

## Article

# A Modified Recursive Regularization Factor Calculation for Sparse RLS Algorithm with $l_1$ -Norm

Junseok Lim <sup>1</sup>, Keunhwa Lee <sup>2</sup> and Seokjin Lee <sup>3,\*</sup>

<sup>1</sup> Department of Electrical Engineering, College of Electronics and Information Engineering, Sejong University, Gwangjin-gu, Seoul 05006, Korea; jslim@sejong.ac.kr

<sup>2</sup> Department of Defense Systems Engineering, College of Engineering, Sejong University, Gwangjin-gu, Seoul 05006, Korea; nasalkh2@sejong.ac.kr

<sup>3</sup> School of Electronics Engineering, School of Electronic and Electrical Engineering, College of IT Engineering, Kyungpook National University, Daegu 41566, Korea

\* Correspondence: sjlee6@knu.ac.kr; Tel.: +82-53-950-5523

**Abstract:** In this paper, we propose a new calculation method for the regularization factor in sparse recursive least squares (SRLS) with  $l_1$ -norm penalty. The proposed regularization factor requires no prior knowledge of the actual system impulse response, and it also reduces computational complexity by about half. In the simulation, we use Mean Square Deviation (MSD) to evaluate the performance of SRLS, using the proposed regularization factor. The simulation results demonstrate that SRLS using the proposed regularization factor calculation shows a difference of less than 2 dB in MSD from SRLS, using the conventional regularization factor with a true system impulse response. Therefore, it is confirmed that the performance of the proposed method is very similar to that of the existing method, even with half the computational complexity.

**Keywords:** sparse impulse response system; sparse system estimation;  $l_1$ -RLS; regularization factor



**Citation:** Lim, J.; Lee, K.; Lee, S. A Modified Recursive Regularization Factor Calculation for Sparse RLS Algorithm with  $l_1$ -Norm. *Mathematics* **2021**, *9*, 1580. <https://doi.org/10.3390/math9131580>

Academic Editors: Annalisa Pascarella and Francesca Pitolli

Received: 14 June 2021  
Accepted: 3 July 2021  
Published: 5 July 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

The sparse channel system refers to a model characterized by only a few numbers of non-zero taps when modeling the impulse response of the channel system, using the tapped-delay-line model. We can experience such channels in TV channels [1], wide-band radio communication [2], underwater sound channels [3], etc. [4]. There have been many recent works on the use of adaptive estimation for sparse channel estimation; subsequently, various LMS-type algorithms [5,6] such as sparse LMF algorithms [7,8] and sparse LMS/F algorithms [9,10] have been proposed for sparse channel estimation. The RLS algorithm is better suited to fast convergence rates than the LMS-type algorithm. Hence, the RLS-based sparse adaptive filtering algorithm is considered one of the most promising fast algorithms in many system estimation applications, such as channel estimation. There are some algorithms based on sparse RLS [11–14] as well as the Total Least Squares (TLS)-based algorithm [15,16]. Many sparse RLS algorithms are algorithmically comparable to the plain RLS algorithm; however, the updated equations in most sparse RLS (SRLS) algorithms are not intrinsically recursive, as in the plain RLS algorithm. Eksioğlu and Tanc [13] proposed a fully recursive SRLS algorithm that is comparable to the plain RLS algorithm. The sparse RLS algorithms handle the sparsity with  $l_1$ -norm. Therefore, it is essential to properly select the regularization factor for  $l_1$ -norm. Many researchers have developed as well as proposed a proper selection for the regularization factor. The authors in [13] also proposed a regularization factor calculation method for the SRLS algorithm. Similar recursive regularization factor selection methods were used in [15–19]. However, the algorithm from [15–18] is not practical because the regularization factor in [15–17] assumed that the true system impulse response is known in advance and set the true system response in part to an arbitrary constant in [18]. The regularization factor selection

method in [19] needs no true system impulse response. However, the regularization factor is recursively updated. Therefore, errors in updating the regularization factor are likely to propagate and accumulate. In [20], Lim proposed the regularization factor by estimating the sparsity of the estimated system. Although the regularization factor required no true system information, it added error to the system. Then Lim in [20] utilized it only to TLS-based system modeling. In [21],  $l_1$ -IWF (iterative Wiener filter) was proposed, which was a kind of a steepest descent method.  $l_1$ -IWF had a regularization factor without requiring a priori knowledge of the true system response. However, it did not show whether the regularization factor converges to the optimal regularization factor.

In this paper, the major contribution is that we propose a new regularization factor for the SRLS algorithm in [13] and show that it converges to the scaled optimal regularization factor. The proposed regularization factor does not require a priori knowledge of the true system response. The minor contribution is that the proposed regularization factor requires less computation complexity than the regularization in [13].

The remainder of the paper is organized as follows. In Section 2, we reformulate the sparse RLS. In Section 3, a new regularization factor is proposed. In this section, we show that the regularization factor in [13] requires a priori knowledge of the true system impulse response. We also show the calculation complexity of the proposed regularization factor. The simulation condition and results are described and illustrated in Section 4, and a discussion of the results follows in Section 5. We conclude in Section 6.

## 2. Problem Formulation

In this section, we reformulate the SRLS in [13]. Consider a sparse weight vector  $\mathbf{w}_o \in \mathbb{R}^N$ , which represents the channel by a delay line with  $N$  taps. By sparsity, the number of significant factors in  $\mathbf{w}_o$ ,  $S$ , is much lower than its total dimensionality (that is,  $S \ll N$ ). The goal is to derive the sparse vector  $\mathbf{w}_o$  based on the input signal vector  $\mathbf{x}(n) \in \mathbb{R}^N$  and the received signal  $y(n)$ , which is assumed to be generated by a linear system, such as (1).

$$y(n) = \mathbf{w}_o^T \mathbf{x}(n) + \eta(n), \tag{1}$$

where  $\eta(n)$  is the additive noise. Consider the following standard RLS optimization problem, subject to a sparsity constraint.

$$\text{minimize}(\hat{\mathbf{w}}(n)) = \frac{1}{2} \sum_{m=0}^n \lambda^{n-m} e(m)^2, \text{ s.t. } \|\hat{\mathbf{w}}(n)\|_1 \leq c, \tag{2}$$

where  $e(m) = y(m) - \hat{\mathbf{w}}(n)^T \mathbf{x}(m)$ ,  $\hat{\mathbf{w}}(n) = [\hat{w}_1(n), \dots, \hat{w}_N(n)]^T$ ,  $\mathbf{x}(m) = [x(m), x(m-1), \dots, x(m-N+1)]^T$ ,  $\lambda$  is a forgetting factor, and  $\|\hat{\mathbf{w}}(n)\|_1 = \sum_{k=0}^{N-1} |\hat{w}_k(n)|$ . As the reformulation in [22], (2) becomes the following:

$$J(\hat{\mathbf{w}}(n), \gamma(n)) = \frac{1}{2} \zeta(\hat{\mathbf{w}}(n)) + \gamma(n)(\|\hat{\mathbf{w}}(n)\|_1 - c), \tag{3}$$

where  $\gamma(n)$  is a real-valued Lagrangian multiplier. We can find the optimal vector by minimizing the regularized cost function (3). The regularized cost function is convex and nondifferentiable; therefore, subgradient analysis is used instead of the normal gradient. When representing a subgradient of  $f$  at  $\hat{\mathbf{w}}$  with  $\nabla^s f(\hat{\mathbf{w}})$ , the subgradient of  $J(\hat{\mathbf{w}}(n), \gamma(n))$  with respect to  $\hat{\mathbf{w}}(n)$  is as follows.

$$\nabla^s J(\hat{\mathbf{w}}(n), \gamma(n)) = \frac{1}{2} \nabla \zeta(\hat{\mathbf{w}}(n)) + \gamma(n) \nabla^s (\|\hat{\mathbf{w}}(n)\|_1), \tag{4}$$

where  $\nabla^s (\|\hat{\mathbf{w}}(n)\|_1) = \text{sgn}(\hat{\mathbf{w}}(n))$  [13]. Here,  $\text{sgn}(\cdot)$  means the element-wise sign function. Hence, for the optimal  $\hat{\mathbf{w}}(n)$  for  $J(\hat{\mathbf{w}}(n), \gamma(n))$ , we set (4) to  $\mathbf{0}$  at the optimal point.

After evaluating the gradient  $\nabla\zeta(\hat{\mathbf{w}}(n))$ , (4) can be written as normal equations in matrix form as follows [13].

$$\Phi(n)\hat{\mathbf{w}}(n) = \mathbf{r}(n) - \gamma(n)\nabla^s(\|\hat{\mathbf{w}}(n)\|_1), \tag{5}$$

where  $\Phi(n) = \sum_{m=0}^n \lambda^{n-m}\mathbf{x}(m)\mathbf{x}(m)^T = \lambda\Phi(n-1) + \mathbf{x}(n)\mathbf{x}(n)^T$  and  $\mathbf{r}(n) = \sum_{m=0}^n \lambda^{n-m}y(m)\mathbf{x}(m) = \lambda\mathbf{r}(n-1) + y(n)\mathbf{x}(n)$ . With  $\mathbf{P}(n) = \Phi^{-1}(n) = (\lambda\Phi(n-1) + \mathbf{x}(n)\mathbf{x}(n)^T)^{-1}$  and using the matrix inversion lemma, we obtain the following:

$$\mathbf{P}(n) = \lambda^{-1}(\mathbf{P}(n-1) - \mathbf{k}(n)\mathbf{x}(n)^T\mathbf{P}(n-1)), \tag{6}$$

where  $\mathbf{k}(n) = \mathbf{P}(n-1)\mathbf{x}(n)/(\lambda + \mathbf{x}(n)^T\mathbf{P}(n-1)\mathbf{x}(n))$  is the gain vector. Assuming that  $\gamma(n-1)$  and  $\nabla^s\|\hat{\mathbf{w}}(n-1)\|_1$  do not change considerably over a single time step, the updated equation of  $\hat{\mathbf{w}}(n)$  can be approximately written as follows [13]:

$$\begin{aligned} \hat{\mathbf{w}}(n) &\cong \hat{\mathbf{w}}(n-1) + \mathbf{k}(n)\xi(n) - (1-\lambda)\gamma(n-1)\mathbf{P}(n)\nabla^s\|\hat{\mathbf{w}}(n-1)\|_1 \\ &= \hat{\mathbf{w}}(n-1) + \mathbf{k}(n)\xi(n) + (1-\lambda)\hat{\gamma}(n-1)\mathbf{P}(n)\nabla^s\|\hat{\mathbf{w}}(n-1)\|_1, \end{aligned} \tag{7}$$

where  $\hat{\gamma}(n) = -\gamma(n)$  and  $\xi(n) = y(n) - \hat{\mathbf{w}}(n-1)^T\mathbf{x}(n)$ .

Algorithm 1 summarizes the conventional RLS algorithm and Algorithm 2 summarizes the  $l_1$ -norm sparse RLS algorithm.

---

**Algorithm 1** conventional RLS algorithm

---

- 1: Initialize:  $\lambda, \mathbf{x}(0), y(0), \mathbf{w}_{RLS}(0) = \mathbf{0}, \eta, \mathbf{P}(0) = \eta^{-1}\mathbf{I}$
  - 2: **for**  $n = 1, 2, \dots$  **do**
  - 3:      $\mathbf{k}(n) = \mathbf{P}(n-1)\mathbf{x}(n)/(\lambda + \mathbf{x}(n)^T\mathbf{P}(n-1)\mathbf{x}(n))$
  - 4:      $e(n) = y(n) - \mathbf{w}_{RLS}(n-1)^T\mathbf{x}(n)$
  - 5:      $\mathbf{P}(n) = \lambda^{-1}(\mathbf{P}(n-1) - \mathbf{k}(n)\mathbf{x}(n)^T\mathbf{P}(n-1))$
  - 6:      $\mathbf{w}_{RLS}(n) = \mathbf{w}_{RLS}(n-1) + \mathbf{k}(n)e(n)$
  - 7: **end for**
- 

---

**Algorithm 2**  $l_1$ -norm sparse RLS algorithm

---

- 1: Initialize:  $\lambda, \mathbf{x}(0), y(0), \hat{\mathbf{w}}(0) = \mathbf{0}, \eta, \mathbf{P}(0) = \eta^{-1}\mathbf{I}$
  - 2: **for**  $n = 1, 2, \dots$  **do**
  - 3:      $\mathbf{k}(n) = \mathbf{P}(n-1)\mathbf{x}(n)/(\lambda + \mathbf{x}(n)^T\mathbf{P}(n-1)\mathbf{x}(n))$
  - 4:      $\xi(n) = y(n) - \hat{\mathbf{w}}(n-1)^T\mathbf{x}(n)$
  - 5:      $\mathbf{P}(n) = \lambda^{-1}(\mathbf{P}(n-1) - \mathbf{k}(n)\mathbf{x}(n)^T\mathbf{P}(n-1))$
  - 6:     calculate  $\hat{\gamma}(n-1)$
  - 7:      $\hat{\mathbf{w}}(n) = \hat{\mathbf{w}}(n-1) + \mathbf{k}(n)\xi(n) + (1-\lambda)\hat{\gamma}(n-1)\mathbf{P}(n)\nabla^s\|\hat{\mathbf{w}}(n-1)\|_1$
  - 8: **end for**
- 

**3. Proposed Recursive Regularization Factor for Sparse RLS Algorithm**

In this section, we derive the regularization factor,  $\hat{\gamma}(n-1)$ , such that  $\|\hat{\mathbf{w}}(n-1)\|_1 = c$ , which means that the  $l_1$ -norm of  $\hat{\mathbf{w}}(n)$  is preserved for all the time steps in  $n$ . This property yields the time invariance in  $\hat{\mathbf{w}}(n)$  [23].

$$\begin{aligned} \frac{d\|\hat{\mathbf{w}}(n)\|_1}{dn} &= \left(\frac{\partial\|\hat{\mathbf{w}}(n)\|_1}{\partial\hat{\mathbf{w}}(n)}\right)^T \frac{d\hat{\mathbf{w}}(n)}{dn} = (\nabla^s\|\hat{\mathbf{w}}(n)\|_1)^T \left(\frac{\hat{\mathbf{w}}(n) - \hat{\mathbf{w}}(n-T_s)}{T_s}\right) \\ &\simeq (\nabla^s\|\hat{\mathbf{w}}(n-1)\|_1)^T \left(\frac{\hat{\mathbf{w}}(n) - \hat{\mathbf{w}}(n-T_s)}{T_s}\right) = 0. \end{aligned} \tag{8}$$

By referring to (7) and assuming a normalized sample time  $T_s$  to 1 [24], (8) is as follows.

$$(\nabla^s\|\hat{\mathbf{w}}(n-1)\|_1)^T(\mathbf{k}(n)\xi(n) + (1-\lambda)\hat{\gamma}(n-1)\mathbf{P}(n)\nabla^s\|\hat{\mathbf{w}}(n-1)\|_1) = 0. \tag{9}$$

Hence,  $\hat{\gamma}(n - 1)$  in (9) becomes the following:

$$\hat{\gamma}(n - 1) = -\frac{(\nabla^s \|\hat{\mathbf{w}}(n - 1)\|_1)^T \mathbf{k}(n) \xi(n)}{(1 - \lambda)(\nabla^s \|\hat{\mathbf{w}}(n - 1)\|_1)^T \mathbf{P}(n) \nabla^s \|\hat{\mathbf{w}}(n - 1)\|_1}. \tag{10}$$

Eksioglu derived the upper bound of the regularization parameter  $\hat{\gamma}(n)$  in [13] as follows:

$$\bar{\gamma}(n) = 2 \frac{(\nabla^s \|\hat{\mathbf{w}}(n)\|_1)^T \mathbf{P}(n) \tilde{\epsilon}(n)}{\|\mathbf{P}(n) \nabla^s \|\hat{\mathbf{w}}(n)\|_1\|_2^2}, \tag{11}$$

where  $\tilde{\epsilon}(n) = \hat{\mathbf{w}}(n) - \mathbf{w}_o$  and  $\hat{\mathbf{w}}(n)$  is the solution to the conventional non-regularized normal equation as  $\hat{\mathbf{w}}(n) = \mathbf{P}(n)\mathbf{r}(n)$  [25]. In (11), the true system impulse response,  $\mathbf{w}_o$ , must be known in advance for  $\tilde{\epsilon}(n) = \hat{\mathbf{w}}(n) - \mathbf{w}_o$ .

If we assume that  $x(n)$  is a white noise process, its autocorrelation matrix,  $\Phi(n)$ , asymptotically becomes  $\sigma_x^2 \mathbf{I}$  and then  $\mathbf{P}(n) = \Phi(n)^{-1} \cong \sigma_x^{-2}(1 - \lambda)\mathbf{I}$  [25]. Therefore, (11) asymptotically becomes the following:

$$\bar{\gamma}(n) \cong \frac{2\sigma_x^2}{(1 - \lambda)} \frac{(\nabla^s \|\hat{\mathbf{w}}(n)\|_1)^T \tilde{\epsilon}(n)}{\|\nabla^s \|\hat{\mathbf{w}}(n)\|_1\|_2^2}. \tag{12}$$

Eksioglu also showed that, if the regularization parameter  $\hat{\gamma}(n)$  is  $0 \leq \hat{\gamma}(n) \leq \bar{\gamma}(n)$ , the following inequality is satisfied.

$$\|\hat{\mathbf{w}}(n) - \mathbf{w}_o\|_2^2 \leq \|\hat{\mathbf{w}}(n) - \mathbf{w}_o\|_2^2. \tag{13}$$

This means that the estimation error with regularization is lower than the estimation error without regularization.

In order to review the asymptotic meaning of the proposed regularization in (10), we also assume  $x(n)$  is a white noise process whose autocorrelation matrix,  $\Phi(n)$ , asymptotically becomes  $\sigma_x^2 \mathbf{I}$  and then  $\mathbf{P}(n) = \Phi(n)^{-1} \cong \sigma_x^{-2}(1 - \lambda)\mathbf{I}$  [25]. We use the simplified gain vector  $\mathbf{k}(n)$  in [25].

$$\mathbf{k}(n) = \mathbf{P}(n)\mathbf{x}(n), \tag{14}$$

and approximate the error,  $\xi(n)$ , in high SNR,

$$\begin{aligned} \xi(n) &= y(n) - \mathbf{x}^T(n)\hat{\mathbf{w}}(n - 1) \cong \mathbf{x}^T(n)\mathbf{w}_o - \mathbf{x}^T(n)\hat{\mathbf{w}}(n - 1) \\ &= -\mathbf{x}^T(n)(\hat{\mathbf{w}}(n - 1) - \mathbf{w}_o) \cong -\mathbf{x}^T(n)(\hat{\mathbf{w}}(n - 1) - \mathbf{w}_o). \end{aligned} \tag{15}$$

When we substitute (10) with (14) and (15), we can rewrite the proposed regularization in (10) as follows:

$$\hat{\gamma}(n - 1) = \frac{(\nabla^s \|\hat{\mathbf{w}}(n - 1)\|_1)^T \mathbf{P}(n)\mathbf{x}(n)\mathbf{x}^T(n)(\hat{\mathbf{w}}(n - 1) - \mathbf{w}_o)}{(1 - \lambda)(\nabla^s \|\hat{\mathbf{w}}(n - 1)\|_1)^T \mathbf{P}(n) \nabla^s \|\hat{\mathbf{w}}(n - 1)\|_1}. \tag{16}$$

In (16),  $\mathbf{P}(n)\mathbf{x}(n)\mathbf{x}^T(n)$  can be asymptotically approximated as the expected value of (17) in [26].

$$E(\mathbf{P}(n)\mathbf{x}(n)\mathbf{x}^T(n)) \cong \frac{(1 - \lambda)}{(1 - \lambda^{k+1})} \mathbf{I} \cong (1 - \lambda)\mathbf{I}. \tag{17}$$

Substituting (16) with (17) and  $\mathbf{P}(n) \cong \sigma_x^{-2}(1 - \lambda)\mathbf{I}$ , the variable regularization factor in (16) becomes the following:

$$\hat{\gamma}(n - 1) \cong \frac{\sigma_x^2}{1 - \lambda} \frac{(\|\hat{\mathbf{w}}(n - 1)\|_1)^T \tilde{\epsilon}(n - 1)}{\|\nabla^s \|\hat{\mathbf{w}}(n - 1)\|_1\|_2^2} = \frac{1}{2} \bar{\gamma}(n - 1), \tag{18}$$

$$\hat{\gamma}(n-1) = \frac{1}{2} \bar{\gamma}(n-1). \tag{19}$$

In (19), the proposed regularization parameter is approximately converse to the scaled optimal regularization parameter in [13]. Therefore, the newly derived regularization parameter  $\hat{\gamma}(n)$  in (19) satisfies  $0 \leq \hat{\gamma}(n) \leq \bar{\gamma}(n)$ . Therefore, the proposed regularization parameter can be used as a regularization parameter on behalf of the optimal regularization parameter. In addition, the proposed regularization parameter needs no true system parameter  $\mathbf{w}_o$ .

In terms of computational complexity, the difference between SRLS in [13] and the proposed algorithm lies only in the computational complexity of the regularization factor. Therefore, we can compare the computational complexity between the proposed regularization factor in (10) and the regularization factor in (11) from [13]. Actually, the regularization factor used in [13] is (20), which is an approximation of (11).

$$\begin{aligned} \hat{\gamma}(n-1) &= 2 \left( \frac{\text{tr}(\mathbf{P}(n-1))}{N} \frac{(f(\hat{\mathbf{w}}(n-1)) - \mathbf{w}_o)}{\|\mathbf{P}(n-1)\nabla^s f(\hat{\mathbf{w}}(n-1))\|_2^2} + \frac{\nabla^s f(\hat{\mathbf{w}}(n-1))\mathbf{P}(n-1)(\hat{\mathbf{w}}(n) - \hat{\mathbf{w}}_{RLS}(n-1))}{\|\mathbf{P}(n-1)\nabla^s f(\hat{\mathbf{w}}(n-1))\|_2^2} \right) \\ &= 2 \left( \frac{\text{tr}(\mathbf{P}(n-1))}{N} \frac{(\|\hat{\mathbf{w}}(n-1)\|_1 - \|\mathbf{w}_o\|_1)}{\|\mathbf{P}(n-1)\text{sgn}(\hat{\mathbf{w}}(n-1))\|_2^2} + \frac{\text{sgn}(\hat{\mathbf{w}}(n-1))\mathbf{P}(n-1)(\hat{\mathbf{w}}(n) - \hat{\mathbf{w}}_{RLS}(n-1))}{\|\mathbf{P}(n-1)\text{sgn}(\hat{\mathbf{w}}(n-1))\|_2^2} \right), \end{aligned} \tag{20}$$

where  $\mathbf{w}_o$  is the true system response,  $f(\hat{\mathbf{w}}(n)) = \|\hat{\mathbf{w}}(n)\|_1$ ,  $\nabla^s \|\hat{\mathbf{w}}\|_1 = \text{sgn}(\hat{\mathbf{w}})$  and  $\hat{\mathbf{w}}_{RLS}(n)$  is a solution from the conventional RLS algorithm [13]. And applying  $\nabla^s \|\hat{\mathbf{w}}\|_1 = \text{sgn}(\hat{\mathbf{w}})$  to the proposed regularization factor (10), it becomes as follows:

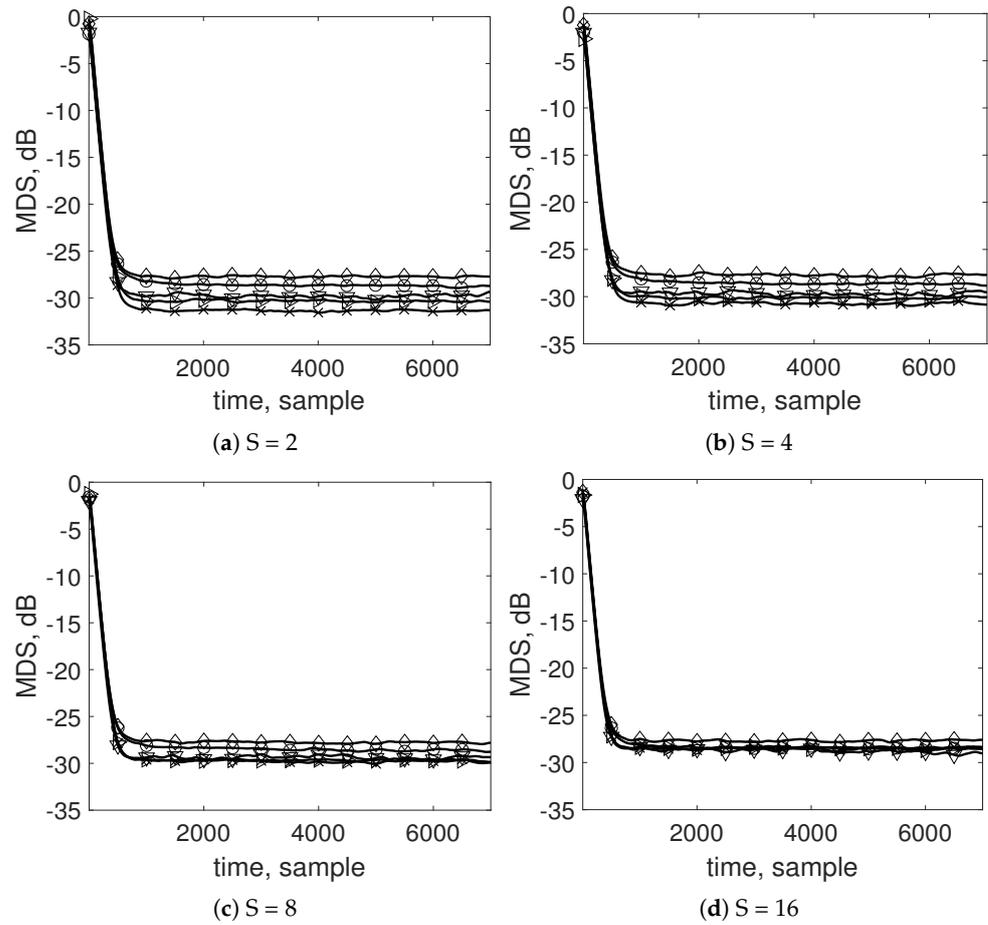
$$\begin{aligned} \hat{\gamma}(n-1) &= - \frac{\nabla^s f(\hat{\mathbf{w}}(n-1))^T \mathbf{k}(n) \zeta(n)}{(1-\lambda) \nabla^s f(\hat{\mathbf{w}}(n-1))^T \mathbf{P}(n-1) \nabla^s f(\hat{\mathbf{w}}(n-1))} \\ &= - \frac{\text{sgn}(\hat{\mathbf{w}}(n-1))^T \mathbf{k}(n) \zeta(n)}{(1-\lambda) \text{sgn}(\hat{\mathbf{w}}(n-1))^T \mathbf{P}(n-1) \text{sgn}(\hat{\mathbf{w}}(n-1))}. \end{aligned} \tag{21}$$

The regularization factor is calculated in line 6 in Algorithm 2. When calculating the complexity of the regularization factor, it should be taken into account that the regularization factor uses the elements calculated before line 6. Taking this into account and calculating the complexity, (20) requires  $2N + 3$  multiplications, whereas (21) requires  $N + 1$  multiplications.

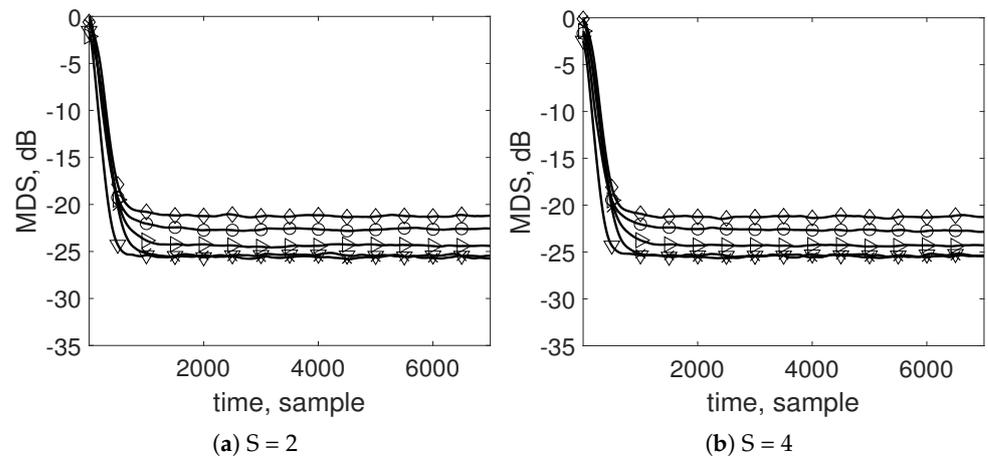
#### 4. Simulation Results

For the simulation in this section, we set the same experimental conditions as in [13] (code is available at <https://web.itu.edu.tr/eksioglu/publ.htm> accessed on 16 March 2021). We assume two system parameters,  $\mathbf{w}_o$ , with  $N = 64$  taps and 256 taps, respectively. Out of the  $N$  coefficients, the only  $S$  coefficients are not zero. We generate the values of coefficients from an  $N(0, 1/S)$  distribution and randomly place the non-zero coefficients.

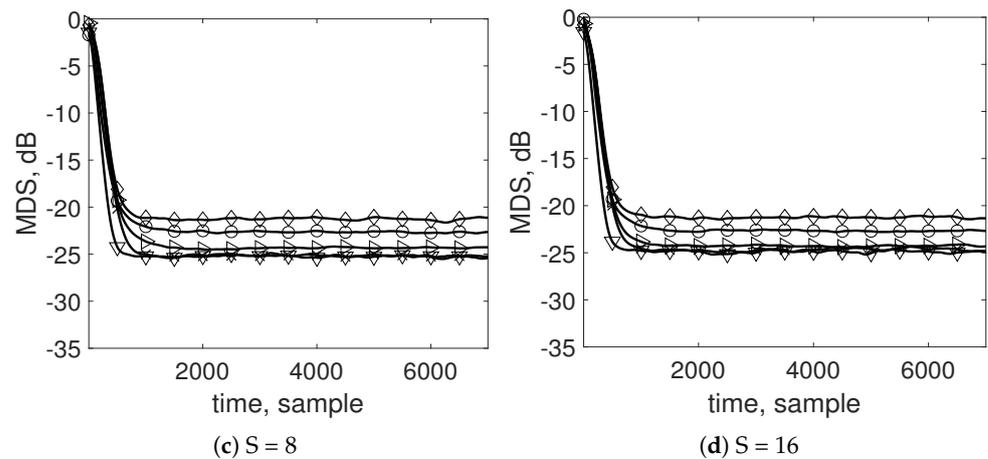
In the simulation, we show the system estimation results, using the proposed regularization factor that does not require the true system response information. We compare the  $l_1$ -RLS using the true system response [13], the  $l_1$ -RLS in [19],  $l_1$ -IWF in [21] and the  $l_1$ -RLS, using the proposed regularization factor selection method. In addition, we compare the estimation result from the conventional RLS. We simulate these algorithms in the sparse impulse response for  $S = 2, 4, 8,$  and  $16$  for the performance evaluation. Figure 1 shows the Mean Square Deviation (MSD) comparison results in the case of an order  $N = 64$  at SNR = 20 dB, where  $\text{MSD}(\hat{\mathbf{w}}(n)) = E\{\|\mathbf{w}_o - \hat{\mathbf{w}}(n)\|^2\}$ . Figure 2 also shows the MSD comparison results in the case of an order  $N = 256$  at SNR = 20 dB. In addition, the MSD comparison results are summarized in Table 1 at SNR = 20 dB, 10 dB and 0 dB, respectively.



**Figure 1.** MSD comparison in  $N = 64$  for  $S = 2, 4, 8,$  and  $16$  when applying the proposed regularization factor to the  $l_1$ -RLS: (a) MSD at  $S = 2$  (b) MSD at  $S = 4$  (c) MSD at  $S = 8$  (d) MSD at  $S = 16$  ( $\rightarrow$ -:  $l_1$ -RLS using the proposed regularization factor without the true system impulse response,  $\times$ -:  $l_1$ -RLS using the conventional regularization factor with the true system impulse response,  $\circ$ -:  $l_1$ -RLS from [19],  $\diamond$ -: conventional RLS without considering sparsity,  $\nabla$ -:  $l_1$ -IWF from [21]).



**Figure 2.** Cont.



**Figure 2.** MSD comparison in  $N = 256$  for  $S = 2, 4, 8,$  and  $16$  when applying the proposed regularization factor to the  $l_1$ -RLS: (a) MSD at  $S = 2$  (b) MSD at  $S = 4$  (c) MSD at  $S = 8$  (d) MSD at  $S = 16$  ( $\diamond$ -:  $l_1$ -RLS using the proposed regularization factor without the true system impulse response,  $\times$ -:  $l_1$ -RLS using the conventional regularization factor with the true system impulse response,  $\circ$ -:  $l_1$ -RLS from [19],  $\diamond$ -: conventional RLS without considering sparsity,  $\nabla$ -:  $l_1$ -IWF from [21]).

**Table 1.** MSD comparison in various SNR.

Channel Length	No. of Non-Zero Coefficients	SNR	20 dB	10 dB	0 dB
L = 64	S = 4	proposed method	-30.0	-20.2	-10.2
		$l_1$ -RLS with true impulse response	-30.8	-20.6	-11.0
		$l_1$ -RLS from [19]	-28.7	-18.7	-8.7
		$l_1$ -IWF from [21]	-29.5	-19.7	-9.0
		conventional RLS	-27.7	-17.5	-7.8
	S = 16	proposed method	-28.4	-18.6	-9.1
		$l_1$ -RLS with true impulse response	-28.5	-18.5	-9.3
		$l_1$ -RLS from [19]	-28.5	-18.4	-8.6
		$l_1$ -IWF from [21]	-29.0	-18.8	-9.2
		conventional RLS	-27.6	-17.7	-7.8
L = 256	S = 4	proposed method	-24.4	-14.3	-4.2
		$l_1$ -RLS with true impulse response	-25.3	-15.5	-5.5
		$l_1$ -RLS from [19]	-22.7	-12.7	-2.6
		$l_1$ -IWF from [21]	-25.2	-15.2	-3.6
		conventional RLS	-21.2	-11.2	-1.4
	S = 16	proposed method	-24.2	-14.3	-4.4
		$l_1$ -RLS with true impulse response	-24.6	-14.6	-5.0
		$l_1$ -RLS from [19]	-22.7	-12.5	-2.6
		$l_1$ -IWF from [21]	-24.8	-14.8	-3.4
		conventional RLS	-21.1	-11.1	-1.4

### 5. Discussion

The MSD comparison results in Figure 1 show that the estimation performance of  $l_1$ -RLS, using the regularization factor of the proposed method, is almost the same as the  $l_1$ -RLS, using the regularization factor with the true system impulse response.  $l_1$ -IWF

also performs almost the same as the  $l_1$ -RLS with the true system information as the results in [21]. The results from the proposed algorithm are even better than those of the  $l_1$ -RLS in [19]. Predictably, the conventional RLS has the worst MSD in all cases. Figure 1 confirms that, despite having no prior knowledge of the true system response, the proposed regularization factor is comparable to the conventional regularization factor with the true system impulse response.

Figure 2 shows the MSD results for a longer-order system:  $N = 256$ . We can see that Figure 2 has very similar results to those in Figure 1. Therefore, it can be confirmed that the proposed regularization factor calculation method works well, regardless of the system dimension.

Table 1 shows the performance of sparse RLS, using the proposed regularization factor calculation method, compared with other algorithms in various SNR situations. The performance of  $l_1$ -IWF is also similar to that of the proposed algorithm, but the performance of the proposed algorithm is better at low SNR. The results of Table 1 shows that, although SRLS using the proposed regularization factor does not actually utilize the impulse response of the true system, the MSD difference between SRLS with the regularization factor using information on the actual impulse response of the target system, and SRLS with the proposed regularization factor, is less than 2 dB. Therefore, it can be said that the performance of the two SRLS is very similar. In addition, as mentioned in Section 3, it is also remarkable that the computational complexity of the regularization factor can be reduced by about half of the conventional regularization factor.

## 6. Conclusions

In this paper, we proposed a new calculation method for a regularization factor in  $l_1$ -RLS, requiring no prior knowledge of the true system response. We also showed that the proposed regularization factor converges to the scaled optimal regularization factor. Therefore, we have shown that the proposed regularization factor can be used on behalf of the optimal regularization factor. The simulation results confirmed that the proposed regularization factor behaves almost the same as the conventional regularization factor with the true system response.

**Author Contributions:** Conceptualization, J.L.; methodology, J.L.; validation, J.L., K.L. and S.L.; formal analysis, J.L.; investigation, J.L., K.L. and S.L.; writing—original draft preparation, J.L.; writing—review and editing, K.L. and S.L.; visualization, J.L.; project administration, J.L. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Acknowledgments:** This research was supported by the Agency for Defense Development (ADD) in Korea (UD190005DD).

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Schreiber, W.F. Advanced television systems for terrestrial broadcasting: Some problems and some proposed solutions. *IEEE Proc.* **1995**, *83*, 958–981. [[CrossRef](#)]
2. Ariyavisitakul, S.; Sollenberger, N.R.; Greenstein, L.J. Tap selectable decision-feedback equalization. *IEEE Trans. Commun.* **1997**, *45*, 1497–1500. [[CrossRef](#)]
3. Li, W.; Preisig, J.C. Estimation of rapidly time-varying sparse channels. *IEEE J. Ocean. Eng.* **2007**, *32*, 927–939. [[CrossRef](#)]
4. Bruckstein, A.M.; Donoho, D.L.; Elad, M. From sparse solutions of systems of equations to sparse modeling of signals and images. *SIAM Rev.* **2009**, *51*, 34–81. [[CrossRef](#)]
5. Chen, Y.; Gu, Y.; Hero, A. Sparse LMS for system identification. In Proceedings of the 2009 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2009), Taipei, Taiwan, 19–24 April 2009; IEEE: Piscataway, NJ, USA, 2009; pp. 3125–3128.

6. Taheri, O.; Vorobyov, S. Sparse channel estimation with  $l_p$ -norm and reweighted  $l_1$ -norm penalized least mean squares. In Proceedings of the 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2011), Prague, Czech Republic, 22–27 May 2011; IEEE: Piscataway, NJ, USA, 2011; pp. 2864–2867.
7. Gui, G.; Xu, L.; Adachi, F. RZA-NLMF algorithm-based adaptive sparse sensing for realizing compressive sensing. *EURASIP J. Adv. Signal Process.* **2014**, *2014*, 125–134. [[CrossRef](#)]
8. Amresh, K.; Ikhlaq, H.; Bhim, S. Double-stage three-phase grid-integrated solar PV system with fast zero attracting normalized least mean fourth based adaptive control. *IEEE Trans. Ind. Electron.* **2018**, *65*, 3921–3931.
9. Gui, G.; Abolfazl, M.; Adachi, F. Sparse LMS/F algorithms with application to adaptive system identification. *Wirel. Commun. Mob. Comput.* **2015**, *15*, 1649–1658. [[CrossRef](#)]
10. Gui, G.; Xu, L.; Adachi, F. Extra gain: Improved sparse channel estimation using reweighted  $l_1$ -norm penalized LMS/F algorithm. In Proceedings of the 2014 IEEE/CIC International Conference on Communications in China (ICCC 2014), Shanghai, China, 13–15 October 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 13–15.
11. Babadi, B.; Kalouptsidis, N.; Tarokh, V. SPARLS: The sparse RLS algorithm. *IEEE Trans. Signal Process.* **2010**, *58*, 4013–4025. [[CrossRef](#)]
12. Angelosante, D.; Bazerque, J.; Giannakis, G. Online adaptive estimation of sparse signals: Where RLS meets the  $l_1$ -norm. *IEEE Trans. Signal Process.* **2010**, *58*, 3436–3477. [[CrossRef](#)]
13. Eksioğlu, E. RLS algorithm with convex regularization. *IEEE Signal Process. Lett.* **2011**, *18*, 470–473. [[CrossRef](#)]
14. Zakharov, Y.; Nascimento, V. DCD-RLS adaptive filters with penalties for sparse identification. *IEEE Trans. Signal Process.* **2013**, *61*, 3198–3213. [[CrossRef](#)]
15. Lim, J.; Pang, H. Mixed norm regularized recursive total least squares for group sparse system identification. *Int. J. Adapt. Control Signal Process.* **2016**, *30*, 664–673. [[CrossRef](#)]
16. Lim, J.; Pang, H.  $l_1$  regularized recursive total least squares based sparse system identification for the error-in-variables. *SpringerPlus* **2016**, *5*, 1460. [[CrossRef](#)] [[PubMed](#)]
17. Chen, Y.; Gui, G. Recursive least square-based fast sparse multipath channel estimation. *Int. J. Commun. Syst.* **2017**, *30*, e3278. [[CrossRef](#)]
18. Yang, J.; Xu, Y.; Rong, H.; Du, S.; Chen, B. Sparse recursive least mean  $p$ -power extreme learning machine for regression. *IEEE Access* **2018**, *6*, 16022–16034. [[CrossRef](#)]
19. Yang, C.; Qiao, J.; Ahmad, Z.; Nie, K.; Wang, L. Online sequential echo state network with sparse RLS algorithm for time series prediction. *Neural Netw.* **2019**, *118*, 32–42. [[CrossRef](#)] [[PubMed](#)]
20. Lim, J.; Lee, S. Regularization factor selection method for  $l_1$ -Regularized RLS and its modification against uncertainty in the regularization factor. *Appl. Sci.* **2019**, *9*, 202. [[CrossRef](#)]
21. Lim, J.  $l_1$ -norm iterative Wiener filter for sparse channel estimation. *Circuits Syst. Signal Process.* **2020**, *39*, 6386–6393. [[CrossRef](#)]
22. Stankovic, L. *Digital Signal Processing with Selected Topics Adaptive Systems, Time-Frequency Analysis, Sparse Signal Processing*; CreateSpace Independent Publishing Platform: Charleston, SC, USA, 2015; p. 766.
23. Khalid, S.; Abrar, S. Blind adaptive algorithm for sparse channel equalization using projections onto  $l_p$ -ball. *Electron. Lett.* **2015**, *51*, 1422–1424. [[CrossRef](#)]
24. Oppenheim, A.; Schaffer, R. *Digital Signal Processing*; Prentice-Hall: Hoboken, NJ, USA, 1975; p. 204.
25. Farhang-Boroujeny, B. *Adaptive Filter Theory and Applications*; John Wiley & Sons: Hoboken, NY, USA, 1999; pp. 419–427.
26. Diniz, P. *Adaptive Filtering Algorithms and Practical Implementation*, 3rd ed.; Springer: New York, NY, USA, 2008; p. 208.