

시변 시스템 추정 성능 개선을 위한 이중 슬라이딩 윈도우 재귀 최소 제곱 알고리즘

임 준 석*

A Dual Sliding Window Recursive Least Squares Algorithm to Improve Estimation Performance in Time-Varying System

Junseok Lim*

요 약

최소제곱(LS) 알고리즘은 회귀 분석 및 신호 처리 등 다양한 분야에서 널리 사용된다. 온라인 응용에서 망각 계수를 갖춘 재귀 최소제곱(RLS) 알고리즘은 과거 데이터의 영향을 지수적으로 줄여 시변 환경에서 좋은 성능을 발휘한다. 슬라이딩 윈도우 RLS (SW-RLS) 알고리즘은 시변성에 따라 윈도우 길이를 조절해야 좋은 결과를 얻을 수 있지만, 이에 적합한 방법은 아직 제안되지 않았다. 본 논문에서는 SW-RLS가 시변 시스템 환경에 맞춰 서로 다른 길이의 윈도우를 자동 선택하는 알고리즘을 제안한다. 그리고 제안한 알고리즘의 변화 추적 성능은 가변 망각 계수 RLS에 필적하며, 수렴 후 추정 성능은 충분히 긴 윈도우를 쓰는 SW-RLS와 같을뿐만 아니라 가변 망각 계수 RLS에 비해서 7dB 향상됨을 보인다.

Abstract

The Least Squares (LS) algorithm is common in fields like regression analysis and signal processing. For online applications in time-varying environments, the Recursive Least Squares (RLS) algorithm with a forgetting factor performs well by reducing past data's influence. While Sliding Window RLS (SW-RLS) can achieve good results by adjusting window length to time variations, a suitable adjustment method has been lacking. This paper introduces an SW-RLS algorithm that automatically selects varying window lengths based on the system environment. Its change tracking performance is comparable to variable forgetting RLS. After convergence, its estimation performance matches SW-RLS with long windows and is 7 dB superior to variable forgetting RLS.

Keywords

recursive least squares, sliding window recursive least squares, time-varying system estimation

* 세종대학교 전자정보통신공학과 교수
- ORCID: <https://orcid.org/0000-0003-1779-3484>

• Received: Mar. 28, 2025, Revised: May 22, 2025, Accepted: May 25, 2025
• Corresponding Author: Junseok Lim
Dept. of Electrical Engineering Sejong University Seoul Korea
Tel.: +82-2-3408-3299, Email: jslim@sejong.ac.kr

I. 서 론

최소제곱(LS) 알고리즘은 회귀 분석, 자동 제어 및 신호 처리 등 다양한 분야에서 널리 사용된다 [1]-[5]. 온라인 응용에서 망각 계수를 갖춘 재귀 최소제곱(RLS) 알고리즘은 구현의 간편성 덕분에 흔히 사용된다. 망각 계수 RLS는 과거 데이터의 영향을 지수적으로 줄여 최근 데이터에 더 큰 비중을 두어 시변 환경에서도 좋은 성능을 발휘한다[6]. 슬라이딩 윈도우 RLS (SW-RLS, Sliding Window RLS) 알고리즘은 유한 길이 데이터 윈도우를 사용하여 오래된 데이터를 폐기함으로써 시변성 환경에 적응하고, 일반적으로 더 우수한 수렴 후 추정 성능을 보인다[7]. 그러나 시변성이 일정하지 않고, 갑자기 변하거나 정상 상태에서부터 점진적으로 변하는 특성이 있다면 윈도우의 길이를 동적으로 조절해야 하지만, SW-RLS 및 유사 알고리즘들에선 적합한 방법이 아직 제안되지 않았다. 본 논문에서는 SW-RLS가 시변성 시스템 환경에 맞춰 길이가 서로 다른 윈도우를 자동 선택하는 알고리즘을 제안하며, 가변하는 채널에서 채널 추적성은 가변 망각 계수 RLS와 유사하고 수렴 후 추정 성능은 충분히 긴 윈도우를 쓰는 SW-RLS의 수렴 후 추정 성능과 유사함을 보인다.

II. 이중 윈도우 스위치 SW-RLS

2.1 SW-RLS

어떤 시스템이 식 (1)과 같이 모델링 될 때,

$$y_k = \mathbf{u}_k \mathbf{x}_k + v_k, \quad k \geq 0 \quad (1)$$

여기서 y_k 는 시스템의 출력 신호이고, x_k 는 시스템의 임펄스 응답이고, v_k 는 주변 잡음이다. 입력 신호로 구성된 입력 벡터는 $\mathbf{u}_k = [u_k, u_{k-1}, \dots, u_{k-N+1}]$ 이다.

슬라이딩 윈도우를 사용하면 오래된 데이터를 폐기하여 과거 데이터가 추정에 미치는 영향을 줄이며, RLS 알고리즘보다 더 나은 추적 성능을 보여준다. 식 (2)과 같이 유한 길이 L_s 로 윈도우된 지수 가

중 제곱 오차 함수가 있을 때,

$$\tilde{J}_k = \sum_{i=k-L_s+1}^k \rho^{k-i} (y_i - \mathbf{u}_i \mathbf{x}_k)^2 \quad (2)$$

여기서 ρ 는 망각 계수이다. 식 (2)을 x_k 에 대해 미분을 수행하고 결과를 0으로 설정하면, $\tilde{J}_k$ 를 최소화 하는 x_k 의 최적 추정값을 얻을 수 있다.

$$\hat{\mathbf{x}}_k = \tilde{\mathbf{R}}_k^{-1} \tilde{\boldsymbol{\gamma}}_k = \tilde{\mathbf{P}}_k \tilde{\boldsymbol{\gamma}}_k \quad (3)$$

여기서 $\tilde{\mathbf{R}}_k = \sum_{i=k-L_s+1}^k \rho^{k-i} \mathbf{u}_i^T \mathbf{u}_i$, $\tilde{\mathbf{R}}_k^{-1} = \tilde{\mathbf{P}}_k$ 이고

$\tilde{\boldsymbol{\gamma}}_k = \sum_{i=k-L_s+1}^k \rho^{k-i} \mathbf{u}_i^T y_i$ 이다. SW-RLS는 식 (4)으로 최신 데이터를 추가하고, 식 (5)으로 제일 오래된 데이터를 탈락시킨다. 이 두 과정을 통해서 매번 L_s 로 미리 정한 길이의 데이터 창을 유지하면서 상호 상관 행렬의 역행렬을 계산한다.

$$\tilde{\mathbf{P}}_k = (\rho \tilde{\mathbf{P}}_{k-1}^{-1} + \mathbf{u}_k^T \mathbf{u}_k)^{-1} \quad (4)$$

$$\tilde{\mathbf{P}}_k = (\tilde{\mathbf{P}}_k^{-1} - \rho^{L_s} \mathbf{u}_{k-L_s}^T \mathbf{u}_{k-L_s})^{-1} \quad (5)$$

그러므로 식 (4), 식 (5) 및 식 (3)을 번갈아 계산하면 SW-RLS가 완성된다[7].

2.2 이중 윈도우 SW-RLS를 위한 윈도우 선택 논리

단일 윈도우를 사용하는 일반 SW-RLS를 사용할 경우, 일반적으로 빠른 시변성 추적과 정상 상태에서 높은 정확도를 동시에 달성하기가 어렵다. 본 논문은 우수한 시변 시스템 추적과 높은 정확도를 동시에 달성하기 위해서 두 개의 다른 길이의 윈도우를 사용하는 SW-RLS 알고리즘을 제안한다. 제안한 알고리즘은 빠른 변화가 감지될 경우, 길이가 짧은 윈도우를 택하고, 변화가 매우 완만하거나 거의 없는 경우, 길이가 긴 윈도우를 택하게 하는 논리를 기존 SW-RLS에 결합한다.

윈도우 길이를 자동으로 조정하기 위한 척도로써 가변 망각 계수를 사용한다. 이는 가변 망각 계수가 시스템의 변화를 반영하도록 고안되었기 때문이다. 본 논문에서는 계산량이 적고 안정적 동작하는 Park[6]이 제안한 가변 망각 계수 $\rho(k) = \rho_{\max} \cdot 2^{-(\alpha |e(k)|^2)}$ (단 $|e(k)|^2$ 는 데이터 윈도우 내의 추정 오차 평균 에너지, $e(k) = y_k - \mathbf{u}_k^T \hat{\mathbf{x}}_k$, $\rho_{\max}$ 는 최대 망각 계수, α 는 스케일 상수)를 사용했다. 제안한 두 개의 데이터 윈도우를 운영하는 논리는 그림 1에 정리하였다. 2.1에 정리한 SW-RLS에 그림 1에 정리한 윈도우 선택 논리를 적용함으로써 시변 환경에 능동적으로 적응하는 이중 윈도우 SW-RLS가 될 수 있다. 제안한 알고리즘의 계산 복잡도를 알아보기 위해서 표 1을 보면 기존 SW-RLS는 그대로 사용하고 여기에 그림 1에 설명한 윈도우 선택 논리만이 추가됨을 알 수 있다. 따라서 부가되는 계산 복잡도를 곱셈 횟수로 생각하면, 가변 망각 계수 $\rho(k) = \rho_{\max} \cdot 2^{-(\alpha |e(k)|^2)}$ 를 계산하는 kN 번이 추가된다. 여기서 N 은 필터 차수이고 k 는 지수함수를 계산할 때 사용하는 테일러 전개식의 차수에 따라 달라진다.

따라서 $5N^2 + 8N + 6$ [7]의 곱셈을 사용하는 기존 SW-RLS에 대해서 상대적으로 적은 계산량 증가라고 할 수 있다. 표 1에 제안한 이중 윈도우 SW-RLS 알고리즘을 정리하였다.

표 1. 이중 윈도우 SW-RLS 알고리즘

Table 1. Dual window SW-RLS algorithm

Initialization: $\hat{\mathbf{x}}_{-1}, \tilde{\mathbf{P}}_{-1} = \epsilon_0 \mathbf{I}$ ($\epsilon_0 > 0$), forgetting factor: $0 < \rho \leq 1$, length of data window: $L_s = L_{long}$, number of data in current data window: $L_{data} = 0$.

Do $k = 1, 2, \dots$

$$\tilde{y}_{k-L_s} = y_{k-L_s} - \mathbf{u}_k^T \hat{\mathbf{x}}_{k-1}, \quad \tilde{y}_k = y_k - \mathbf{u}_k^T \hat{\mathbf{x}}_{k-1}$$

$$\tilde{\mathbf{P}}_k = \frac{1}{\rho} \left(\tilde{\mathbf{P}}_{k-1} - \frac{\tilde{\mathbf{P}}_{k-1} \mathbf{u}_k^T \mathbf{u}_k \tilde{\mathbf{P}}_{k-1}}{\rho + \mathbf{u}_k^T \tilde{\mathbf{P}}_{k-1} \mathbf{u}_k} \right)$$

$$\tilde{\mathbf{P}}_k = \tilde{\mathbf{P}}_k + \frac{\tilde{\mathbf{P}}_k \mathbf{u}_{k-L_s}^T \mathbf{u}_{k-L_s} \tilde{\mathbf{P}}_k}{1/\rho^{L_s} + \mathbf{u}_{k-L_s}^T \tilde{\mathbf{P}}_k \mathbf{u}_{k-L_s}}$$

Choose data window length from $L_s = L_{long}$ or $L_s = L_{short}$ according to Figure 1.

End Do:

III. 모의실험을 통한 성능 비교

3.1 급격한 채널 변화가 있는 시스템의 추정

본 실험에서는 Li[8] 등이 제안한 가변 망각 계수 논문에서와 같은 가변 채널 환경에서 여러 알고리즘의 성능을 서로 비교하였다.

설정된 가변 시스템은 임펄스 응답 $\mathbf{h}_1 = [1.0, 0.5, 2.0, 0.3, 0.7]$ 이 시간 스텝 500에서, $\mathbf{h}_2 = [-0.2, -0.5, -1.0, -0.4, -2.0]$ 으로 급격한 변화하는 환경에서 윈도우 길이가 200인 SW-RLS[7]와 윈도우 길이가 10인 SW-RLS[7] 및 가변 망각 계수를 사용하는 RLS[8]와 길이 200인 윈도우와 길이 10인 윈도우를 다 사용하는 제안된 알고리즘을 서로 비교하였다. 윈도우 선택에 사용되는 망각 계수 $\rho(k)$ 의 $\rho_{\max}$ 는 0.99로 설정하였다. 그리고 각 알고리즘을 신호 대 잡음 비 20dB와 5dB 환경에서 1000번씩 반복 실험하였다. 추정 성능을 알아보기 위해서 추정 성능의 척도는 추정 시스템의 평균 표준 편차(MSD, Mean Standard Deviation)를 사용하였다.

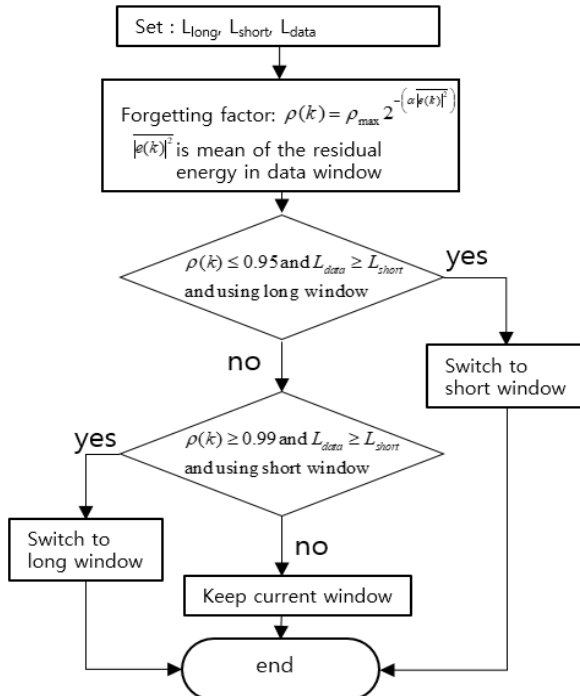
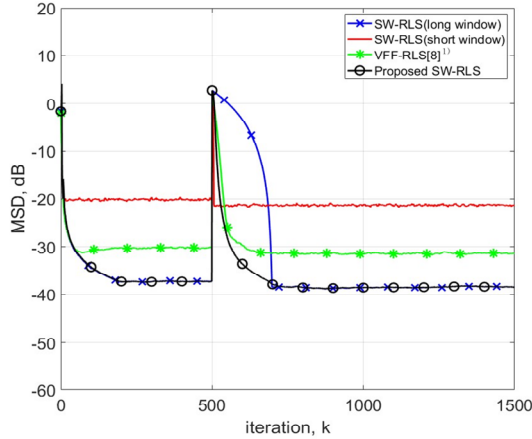


그림 1. 윈도우 선택 논리

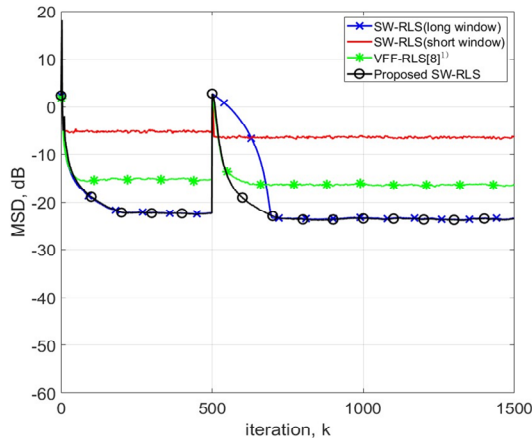
Fig. 1. Window selection logic

$$\text{MSD} = \text{Mean} \left(10 \log_{10} \frac{|\mathbf{x}_{\text{true}} - \mathbf{x}_{\text{estim}}|^2}{|\mathbf{x}_{\text{true}}|^2} \right) \quad (7)$$

여기서 x_{true} 는 시스템 파라미터 참값이고, x_{estim} 은 추정된 파라미터 값이다.



(a) 신호 대 잡음 비 20dB인 경우
(a) Case of SNR 20dB



(b) 신호 대 잡음 비 5dB인 경우
(b) Case of SNR 5dB

그림 2. 급변 시스템 경우에 MSD 비교

Fig. 2. MSD comparison results in the abruptly changed system case

* ¹⁾VFF-RLS (variable forgetting factor RLS)

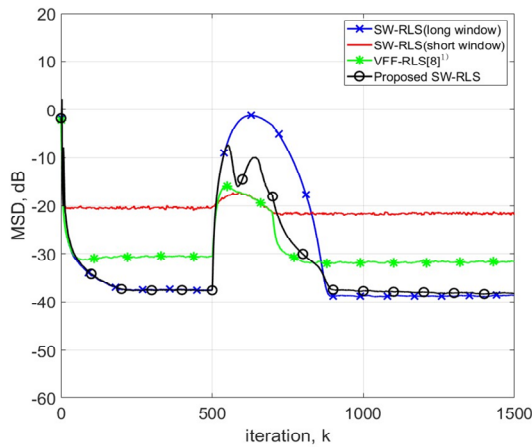
그림 2(a)는 신호 대 잡음 비가 20dB일 때 4가지 알고리즘의 시스템 추정 정확도를 비교한 결과이다. 짧은 윈도우 SW-RLS는 변화에 빠른 반응을 하는 한편 수렴 후 MSD가 -21dB로 가장 나빴다. 반면 긴 윈도우 SW-RLS는 변화에 가장 늦게 반응한 반면 수렴 후 MSD가 -38dB로 가장 좋았다. 그리고 가변 망각 계수를 사용하는 RLS[8]의 경우 짧은 윈

도우 SW-RLS만큼 빨리 변화에 반응하였고, MSD 면에서는 -31dB로 짧은 윈도우 SW-RLS보다는 좋고 긴 윈도우 SW-RLS보다는 나쁜 결과를 보였다. 이에 비해서 제안한 알고리즘은 가변 망각 계수를 사용하는 RLS만큼 빨리 변화에 반응하였고, 수렴 후 MSD도 -38dB가 되어 긴 윈도우 SW-RLS만큼 좋은 추정 성능을 보였다. 그림 2(b)는 신호 대 잡음 비가 5dB일 때 4가지 알고리즘의 시스템 인식 정확도를 비교한 결과이다. 이 경우에도 그림 2(a)와 같은 경향을 보임을 확인할 수 있다. 이와 같은 결과는 변화가 일어나는 부분에서 빠르게 창길이를 줄여서 짧은 윈도우 SW-RLS로 동작하다가 변화가 완만해지면 긴 윈도우 SW-RLS로 동작하도록 만든 제안한 알고리즘의 특성이 잘 나타났다고 할 수 있다.

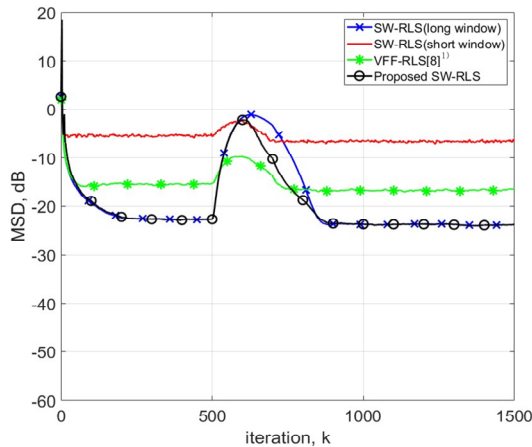
3.2 점진적인 채널 변화가 있는 시스템의 추정

본 비교 실험에서는 시스템의 임펄스 응답이 h_1 에서 h_2 로 500 스텝부터 700 스텝 사이에서 선형적으로 변하는 환경에서 기존의 SW-RLS와 가변 망각 계수를 사용하는 RLS[8] 및 제안한 알고리즘을 비교하였다. 나머지는 3.1절의 실험 조건과 같다. 본 실험은 가변 망각 계수 RLS에선 비교적 좋은 결과가 나오는 반면 기존의 단일 길이 윈도우 SW-RLS의 경우 좋은 성능을 얻기 어려운 실험 환경에 해당한다.

그림 3(a)은 신호 대 잡음 비 20dB일 때 4가지 알고리즘의 시스템 인식 정확도를 비교한 결과이다. 결과를 보면 채널이 시스템이 선형적으로 변동하는 구간에서, 긴 윈도우 SW-RLS는 변화에 매우 느리게 적응을 하고 있지만, 수렴 후 MSD는 -38dB였고, 짧은 윈도우 SW-RLS는 변화 구간에 거의 정확하게 맞춰서 적응하였지만, 수렴 후 MSD는 -21dB로 제일 나빴다. 가변 망각 계수 RLS는 짧은 윈도우 SW-RLS와 거의 유사한 변화 추적성을 보였고 수렴 후 MSD는 -31dB이 되었다. 제안한 알고리즘은 변화 추적 시작은 다른 알고리즘과 같고 종료 시점은 긴 윈도우 SW-RLS와 같아 보이지만, 가변 망각 계수와 같은 수렴 후 MSD를 기준으로 할 때, 추정 성능은 가변 망각 계수 RLS에 가깝다고 할 수 있다. 그러나 수렴 후 MSD는 약 -38dB로써 긴 윈도우 SW-RLS와 같은 성능을 보였다.



(a) 신호 대 잡음 비 20dB인 경우
(a) Case of SNR 20dB



(b) 신호 대 잡음 비 5dB인 경우
(b) Case of SNR 5dB

그림 3. 점진적 변화 시스템 경우에 MSD 비교

Fig. 3. MSD comparison results in the gradually changed system case

* ¹⁾VFF-RLS (variable forgetting factor RLS)

그림 3(b)는 신호 대 잡음 비가 5dB일 때 4가지 알고리즘의 시스템 인식 정확도를 비교한 결과이다. 이 경우에도 그림 3(a)과 같은 경향을 보임을 확인할 수 있다. 점진적인 채널 변화 환경에서 시스템 추정 실험에서도 제안한 알고리즘이 빠른 변화 추적 성능 및 우수한 수렴 후 추정 정확도를 제공한다는 것을 확인할 수 있었다. 그러나 점진적 변화 구간에서 추적 성능이 짧은 윈도우 SW-RLS와 똑같이 동작하지 않은 점은 채널이 점진적으로 변화하기 때문에 그림 1의 알고리즘상에서 긴 윈도우 SW-RLS와 짧은 윈도우 SW-RLS가 수시로 바뀌는 현상이 일어나기 때문으로 분석된다. 그러나 그림 3

전체 결과를 보면 제안된 방법이 점진적으로 변화하는 시스템 환경에서도 변화 구간 내에서 짧은 윈도우와 긴 윈도우 간에 스위칭을 통하여 기존 단일 윈도우를 사용하는 SW-RLS의 취약점을 충분히 만회하고 있음을 확인할 수 있다.

IV. 결 론

본 논문에서는 짧고 긴 두 개의 고정 길이 데이터 윈도우를 사용하는 SW-RLS를 제안하였다. 제안한 알고리즘은 시스템 변환에 맞춰서 두 데이터 윈도우 중 적당한 것을 선택할 수 있도록 만들었다. 이런 알고리즘 개선을 통해서 기존 SW-RLS보다 시변 시스템 추정에서 더 우수한 성능을 얻을 수 있음을 확인하였다. 즉 제안한 알고리즘의 변화 추적 성능은 가변 망각 계수 RLS와 유사하며, 수렴 후 추정 성능은 충분히 긴 윈도우를 쓰는 SW-RLS와 같을 뿐만 아니라 가변 망각 계수 RLS의 추정 성능보다 7dB 향상됨을 보였다.

References

- [1] C. Wu, X. Wang, and Y. Guo, "Robust uncertainty control of the simplified Kalman filter for acoustic echo cancelation", *Circuits Syst. Signal Process.*, Vol. 35, No. 12, pp. 4584-4595, Dec. 2016. <https://doi.org/10.1007/s00034-016-0263-1>.
- [2] Y. Liu and W. B. Mikhael, "A fast-converging adaptive fir technique for channel equalization", *Proc. IEEE 55th International Midwest Symposium on Circuits and Systems (MWSCAS)*, Boise, ID, USA, pp. 828-831, Aug. 2012. <https://doi.org/10.1109/MWSCAS.2012.6292148>.
- [3] C. Auer, T. Paireder, and M. Huemer, "Self-interference cancellation in LTE/5G transceivers with sliding window kernel recursive least squares filters", *Proc. 2021 55th Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, USA, pp. 976-982, Oct. 2021. <https://doi.org/10.1109/IEEECONF53345.2021.9723127>.

- [4] L. Shen, Y. Zakharov, B. Henson, N. Morozs, and P. Mitchel, "Adaptive Filtering for Full-Duplex UWA Systems With Time-Varying Self-Interference Channel", IEEE Access, Vol. 8, pp. 187590-187604, Oct. 2020. <https://doi.org/10.1109/ACCESS.2020.3031010>.
- [5] L. Shuhua and Q. Zhi, "An Improved Strong Tracking Variable Forgetting Factor RLS Algorithm with Low Complexity for Dynamic System Identification", Proc. 2019 IEEE 2nd International Conference on Electronics and Communication Engineering, Xi'an, China, pp. 444-450, Dec. 2019. <https://doi.org/10.1109/ICECE48499.2019.9058504>.
- [6] D. Park, B. Jun, and J. Kim, "Fast tracking RLS Algorithm using novel variable forgetting factor with unity zone", Electronics Letters, Vol. 27 No. 23, pp. 2150-2151, Nov. 1991. <https://doi.org/10.1049/el:19911331>.
- [7] K. Nishiyama, "A New Formalism of the Sliding Window Recursive Least Squares Algorithm and Its Fast Version", IEICE Trans. Fundamentals, Vol. E94-A, No. 6 pp. 1394-1400, Jun. 2011. <https://doi.org/10.1587/transfun.E94.A.1394>.
- [8] K. Li, J. Xiao, J. Xie, and R. Wu, "A novel adaptive variable forgetting factor RLS algorithm", Proc. 2022 International Conference on Informatics, Networking and Computing, Nanjing, China, pp. 228-232, Oct. 2022. <https://doi.org/10.1109/ICINC58035.2022.00053>.

저자소개

임 준 석 (Junseok Lim)



1986년 2월 : 서울대학교

전자공학과(공학사)

1988년 2월 : 서울대학교

전자공학과(공학석사)

1996년 8월 : 서울대학교

전자공학과(공학박사)

1998년 3월 ~ 현재 : 세종대학교

전자정보통신공학과 교수

관심분야 : 신호처리