

# 수치적으로 안정한 고속 Transversal 필터

김 의 준<sup>o</sup>, 정 용 희, 정 한 수  
승실대학교 전기공학과

## Numerically Stable Fast transversal filter

Eui-Jun Kim<sup>o</sup>, Yong-Hee Chung, Chan-Soo Chung  
Department of Electrical Engineering, Soong Sil University

### Abstract

In this paper, it is proposed to improve the robustness of the Fast Recursive Least Squares(FRLS) algorithm with the exponential weighting, which is an important class of algorithms for adaptive filtering. It is well known that the FRLS algorithm is numerically unstable with exponential weighting factor  $\lambda < 1$ .

However, introducing some gains into this algorithm, numerical errors can be reduced. An accurately choice of the gains then leads to a numerically stable FRLS algorithm with a complexity of  $8m$  multiplications and we show it by computer simulations.

### 1. 서론

적용 스펙트럼 추정을 위해서는 신호를 모델링하고 그 모델링 파라미터를 사용하는 스펙트럼 추정법이 적합하다. 특히 고속적용 스펙트럼 추정을 위해서는 신호의 모델링을 고속으로 하는 것이 요구되고 이에 적합한 알고리즘은 최소평균자승(Least Mean Square: LMS) 알고리즘과 고속순차 최소자승(Fast Recursive Least Square: FRLS) 알고리즘을 꼽을 수 있다. 이들은 서로 장단점이 있으므로 필요에 따라 선택된다. FRLS 알고리즘은 LMS에 비해 수렴속도가 빠르며, 엄격한 최소자승법에 근거한 알고리즘으로 계산량은 LMS의 2~4배 정도이다. 따라서 FRLS 알고리즘은 LMS 알고리즘과 충분히 경쟁력있는, 특히 빠른 수렴속도가 요구되는 분야에서는 유용한 알고리즘이지만 시스템 변화에 적응력을 향상시키기 위해 1보다 작은 하중계수를 사용할 경우 강인성이 보장되지 않는 결점을 가지고 있다.

이러한 FRLS의 단점을 개선하기 위해 많은 연구가 진행되어 왔다. [1], [2], [3], [13], [14] 본 논문에서는 이러한 연구를 토대로 하여 강인성이 보장되는 새로운 FRLS 알고리즘을 제시하고자 한다.

본 논문의 구성은 다음과 같다. 2장에서 FRLS의 기본개념과 강인성 보장문제에 대해 서술하고, 본 논문의 주제인 강인성이 보장된 FRLS 알고리즘은 3장에서 설명한다. 4장에서는 이 알고리즘에 대한 모의실험 결과를 제시했다.

### II. 고속순차 최소자승법과 강인성문제

최소자승법은 측정치  $d(n)$ 와 추정치  $y(n)$ 간 오차의 제곱합이 최소가 되도록 추정치  $y(n)$ 의 매개변수를 결정하는 방법이다. 이를 위해 평가함수  $J$ 를 다음과 같이 한다.

$$J = \sum_{k=0}^n \{y(k) - d(k)\}^2$$

또한 평가함수  $J$ 를 최소화하기 위한 정규방정식은 다음과 같다.

$$w(n) = R_n^{-1}(n) r_n(n)$$

여기서,  $R_n(n)$ 과  $r_n(n)$ 는

$$R_n(n) = \sum_{k=0}^n X_n(k) X_n^T(k), \quad r_n(n) = \sum_{k=0}^n X_n(k) d(k)$$

이고,  $X_n(n)$ 은 입력신호 벡터,  $d_n(n)$ 은 추정값,  $w_n(n)$ 은 필터의 계수벡터이다.

이 정규방정식의 해를 얻기 위해 LS는 역행렬의 계산에  $O(m^3)$ 의 계산량이 필요로 하다. 역행렬을 구하는 과정에서의 계산량을 줄이기 위해 역행렬의 정리를 이용하는 것이 일반형 순차최소자승(Recursive Least Square: RLS) 알고리즘인데 이는  $O(m^2)$ 의 계산량이 소요된다. 그러나 이 알고리즘도 시스템의 차수  $m$ 이 큰 경우에는 막대한 계산량이 필요하므로 실시간 처리에는 이용에 무리가 있다. 이러한 계산량문제를 해결하고자 제시된 FRLS은  $O(m)$ 의 계산량만이 필요할 뿐이다. 이는 공성의 횟수에 따라  $8m$ 형태와  $5m$ 형태가 있으며 그 알고리즘은 표 2-1, 2-2와 같다. [4] [5], [6], [7], [8]

표 2-1.  $8m$  FRLS 알고리즘

이동벡터의 시간갱신 공식	계산량
$e_m^a(n+1) = x(n+1) \cdot a_m^a(n) x_m(n)$	(1) $m$
$a_m^a(n+1) = a_m^a(n) + w_m^a(n) e_m^a(n+1)$	(2) $m$
$e_m^b(n+1) = x(n+1) \cdot a_m^b(n) x_m(n)$	(3) $m$
$a_m^b(n+1) = a_m^b(n) + e_m^b(n+1) z_m^b(n+1)$	(4) 1
$w_m^{a,b}(n+1) = \begin{bmatrix} 0 \\ w_m^a(n) \end{bmatrix} - \frac{e_m^a(n+1)}{a_m^a(n+1)} \begin{bmatrix} 1 \\ a_m^a(n+1) \end{bmatrix}$	(5) $m+1$
분할식 $w_m^{a,b}(n+1) = \begin{bmatrix} d_m^a(n+1) \\ \delta_m^a(n+1) \end{bmatrix}$	(6) 0
$e_m^b(n+1) = x(n+1) \cdot a_m^b(n) x_m(n+1)$	(7) $m$
$w_m^a(n+1) = \frac{d_m^a(n+1) - \delta_m^a(n+1) b_m(n)}{1 - \delta_m^a(n+1) e_m^b(n+1)}$	(8) $2m$
$b_m(n+1) = b_m(n) + w_m^a(n+1) e_m^b(n+1)$	(9) $m$
전체 계산량	$8m+2$

표 2-2. 5m FRLS 알고리즘

이득 시간 갱신	계산량
$e^f_m(n+1) = x(n+1) + a^f_m(n) x_m(n)$ (10)	m
$e^f_m(n+1) = e^f_m(n+1) / a_m(n)$ (11)	1
$a_m(n+1) = a_m(n) + w_m(n) e^f_m(n+1)$ (12)	m
$a^f_m(n+1) = a^f_m(n) + e^f_m(n+1) e^f_m(n+1)$ (13)	1
$w_{m+1}(n+1) = \begin{bmatrix} 0 \\ w_m(n) \end{bmatrix} - \frac{e^f_m(n+1)}{a^f_m(n+1)} \begin{bmatrix} 1 \\ a_m(n) \end{bmatrix}$ (14)	m+1
분할식 : $w_{m+1}(n+1) = \begin{bmatrix} \delta_m(n+1) \\ \delta_m(n+1) \end{bmatrix}$ (14)	0
$e^b_m(n+1) = -\delta_m(n+1) a^b_m(n)$ (16)	1
$w_m(n+1) = \delta_m(n+1) - \delta_m(n+1) b_m(n)$ (17)	m
$a_{m-1}(n+1) = a_m(n) + \frac{e^f_m(n+1)}{a^f_m(n)} e^f_m(n+1)$ (18)	1
$e^b_m(n+1) = e^b_m(n+1) / a_m(n+1)$ (19)	1
$a^b_m(n+1) = a^b_m(n) + e^b_m(n+1) e^b_m(n+1)$ (20)	1
$b_m(n+1) = b_m(n) + w_m(n+1) e^b_m(n+1)$ (21)	m
전체 계산량	5m+8

그러나, 잡음에 대한 감도를 증가시켜 시스템에 사용하기 위해서는 망각계수  $\lambda$ 를 1보다 작은 값으로 사용하게 되는데, 이때 FRLS는 수치적으로 불안정하여 정상상태에서는 급격히 발산하게 된다. (9)~(13) 이러한 문제점을 개선하기 위해 Normalization FRLS 알고리즘 (9), (10)과 배정도 계산을 사용할 것(12) 등이 제시 되기도 하였고, 특히 다음과 같이 보정 상수를 사용한 방법이 제시 되기도 하였다. (14)

표 2-1의 (4)식을 다음과 같이 수정하면

$$a^f_m(n+1) = a^f_m(n) \cdot e^f_m(n+1) \cdot k$$

이 된다. 즉 여기서 모형의 차수  $m$ 과 잡음의 크기 등에 따라 결정되는 보정상수  $k$ 를 이용하여 강인성을 개선하였다. 그렇지만 위의 어떠한 방법도 입력신호의 성질에 따라 강인성이 보장되지 않으므로 구조변수(rescue value)를 사용하여 이 변수를 감시하고 있다가 발산이 시작되면 이를 감지하고 알고리즘을 초기화하는 방법이 있다. 그러나 이러한 방법은 초기화시킬 때마다 상당한 양의 출력은 신뢰성이 없는 자료가 되어 수치적으로 불안하게 되므로 FRLS의 응용범위가 크게 제한받게 된다.

Ⅲ. 강인성이 보장된 고속 순차 최소자승알고리즘

FRLS가 강인성이 약한 이유는 계산량을 줄이기 위해 과거의 자료에 대한 정보를 행렬로 보관하지 않고 벡터로 처리하기 때문에 이로 인한 계산오차가 누적되기 때문이다. [2] 즉, 이론적으로는 8m형태와 5m형태 변수의 계산값이 일치해야 하나, 사용된 디지털 계산기의 어장에 의한 반올림오차 및 절단오차 등으로 인해 실제로는 일치하지 않는다. 이러한 점으로 인해 Botto 등은 이 오차를 최소화하는 보상방법을 제시하였다. (2), (3)

본 연구에서는 이론적으로 같아야 하는 이들 변수들에 적당한 값의 가중치를 줌으로써 상호 보완작용을 유도하여 오차의 누적현상을 막고, FRLS의 단점인 수치적 불안정성을 개선하여 강인성을 보완한 것이다. 즉, 전향예측기  $A_m(n)$ 와 후향예측기  $B_m(n)$ , 칼만이득  $w_m(n)$ 의 계산식이 8m형태와 5m형태에서 각기 다른 계산식을 갖는 것에 착안하여 표 2-1, 2-1에 있는 식을 다시 쓰면 다음과 같다.

$$e^f_m(n+1) = x(n+1) + a^f_m(n) x_m(n) \quad (3)$$

$$e^f_m(n+1) = e^f_m(n+1) / a_m(n) \quad (11)$$

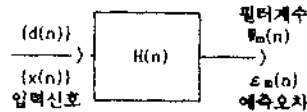
$$w_{m+1}(n+1) = \begin{bmatrix} 0 \\ w_m(n) \end{bmatrix} - \frac{e^f_m(n+1)}{a^f_m(n+1)} \begin{bmatrix} 1 \\ a_m(n) \end{bmatrix} \quad (5)$$

$$w_{m+1}(n+1) = \begin{bmatrix} 0 \\ w_m(n) \end{bmatrix} - \frac{e^f_m(n+1)}{a^f_m(n+1)} \begin{bmatrix} 1 \\ a_m(n) \end{bmatrix} \quad (14)$$

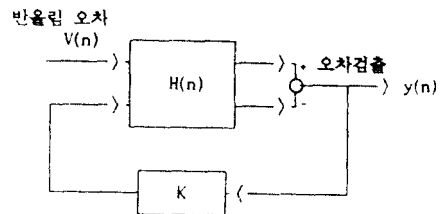
$$e^b_m(n+1) = x(n+1-m) \cdot b^b_m(n) x_m(n+1) \quad (7)$$

$$e^b_m(n+1) = -\delta_m(n+1) a^b_m(n) \quad (16)$$

만일 무한정도를 갖는 디지털 계산기를 사용한다면 각각 같은 물리량을 의미하는 위의 계산식 (1)과(2), (3)과(4), (5)와(6)이 각기 다른 방법으로 표현되었다 하더라도 그 결과는 일치할 것이다. 그러나 실제로 구현함에 있어서 그 결과는 다르게 되는데 그 이유는 수치 계산상의 오차도 볼 수있다. 그러므로 각기 다른 이들 계산결과는 오차 누적시스템의 출력으로 볼 수 있으므로, 본 논문에서는 오차의 누적현상을 막기 위해 그림 1.과 같은 피드백구조를 제안한다.



(a) FRLS 알고리즘



(b) 오차 귀환 시스템

그림 1. FRLS 알고리즘과 오차 귀환 시스템

또한, 귀환상수  $K$ 를 고려한 새로운 알고리즘의 변수들은 표 3.과 같다.

표 3. 귀환상수  $K$ 를 갖는 새로운 변수

$$\begin{aligned} e^f_m(n+1) &= K_3 e^f_m(n+1) \cdot (1-K_3) e^f_m(n+1) \\ w_m(n+1) &= K_4 w_m(n+1) \cdot (1-K_4) w_m(n+1) \\ e^b_m(n+1) &= K_{(i)} e^b_m(n+1) \cdot (1-K_{(i)}) e^b_m(n+1) \quad i=1, 2, 5 \end{aligned}$$

여기에서 특정한  $K$ 의 선택은 실험적 경험과 시뮬레이션으로 결정하게 되는데 일반적으로 최적  $K$ 의 선택은 망각계수  $\lambda$ 와 시스템 차수  $m$ , 신호의 특성에 따라 결정된다.

#### IV. 모의실험 및 결과

모의 실험에 사용한 입력신호는 각기 SNR=20[dB]와 14[dB]를 갖는 두개의 정현파를 이용했으며 잡음신호는 공분산이 1인 백색잡음(White Noise)을 이용하였다. 또한  $m=12$  이며  $\lambda=0.995$ 로 하였다. 또한 이득 K는 각각  $k_1=1.25, k_2=1.25, k_3=0.75, k_4=0.0, k_5=1.25$ 으로 하였다. 아래 그림에서 매개변수  $a_m(n)$ 의 추정 과정을 도시했다.

그림 2.는 입력신호이며 그림 3.은 매개변수의 최초수렴과정이고, 그림 5.와 그림 7.은 각각 1만, 2만번째 이산시간에서 매개변수가 발산하지 않고 그 값을 유지하고 있음을 보인것이다. 또한 그림 4.와 그림 6., 그림 8. 은 입력신호와 측정신호간의 오차로써 초기과정에서는 그 차이가 컸으나  $a(n)$ 이 수렴할 수록 그 차이가 줄어들고 있다.

시뮬레이션을 20만번째 이산시간에서도 시행한 결과 만족할 만한 결과를 얻을 수 있었다. 결론적으로 위에서 제시한 알고리즘이 강인성을 가지고 있음을 알 수있다.

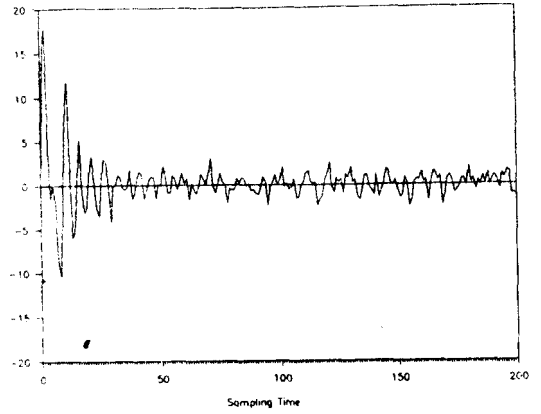


그림 4. 측정신호  $d(n)$ 과 예측신호  $y(n)$ 간의 오차

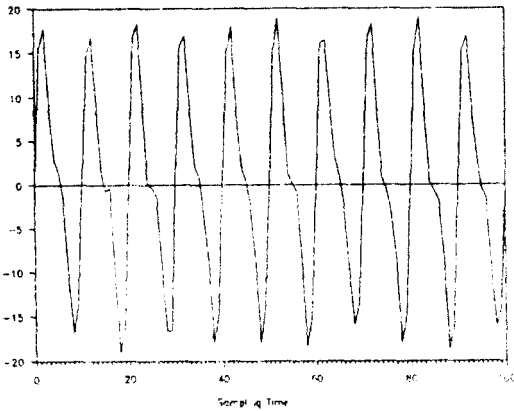


그림 2. 측정신호  $d(n)$

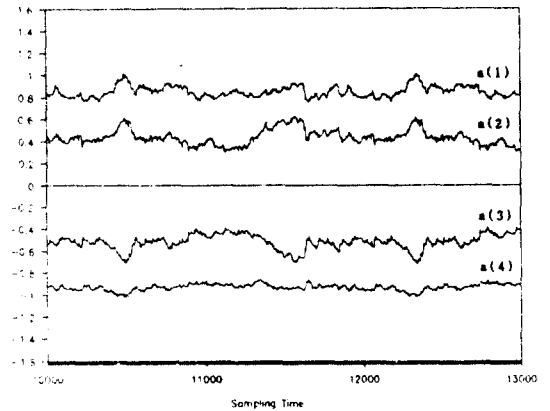


그림 5. 매개변수  $a(n)$ 의 1만번째 수렴특성

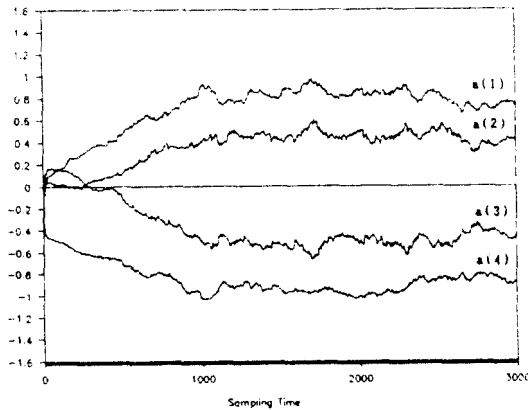


그림 3. 매개변수  $a(n)$ 의 초기 수렴특성

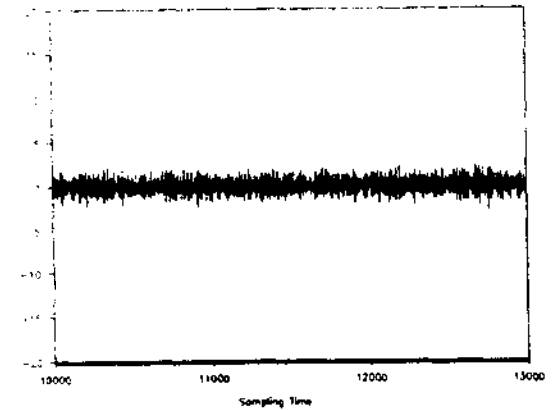


그림 6. 측정신호  $d(n)$ 과 예측신호  $y(n)$ 간의 오차

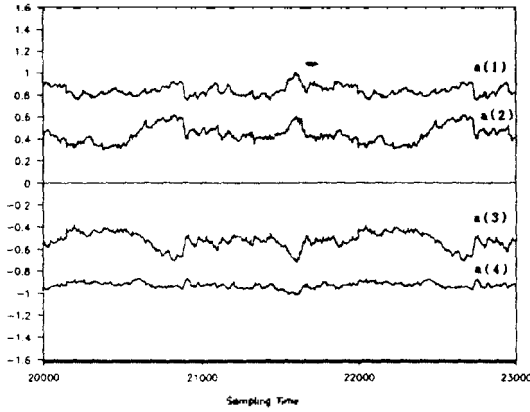


그림 7. 매개변수  $a(n)$ 의 2만번에 수렴특성

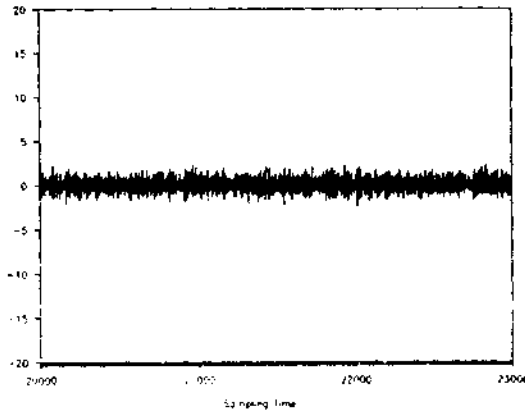


그림 8. 측정신호  $d(n)$ 과 예측신호  $y(n)$ 간의 오차

## V. 결론

착용 스펙트럼 추정에 많이 사용되는 AR 모형의 매개변수의 추정 방법중 FRLS 알고리즘은 적은 계산량과 빠른 수렴속도를 갖지만 수치적으로 불안정하므로 본 논문에서는 이를 개선하기 위해 귀환이득  $k$ 를 사용하여 이론적으로 같아야 하는 FRLS의 변수들에 적당한 값의 가중치를 줌으로써 오차의 누적 현상을 막고 FRLS와 단점인 수치적 불안정성을 개선하여 강인성을 보완하는 방안을 제시하였다. 제안된 알고리즘의 안정성을 확인하기 위하여 컴퓨터 시뮬레이션을 하였으며 좋은 결과를 얻었다.

## VI. 참고문헌

- [1] M. Bellanger, "Engineering aspects of fast least squares algorithms in transversal adaptive filters," in Proc. ICASSP 87 Conf. (Dallas, TX), Apr. 1987, pp. 2149-2152.
- [2] J. L. Botto and G. V. Moustakides, "Stabilization of fast Kalman algorithms," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-37, pp. 1342-1348, Sept. 1989.
- [3] J. L. Botto and G. V. Moustakides, "Stabilization of fast recursive least-squares transversal filters for adaptive filtering," in Proc. ICASSP 87 Conf. (Dallas, TX), Apr. 1987, pp. 403-407.
- [4] G. Carayannis, D. Manolakis, and N. Kalouptsidis, "A fast sequential algorithm for least filtering and prediction," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-31, no. 6, pp. 1394-1402, 1983.
- [5] J. M. Cioffi and T. Kailath, "Fast, recursive least squares transversal filters for adaptive filtering," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-32, no. 2, pp. 304-337, Apr. 1984.
- [6] L. Ljung, M. Morf, D. D. Falconer, "Fast calculation for gain matrices for recursive estimation schemes," Int. J. Control, Vol. 27, no. 1, 1978.
- [7] D. D. Falconer and L. Ljung, "Application of fast Kalman estimation to adaptive equalization," IEEE Trans. Commun., vol. COM-26, pp. 1439-1446, Oct. 1978.
- [8] C. Haikias, G. Carayannis, J. Dologlou, and D. Emmanouilpoulos, "A new generalized recursion for the fast computation of the Kalman gain to solve the covariance equations," in Proc. ICASSP 82, Paris, France, May 1982, pp. 1760-1763.
- [9] P. Fabre, C. Gueguen, "Improvement of the fast recursive least squares algorithms via normalization: A comparative study," IEEE Trans. on Acoustics, Speech, and Signal Processing, vol. ASSP-34, no. 2, April, 1986.
- [10] F. Ling, J. G. Proukis, "Numerical accuracy and stability of adaptive estimation algorithms caused by round-off error," Proc. ICASSP 84, San Diego.
- [11] S. Ljung, L. Ljung, "Error propagation properties of recursive least-squares adaptation algorithms," Automatica, Vol. 21, no. 2, 1985.
- [12] D. W. Lin, "On digital implementation of the fast kalman algorithms," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-32, no. 5, Oct. 1984.
- [13] D. T. M. Stock and T. Kailath, "Numerically stable fast transversal filters for recursive least squares adaptive filtering," IEEE Trans., Acoust., Speech, Signal Processing, vol. 39, no. 1, pp. 92-114, January, 1991.
- [14] 정찬수, "적용잡음제거시스템을 위한 고속 칼만 알고리즘에 관한 연구", 서울대학교 전기공학부, 1986. 박사학위논문
- [15] 정찬수 외 3명, "적용제어 기술에 관한 연구", 한국과학기술재단, 목적기초 연구사업 보고서, 1990.