 실시하였다. 적응필터의 입력신호 $x(n)$ 은 전달함수가 다음과 같은 저역통과필터에 평균이 0인 백색 가우시안 잡음을 통과하여 얻었고 신호의 크기는 출력신호의 분산이 1이 되도록 조정하였다.

$$H(z) = \frac{1}{1 - 1.6z^{-1} + 0.95z^{-2}} \quad (4)$$

원하는 응답신호 $y(n)$ 은 표3에 있는 계수를 가진 쌍선형 필터의 출력에 잡음을 섞었다. 이 때 사용된 관측잡음은 평균이 0인 백색 의사 가우시안 잡음으로 입력신호와와는 상관관계가 없다. 모든 실험결과는 50번의 독립된 실험의 앙상블 평균치이다.

그림2와 그림3은 각각 $\lambda=0.995$ 일 때의 식오차와 출력오차 방법의 평균자승 예측오차를 나타낸다. 신호대 잡음비 (SNR)가 $\infty, 30\text{dB}, 20\text{dB}$ 인 세 개의 서로 다른 잡음 레벨에 대해 실험이 행해졌다. 이 그림들로부터 두 알고리즘이 좋은 수렴속도를 가짐을 알 수 있다. $n=4001$ 부터 $n=5000$ 까지의 평균자승 오차의 앙상블 평균치의 시간 평균치가 표4에 나타나 있다.

표5는 측정오차가 있을 경우 출력오차 알고리즘이 식오차 알고리즘 보다 좋음을 나타낸다. 두 알고리즘의 성능 차이는 서로 다른 잡음레벨에서의 평균 계수제적을 보면 더욱 잘 알 수 있다. $\lambda=0.995$ 일 때의 서로 다른 잡음레벨에 대한 계수 $a_1(n), c_1(n), b_{22}(n)$ 의 평균제적이 각각 그림4-그림6에 나타나 있다. 앙상블 평균치는 격자 파라미터를 직접형 파라미터로 변환시킨 후 평균하여 얻은 것이다. 표5는

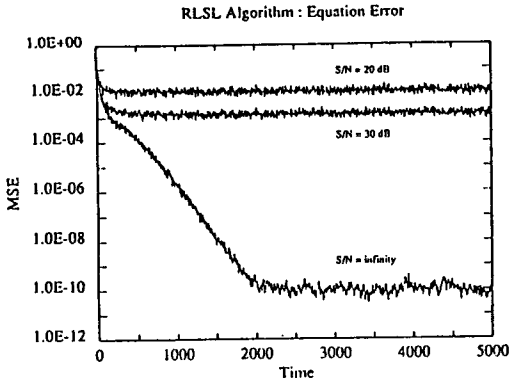


그림 2. 식오차 알고리즘의 학습곡선
Fig. 2. Learning curves for the equation error algorithm.

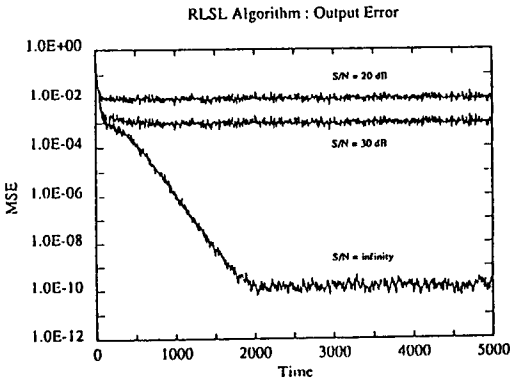


그림 3. 출력오차 알고리즘의 학습곡선
Fig. 3. Learning curves for the output error algorithm.

마지막 1000개의 샘플에 대한 계수제적의 시간 평균치를 나타낸다. 여기서 출력에 잡음이 있는 경우 식오차 방법에서는 계수들이 미지의 시스템의 계수와 다른 값으로 수렴하는 것을 알 수 있다. 출력오차 적응필터에서의 계수의 평균치가 측정잡음에 상당히 덜 민감한 것을 알 수 있다. 또한 두 알고리즘 모두 비선형 항에 대한 계수가 선형 항에 대한 계수들에 비해 잡음에 더욱 민감함을 알 수 있다.

참고문헌 [19]에서 논의된 계산량이 많은 직접형의 두 알고리즘과의 성능비교에 관한 실험도 실시하였다. 그림7과 그림8은 각각 확장된 최소자승(ELS) 알고리즘과 순환 예측오차 방법(RPEM)에서의 계수

표 4. 실험에 사용된 미지의 쌍선형 시스템의 계수

Table 4. Coefficients of the unknown bilinear system used in the experiments.

$a_0 = 1.0$	$a_1 = 1.0$	$a_2 = 1.0$
$b_{0,1} = 0.3$	$b_{1,1} = -0.2$	$b_{2,1} = 0.1$
$b_{0,2} = 0.1$	$b_{1,2} = -0.2$	$b_{2,2} = 0.3$
	$c_1 = 0.5$	$c_2 = -0.5$

표 5. 마지막 1000개의 데이터 샘플의 평균자승오차의 시간 평균치

Table 5. Time-average mean-squared error over the last 1000 data samples.

SNR	Equation Error		Output Error	
	$\lambda = 0.995$	$\lambda = 0.99$	$\lambda = 0.995$	$\lambda = 0.99$
∞	0.104E-9	0.577E-10	0.137E-9	0.714E-10
30dB	0.136E-2	0.127E-2	0.942E-3	0.887E-3
20dB	0.118E-1	0.109E-1	0.941E-2	0.886E-2

표 6. 마지막 1000개의 데이터 샘플의 계수의 시간 평균치

Table 6. Time-averaged coefficients over the last 1000 data samples.

	Equation Error			Output Error			Optimal Coeff.
	∞	30dB	20dB	∞	30dB	20dB	
a_0	1.0	0.997	0.993	1.0	0.998	0.994	1.0
a_1	1.0	1.146	1.757	1.0	1.005	1.036	1.0
a_2	1.0	1.003	0.808	1.0	0.997	0.990	1.0
$b_{0,1}$	0.3	0.095	0.309	0.3	0.305	0.315	0.3
$b_{0,2}$	0.1	-0.215	0.091	0.1	0.096	0.086	0.1
$b_{1,1}$	-0.2	-0.171	-0.277	-0.2	-0.210	-0.233	-0.2
$b_{1,2}$	-0.2	0.090	-0.077	-0.2	-0.193	-0.176	-0.2
$b_{2,1}$	0.1	0.090	0.116	0.1	0.108	-0.127	0.3
$b_{2,2}$	0.3	0.313	0.311	0.3	0.294	0.281	0.3
c_1	0.5	0.436	0.203	0.5	0.499	0.489	0.5
c_2	-0.5	-0.471	-0.322	-0.5	-0.499	-0.494	-0.5

$c_1(n)$ 의 평균제적을 나타낸 것이다. 참고문헌 [14]에서 논의한 바와 같이 ELS 알고리즘은 수치적으로 불안함을 그림7이 잘 나타내고 있다. RPEM 알고리즘은 참고문헌 [14]에서 논의된 $O(N^4)$ 의 계산량을 갖는 알고리즘 중 가장 좋은 것으로서 본 논문의 쌍선형 격자필터의 출력오차 방법과 거의 같은 정도의 잡음 영향을 받는 것을 알 수 있다. 따라서 본 논문에서 제안한 출력오차 쌍선형 격자 알고리즘이 $O(N^3)$ 의 계산량을 갖으면서도 RPEM 알고리즘과 거의 같은 성능을 갖음을 잘 알 수 있다.

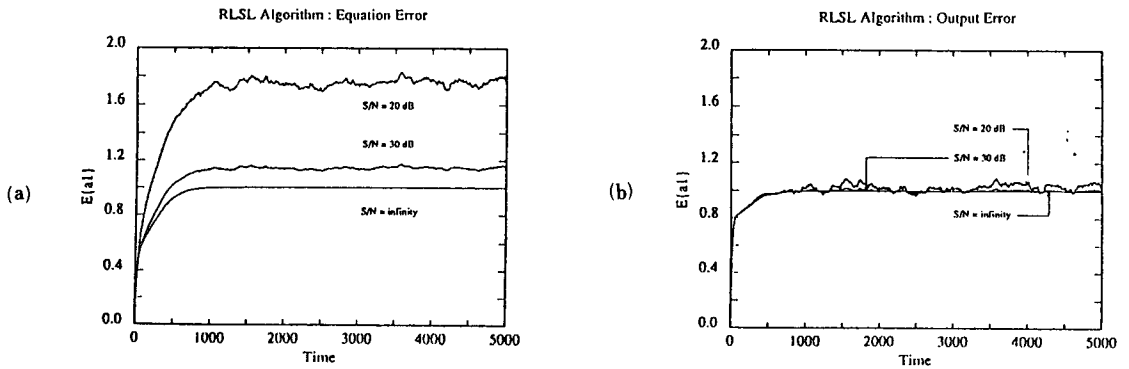


그림 4. 서로 다른 신호대 잡음비에 대한 계수 $a_1(n)$ 의 평균 궤적

(a) 식오차 알고리즘 (b) 출력오차 알고리즘

Fig. 4. Mean trajectories of coefficient $a_1(n)$ for different output SNRs:

(a) equation error algorithm, (b) output error algorithm.

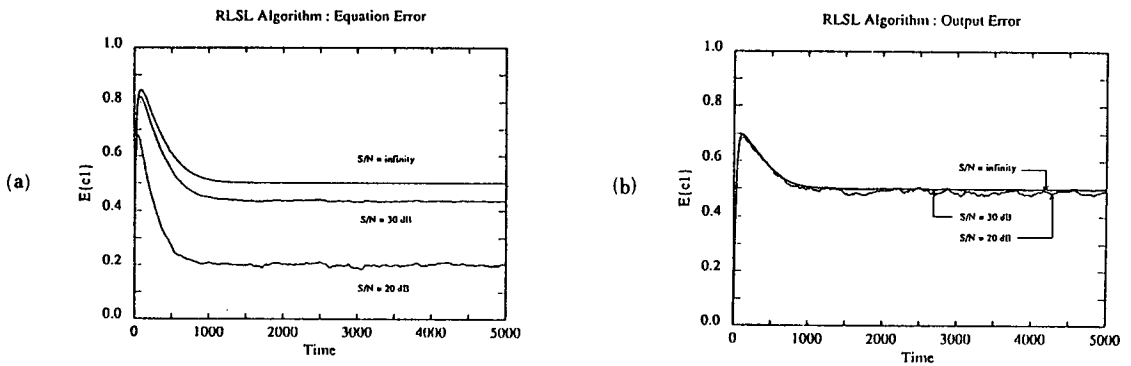


그림 5. 서로 다른 신호대 잡음비에 대한 계수 $c_1(n)$ 의 평균 궤적

(a) 식오차 알고리즘 (b) 출력오차 알고리즘

Fig. 5. Mean trajectories of coefficient $c_1(n)$ for different output SNRs:

(a) equation error algorithm, (b) output error algorithm.

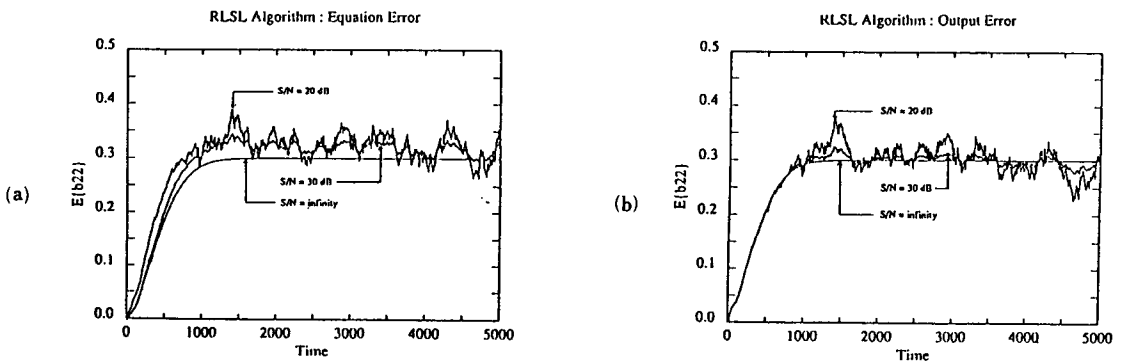


그림 6. 서로 다른 신호대비 잡음비에 대한 계수 $b_{2,2}(n)$ for different output SNRs:

(a) 식오차 알고리즘 (b) 출력오차 알고리즘

Fig. 6. Mean trajectories of coefficient $b_{2,2}(n)$ for different output SNRs:

(a) equation error algorithm, (b) output error algorithm.

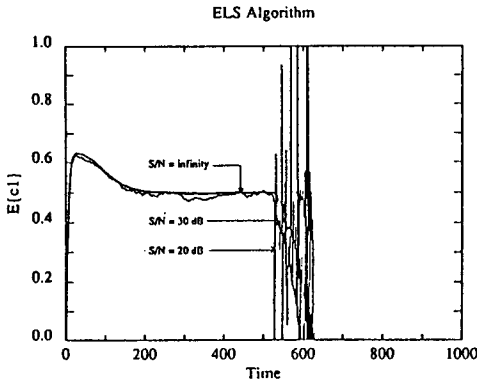


그림 7. ELS 알고리즘에서의 서로 다른 신호대 잡음비에 대한 계수 $c_1(n)$ 의 평균 궤적
 Fig. 7. Mean trajectories of coefficient $c_1(n)$ for different output SNRs for ELS algorithm.

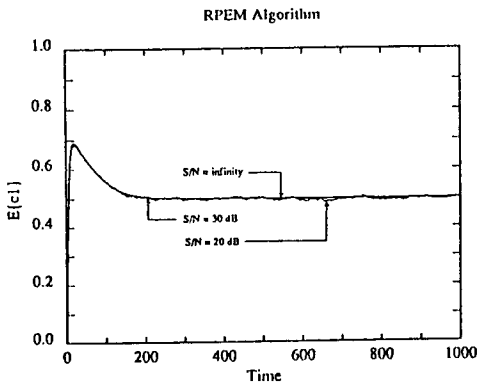


그림 8. RPEM에서의 서로 다른 신호대 잡음비에 대한 계수 $c_1(n)$ 의 평균궤적
 Fig. 8. Mean trajectories of coefficient $c_1(n)$ for different output SNRs for RPEM.

V. 결 론

쌍선형 필터의 격자구조와 $O(N^3)$ 의 계산량을 갖는 이에 대한 두 개의 적응 알고리즘이 본 논문에서 제시되어 있다. 이 계산량은 기존의 알고리즘이 $O(N^4)$ 의 계산량을 갖는 것에 비하면 상당히 작음을 잘 알 수 있다. 참고문헌[8]의 방법을 사용함으로써 $O(N^3)$ 의 계산량을 갖는 적응 쌍선형 필터의 직접형의 알고리즘을 유도하는 것은 가능하다. 이러한 알고리즘에서 대한 본 논문에서 제안한 격자필터의 장

점은 직접형 알고리즘이 일반적으로 좋지 않은 수치적 특성을 갖는 데 비해 고속이며, 수치적으로 안정하다는 점이다. 따라서 본 논문에서 제시한 알고리즘은 상대적으로 작은 계산량을 필요로 하고 좋은 수치적 특성을 갖기 때문에 적응 쌍선형 필터의 분야에 큰 기여를 하리라 믿는다.

식오차 알고리즘은 단 하나의 최소점으로 수렴하는 장점이 있다. 그러나 이 최소점은 시스템 인식을 포함하는 응용등에서 관측잡음이 있는 경우 미지의 시스템의 개수에 해당되지 않는다. 식오차 알고리즘의 성능에 대한 관측잡음의 영향은 IV장에 제시된 실험결과에 명백히 나타나 있다. 출력오차 알고리즘은 적응필터 성능에 대한 관측잡음을 줄이도록 설계되었다. 본 논문에서 제시된 실험결과에 출력오차 알고리즘이 식오차 알고리즘에 비해 관측잡음의 영향에 대해 덜 민감함을 잘 나타내고 있다. 비록 출력오차 알고리즘에 대한 이론적인 성능해석이 이루어지고 있지 않으나 참고문헌 [22]의 결과가 본 논문에서 제시된 알고리즘에 직접 적용될 수 있으리라 믿는다.

參 考 文 獻

- [1] M. Bellafemina and S. Benedetto, "Identification and equalization of nonlinear channels for digital transmission," *Proc. IEEE Int. Symp. Circuits and Systems (Kyoto, Japan)*, pp. 1477-1480, June 1985.
- [2] J.R. Casar-Corredera, M. Garcia-Otera, and A. Figeiras-Videl, "Data echo nonlinear cancellation," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing (Tampa, Florida)*, p. 32.4.1-4, Mar. 1985.
- [3] G.L. Sicouranza, A. Bucconi, and P. Mitri, "Adaptive echo cancellation with nonlinear digital filters," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing (San Diego, California)*, pp. 3.10.1-4, Mar. 1984.
- [4] M.J. Coker and D.M. Simkins, "A nonlinear adaptive noise canceller," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, pp. 470-473, 1980.
- [5] C.F.N. Cowan and P.M. Grant, "Nonlinear system modeling-concept and application," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing (San Diego, California)*, pp. 45.6.1-4, Mar. 1984.
- [6] T. Koh and E.J. Power, "Second-order

- Volterra filtering and its application to non-linear system identification," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-33, no. 6, pp. 1445-1455, Dec. 1985.
- [7] C.E. Davila, A.J. Welch, and H.G. Rylander III., "A second-order adaptive Volterra filter with rapid convergence," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-35, pp. 1259-1263, Sep. 1987.
- [8] D.H. Youn, K.K. Yu, and V.J. Mathews, "Adaptive nonlinear digital filter with sequential regression algorithm," *Proc. 22nd Annual Allerton Conf. Contr., Comm., and Computing (Urbana-Champaign, Illinois)*, pp. 152-161, Oct. 1984.
- [9] V.J. Mathews and J. Lee, "A fast least-squares second-order Volterra filter," *Proc. Int. Conf. Acoust., Speech, Signal Processing (New York)*, 1988.
- [10] M.A. Syed and V.J. Mathews, "Lattice and QR decomposition-based algorithms for recursive least squares adaptive nonlinear filters," *Proc. IEEE Int. Symp. Circuits and Systems (New Orleans, Louisiana)*, May 1990.
- [11] S.A. Billings and W.S.F. Voon, "Least square parameter estimation algorithms for nonlinear systems," *Int. J. System Sci.*, vol. 15, no. 6, pp. 601-615, 1984.
- [12] X.Y. Gao, W.M. Snelgrove, and D.A. Johns, "Nonlinear IIR adaptive filtering using a bilinear structure," *Proc. IEEE Int. Symp. Circuits and Systems (Portland, Oregon)*, May 1989.
- [13] S.R. Parker and F.A. Perry, "A discrete ARMA model for nonlinear system identification," *IEEE Trans. Circuits and Systems*, vol. CAS-28, no. 3, Mar. 1981.
- [14] F. Fnaiech and L. Ljung, "Recursive identification of bilinear systems," *Int. J. Control*, vol. 45, no. 2, pp. 453-470, 1987.
- [15] X. Yang, R.R. Mohler, and R.M. Burton, "Adaptive suboptimal filtering of bilinear filtering," *Int. J. Control*, vol. 52, no. 1, pp. 135-158, 1990.
- [16] H.K. Baik, V.J. Mathews, and R.T. Short, "Adaptive lattice bilinear filters," *SPIE Conf. Advanced Signal Processing Algorithm, Architecture, and Implementation (San Diego, California)*, 1990.
- [17] H.K. Baik and V.J. Mathews, "Adaptive lattice bilinear filters," Submitted to *IEEE Trans. Acoust., Speech, Signal Processing*, Nov. 1990.
- [18] H.K. Baik and V.J. Mathews, "Adaptive algorithms for identifying recursive nonlinear systems," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing (Toronto, Canada)*, May 1991.
- [19] F. Ling and J.G. Proakis, "A generalized multichannel least squares lattice algorithm based on sequential processing stages," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, no. 2, pp. 381-389, Apr. 1984.
- [20] G. Carayannis, D. Manolakis, and N. Kalouptsidis, "A unified view of parametric processing algorithm for prewindowed signals," *Signal Processing*, vol. 10, pp. 335-368, 1986.
- [21] S. Haykin, *Adaptive Filter Theory*, Englewood Cliffs, New Jersey: Prentice Hall, 1985.
- [22] J.B. Moore, "Global convergence of output error recursions in colored noise," *IEEE Trans. Automat. Contr.*, vol. AC-27, no. 6, pp. 1189-1199, Dec. 1982.

 著 者 紹 介

白 興 基 (正會員) 第29卷 B編 第1號 參照

현재 전북대학교 공과대학
전자공학과 부교수