applied sciences

MDPI

Article

A Recursive Least-Squares with a Time-Varying Regularization Parameter

Maaz Mahadi [1,2,*], Tarig Ballal [3], Muhammad Moinuddin [1,2] and Ubaid M. Al-Saggaf [1,2]

[1] Center of Excellence in Intelligent Engineering Systems (CEIES), King Abdulaziz University, Jeddah 21589, Saudi Arabia; mmsansari@kau.edu.sa (M.M.); usaggaf@kau.edu.sa (U.M.A.-S.)
[2] Department of Electrical and Computer Engineering, King Abdulaziz University, Jeddah 21589, Saudi Arabia
[3] Division of Computer, Electrical and Mathematical Sciences and Engineering (CEMSE), King Abdullah University of Science and Technology (KAUST), Jeddah 23955, Saudi Arabia; tarig.ahmed@kaust.edu.sa
* Correspondence: mmahadi@stu.kau.edu.sa

Abstract: Recursive least-squares (RLS) algorithms are widely used in many applications, such as real-time signal processing, control and communications. In some applications, regularization of the least-squares provides robustness and enhances performance. Interestingly, updating the regularization parameter as processing data continuously in time is a desirable strategy to improve performance in applications such as beamforming. While many of the presented works in the literature assume non-time-varying regularized RLS (RRLS) techniques, this paper deals with a time-varying RRLS as the parameter varies during the update. The paper proposes a novel and efficient technique that uses an approximate recursive formula, assuming a slight variation in the regularization parameter to provide a low-complexity update method. Simulation results illustrate the feasibility of the derived formula and the superiority of the time-varying RRLS strategy over the fixed one.

Keywords: recursive least-squares (RLS); tikhonov regularization; Taylor's series

check for updates

Citation: Mahadi, M.; Ballal, T.; Moinuddin, M.; Al-Saggaf, U.M. A Recursive Least-Squares with a Time-Varying Regularization Parameter. *Appl. Sci.* **2022**, *12*, 2077. https://doi.org/10.3390/app12042077

Academic Editor: Javier Hernando

Received: 18 November 2021
Accepted: 11 February 2022
Published: 16 February 2022

1. Introduction

Least-squares (LS) estimation methods emerge crucially in many applied sciences and engineering applications such as geodesy, adaptive filters, statistics, signal processing and control theory, to name a few. It is widespread due to its simple and efficient implementation. The standard LS problem dates back to 1795 AD when Gauss used it to study planetary motions [1]. An obvious advantage of this estimation method is that it does not require prior information on data; only a signal model is assumed [2]. In many applications, data are increased progressively in time. In this case, data can be processed for least-squares problems in two manners, either waiting for all the available data to be collected or processing them sequentially in time. The latter is known as recursive least-squares (RLS), or sequential least squares [2]. The RLS algorithm is one of the most significant advancements of the LS estimation in the twentieth century, which Gauss also conceived in his work on celestial bodies. The original work on the RLS algorithm in modern times is often credited to Plackett [3].

When working on LS problems, there are some cases where regularization is needed to compromise solution biasing and algorithmic robustness [4]. Different regularized LS and regularized RLS algorithms have emerged. One of these algorithms is the standard regularized exponentially weighted RLS that employs a regularization matrix whose elements fade exponentially to zero. However, some applications require maintaining regularization throughout the adaptive estimation process, such as beamforming. In addition, regularization ensures the existence of the sample covariance matrix inverse when the matrix has fewer rows than columns or is rank deficient. Regularization has a rich history in matrix computation and numerical analysis methods [5]. Tikhonov regularization methods are among the famous and earliest references on the regularization topic [6].

The literature concerning regularized RLS is quite limited because of the complexity associated with implementation [7]. In many research works, the regularization term does not fade but is treated fixed in all recursion steps, e.g., [8,9]. However, in some cases, such as adaptive beamforming, this regularization term needs to be adjusted, i.e., time-varying, to cope with changes in the data or noise [10,11]. The work in [12] proposes a special regularization structure that is updated in time using rank-1 matrices. The structure allows inflation or deflation of the regularization matrix and can ensure, on average, the invertibility of the sample covariance matrix with reasonable complexity. However, the sample covariance matrix can be positive-semidefinite, i.e., its smallest eigenvalue can become zero. The work presented in [4] proposes a time-varying regularized RLS with a generic structure. However, its implementation is complex.

Different variations of the regularized least squares for many applications are presented in the literature. For acoustic echo cancellation, the problems of acoustic impulse response estimation and nonlinearity modeling are addressed in [13] with a regularized RLS solution to handle these problems. A variable regularization technique is presented in [14] for the fast affine projection algorithm. The work in [15] introduced an algorithm that accounts for a cost function with a time-varying weighted regularization RLS using a sliding window approach.

Another parameter that is related to the regularization parameter is the forgetting factor parameter. In some cases, it is preferable to keep the newer data and ignore the older ones by using a forgetting factor parameter which applies higher weighting to recent data [16]. The problem of updating the forgetting factor parameter recursively in time can be found, for example, in [17–19].

In this paper, we consider the problem of solving regularized least-squares recursively. The recursion is viewed as a time-update process, which means that the number of observations increases as time progresses. The recursive solution helps in reducing the complexity associated with computations and memory storage requirements.

2. The Least-Squares Estimation Problem

This paper deals with estimating a vector $\mathbf{x} \in \mathbb{R}^{n\times 1}$ from a linear model of the form

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{z}, \tag{1}$$

where $\mathbf{y} \in \mathbb{R}^{m\times 1}$ is the observation vector, $\mathbf{A} \in \mathbb{R}^{m\times n}$ is a full-rank data matrix, (i.e., rank $(\mathbf{A}) = m$ if $m \leq n$ or rank $(\mathbf{A}) = n$ if $n \leq m$), and $\mathbf{z} \in \mathbb{R}^{m\times 1}$ is a Gaussian noise vector, (i.e., each element of $\mathbf{z}$ follows a Gaussian distribution). The LS estimation solves the following unconstrained minimization problem [20]:

$$\min_{\hat{\mathbf{x}}} \|\mathbf{y} - \mathbf{A}\hat{\mathbf{x}}\|_2^2, \tag{2}$$

where $\|.\|$ is the Euclidean norm. That is, we seek an estimate of $\mathbf{x}$, (i.e., $\hat{\mathbf{x}}$) that minimizes the l_2-norm of the residual error. Two cases can be identified based on dimensions m and n. The first case is the over-determined least-squares, where the number of rows of data matrix $\mathbf{A}$ is at least equal to the number of its columns, i.e., $m \geq n$; therefore, Equation (2) either has a unique solution or an infinite number of solutions [20]. In this case, the solution of Equation (2) is given by

$$\hat{\mathbf{x}}_{\mathrm{LS}} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{y}. \tag{3}$$

The second case is the under-determined least-squares, where the number of columns of $\mathbf{A}$ exceeds the number of rows, i.e., $m < n$. In this case, there is an infinite number of solutions [20].

A variation of the standard least-squares optimization problem involves a regularization parameter which servers two purposes. First, it allows the incorporation of some a priori information about the solution into the problem formulation. Second, it alleviates problems associated with rank-deficiency or ill-conditioning of the data matrix $\mathbf{A}$ [20].

Different regularization methods have been presented in the literature which include Tikhonov regularization, the shrunken estimator, the covariance shaping LS estimator [21] and ridge regression [22]. In Tikhonov regularization methods [6], a penalty term is added to Equation (2) to shrink the elements of $\hat{\mathbf{x}}$ towards zero. A variety of penalty terms have been proposed; a popular form is the l_2 penalization defined as follows [23]:

$$\min_{\hat{\mathbf{x}}} \|\mathbf{y} - \mathbf{A}\hat{\mathbf{x}}\|_2^2 + \gamma\|\hat{\mathbf{x}}\|_2^2, \tag{4}$$

where $\gamma \in \mathbb{R}^+$ is a regularization parameter. The solution of (4) is given by

$$\hat{\mathbf{x}}_{\text{RLS}} = (\mathbf{A}^T\mathbf{A} + \gamma\mathbf{I}_n)^{-1}\mathbf{A}^T\mathbf{y}, \tag{5}$$

where $\mathbf{I}_n$ is the identity matrix of dimension n. It is shown that there always exists $\gamma > 0$ for which (5) produces lower mean squared-error (MSE) than the LS estimator in (3) [23].

3. Regularization Parameter Selection

The optimal regularization parameter can be set to minimize the MSE error of the estimate as follows [23–25]:

$$\gamma_{\text{opt}} = \arg\min_{\gamma} \text{MSE}, \tag{6}$$

where $\text{MSE} := \mathbb{E}[\|\mathbf{x} - \hat{\mathbf{x}}_{\text{RLS}}\|_2^2]$. Practically, γ_{opt} does not have a closed-form solution because the actual vector $\mathbf{x}$ is unknown. Some methods are proposed in the literature that assume a specific structure of the noise vector $\mathbf{z}$ and/or data vector $\mathbf{x}$. For example, if the noise vector $\mathbf{z}$ is zero-mean with i.i.d. elements, and $\mathbf{x}$ is of zero-mean and independent components, the optimal regularization parameter can be approximated as follows [23]:

$$\gamma_{\text{opt}} \approx \frac{n\sigma_{\mathbf{z}}^2}{\text{tr}[\mathbf{R}_{\mathbf{xx}}]}, \tag{7}$$

where $\sigma_{\mathbf{z}}^2$ is the variance of $\mathbf{z}$ and $\mathbf{R}_{\mathbf{xx}} \triangleq \mathbb{E}[\mathbf{x}\mathbf{x}^T]$ is the covariance matrix of $\mathbf{x}$. From Equation (7), we observe that optimal tuning of the regularization parameter should take into account any variation in the noise of signal second-order statistics. In RLS, computing a new regularization parameter at each iteration will increase the computational complexity.

In the following section, we present a method for recursively updating regularization parameter in a computationally efficient manner.

4. Recursive Least-Squares

We assume that the current dimension of $\mathbf{y}$ and the number of rows of $\mathbf{A}$ is m. We denote this by $\mathbf{y}_m$ and $\mathbf{A}_m$, respectively. Moreover, we can denote the individual entries of the observation vector $\mathbf{y}$ by $\{d_i\}$, and the individual rows of $\mathbf{A}$ by $\{\mathbf{u}_i^T\}$. As time progresses, the dimension of $\mathbf{y}_m$ and the rows of $\mathbf{A}_m$ increase by 1 and become $\mathbf{y}_{m+1}$ and $\mathbf{A}_{m+1}$, which can be written as follows:

$$\mathbf{y}_{m+1} = \begin{bmatrix} \mathbf{y}_m \\ d_{m+1} \end{bmatrix}, \tag{8}$$

$$\mathbf{A}_{m+1} = \begin{bmatrix} \mathbf{A}_m \\ \mathbf{u}_{m+1}^T \end{bmatrix}, \tag{9}$$

where d_{m+1} is the new observed data and $\mathbf{u}_{m+1}^T$ is the new row of $\mathbf{A}$. Note that the subscript m can be viewed as a time index as well. The estimate $\hat{\mathbf{x}}$ at dimension m can be written as follows:

$$\hat{\mathbf{x}}_m = (\mathbf{A}_m^T\mathbf{A}_m + \gamma\mathbf{I}_n)^{-1}\mathbf{A}_m^T\mathbf{y}_m, \tag{10}$$

and for the time-updated estimate, $\hat{\mathbf{x}}_{m+1}$,

$$\hat{\mathbf{x}}_{m+1} = (\mathbf{A}_{m+1}^T \mathbf{A}_{m+1} + \gamma \mathbf{I}_n)^{-1} \mathbf{A}_{m+1}^T \mathbf{y}_{m+1}, \tag{11}$$

Going from Equations (10) to (11) is known as a time-update step because it amounts to employing new data $\{d_{m+1}, \mathbf{u}_{m+1}^T\}$ in addition to all previous data [20]. Performing time-updates based on Equations (10) and (11) is costly in terms of computation since it requires inverting two $n \times n$ matrices, in addition to memory storage, required for storing the entries of $\mathbf{y}_m$ and $\mathbf{A}_m$ [20]. Hence, it is important to seek a low-cost update method. The following section introduces two regularized recursive least-squares (RRLS) algorithms. The first one is well-established in the literature, which assumes a fixed regularization parameter through the update process [2,20]. The second (proposed) algorithm handles the case when the regularization parameter varies during the update process.

4.1. Recursive Least-Squares with Fixed Regularization

To obtain a recursive formula of the estimate before the update ($\hat{\mathbf{x}}_m$) and after the update, ($\hat{\mathbf{x}}_{m+1}$), we can write

$$\hat{\mathbf{x}}_m = \mathbf{P}_m \mathbf{A}_m^T \mathbf{y}_m, \tag{12}$$

$$\hat{\mathbf{x}}_{m+1} = \mathbf{P}_{m+1} \mathbf{A}_{m+1}^T \mathbf{y}_{m+1}, \tag{13}$$

where

$$\mathbf{P}_m \triangleq (\mathbf{A}_m^T \mathbf{A}_m + \gamma \mathbf{I}_n)^{-1}, \tag{14}$$

$$\mathbf{P}_{m+1} \triangleq (\mathbf{A}_{m+1}^T \mathbf{A}_{m+1} + \gamma \mathbf{I}_n)^{-1}. \tag{15}$$

The initial condition is $\mathbf{P}_0 = \frac{1}{\gamma}\mathbf{I}_n$. Note that $\mathbf{P}_m$ can be viewed as the inverse of scaled regularized sample covariance matrix. We can write $\mathbf{P}_{m+1}$ as follows:

$$\mathbf{P}_{m+1} = (\mathbf{P}_m^{-1} + \mathbf{u}_{m+1}\mathbf{u}_{m+1}^T)^{-1}. \tag{16}$$

Using the matrix inversion lemma (Equation (30.11) in [20]), we obtain

$$\mathbf{P}_{m+1} = \mathbf{P}_m - \frac{\mathbf{P}_m \mathbf{u}_{m+1} \mathbf{u}_{m+1}^T \mathbf{P}_m}{1 + \mathbf{u}_{m+1}^T \mathbf{P}_m \mathbf{u}_{m+1}}. \tag{17}$$

This recursion updates $\mathbf{P}_m$ to obtain $\mathbf{P}_{m+1}$. By substituting Equation (17) in Equation (13) and rearranging, the estimate $\hat{\mathbf{x}}_{m+1}$ can be obtained by updating $\mathbf{x}_m$ as follows [20]:

$$\hat{\mathbf{x}}_{m+1} = \hat{\mathbf{x}}_m + \frac{\mathbf{P}_m \mathbf{u}_{m+1}}{1 + \mathbf{u}_{m+1}^T \mathbf{P}_m \mathbf{u}_{m+1}} [d_{m+1} - \mathbf{u}_{m+1}^T \hat{\mathbf{x}}_m], \tag{18}$$

with $\hat{\mathbf{x}}_0 = \mathbf{0}$. The RRLS method with fixed regularization is summarized in Algorithm 1.

Algorithm 1: Fixed RRLS

```
1. Start with $\hat{\mathbf{x}}_0 = \mathbf{0}$ and $\mathbf{P}_0 = \frac{1}{\gamma_0}\mathbf{I}$.
2. For each new observed data, $d_{m+1}$, and new row, $\mathbf{u}_{m+1}^T$ do the following:
   • Calculate: $\mathbf{P}_{m+1}$ using Equation (17).
   • Calculate: $\hat{\mathbf{x}}_{m+1}$ using Equation (18).
```

4.2. The Proposed Regularization Recursive Least-Squares

This section aims to derive a recursive formula of the RRLS estimator that takes into consideration the requirement to also update the regularization parameter. If the regularization parameter before the update is γ_m and after the update is γ_{m+1}, then Equations (14) and (15) become

$$\mathbf{P}_m \triangleq (\mathbf{A}_m^T \mathbf{A}_m + \gamma_m \mathbf{I}_n)^{-1}, \tag{19}$$

and

$$\mathbf{P}_{m+1} \triangleq (\mathbf{A}_{m+1}^T \mathbf{A}_{m+1} + \gamma_{m+1} \mathbf{I}_n)^{-1}, \tag{20}$$

with an initial condition $\mathbf{P}_0 = \dfrac{1}{\gamma_0}\mathbf{I}_n$. Now, $\mathbf{P}_{m+1}$ becomes

$$\mathbf{P}_{m+1} = (\mathbf{P}_m^{-1} + \mathbf{u}_{m+1}\mathbf{u}_{m+1}^T + (\gamma_{m+1} - \gamma_m)\mathbf{I}_n)^{-1}. \tag{21}$$

Manipulating Equation (21) similar to Equation (17) using the matrix inversion lemma for Equation (21) is not helpful because the presence of $(\gamma_{m+1} - \gamma_m)\mathbf{I}_n$. To obtain a desirable update formula, we rely on the assumption that $(\gamma_{m+1} - \gamma_m)$ is small compared to $\mathbf{P}_m^{-1} + \mathbf{u}_{m+1}\mathbf{u}_{m+1}^T$. Hence, the following approximation can be utilized [26]:

$$(\mathbf{Q} + \delta\mathbf{M})^{-1} \approx \mathbf{Q}^{-1} - \delta\mathbf{Q}^{-1}\mathbf{M}\mathbf{Q}^{-1}, \tag{22}$$

where

$$\mathbf{Q} \triangleq \mathbf{P}_m^{-1} + \mathbf{u}_{m+1}\mathbf{u}_{m+1}^T, \tag{23}$$

$$\mathbf{M} \triangleq \mathbf{I}_n, \tag{24}$$

$$\delta \triangleq \gamma_{m+1} - \gamma_m. \tag{25}$$

With this assumption, the following recursions can be easily derived as:

$$\mathbf{P}_{m+1} \approx \left(\mathbf{P}_m - \frac{\mathbf{P}_m\mathbf{u}_{m+1}\mathbf{u}_{m+1}^T\mathbf{P}_m}{1 + \mathbf{u}_{m+1}^T\mathbf{P}_m\mathbf{u}_{m+1}}\right) - \delta\left(\mathbf{P}_m - \frac{\mathbf{P}_m\mathbf{u}_{m+1}\mathbf{u}_{m+1}^T\mathbf{P}_m}{1 + \mathbf{u}_{m+1}^T\mathbf{P}_m\mathbf{u}_{m+1}}\right)^2, \tag{26}$$

$$\hat{\mathbf{x}}_{m+1} \approx \left[\mathbf{I}_n - \delta\left(\mathbf{P}_m - \frac{\mathbf{P}_m\mathbf{u}_{m+1}\mathbf{u}_{m+1}^T\mathbf{P}_m}{1 + \mathbf{u}_{m+1}^T\mathbf{P}_m\mathbf{u}_{m+1}}\right)\right] \cdot \left[\hat{\mathbf{x}}_m + \frac{\mathbf{P}_m\mathbf{u}_{m+1}}{1 + \mathbf{u}_{m+1}^T\mathbf{P}_m\mathbf{u}_{m+1}}[d_{m+1} - \mathbf{u}_{m+1}^T\hat{\mathbf{x}}_m]\right]. \tag{27}$$

Remark 1. *The approximation Equation* (22) *is based on the first-order Taylor's series expansion of matrices. Consequently, more accurate results can be obtained by expanding to higher orders.*

The proposed method for RRLS is summarized in Algorithm 2.

Algorithm 2: Time-varying RRLS (Proposed)

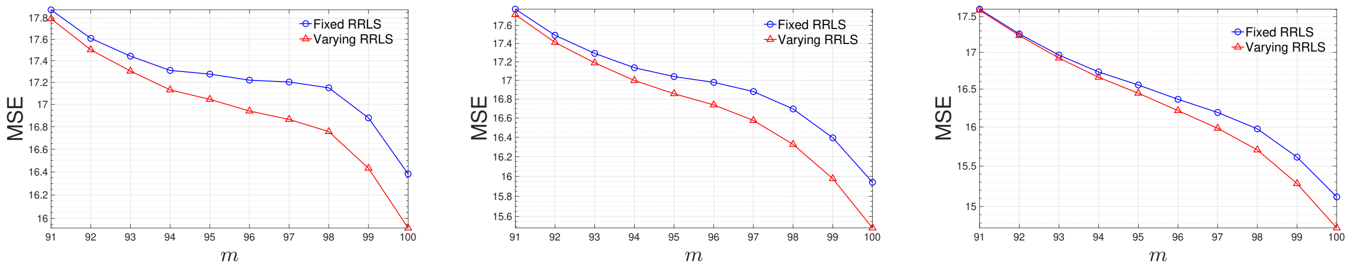
```
1. Start with $\hat{\mathbf{x}}_0 = \mathbf{0}$ and $\mathbf{P}_0 = \frac{1}{\gamma_0}\mathbf{I}$.
2. For each new observed data, $d_{m+1}$, and new row, $\mathbf{u}_{m+1}^T$ do the following:
   • Estimate the new regularization parameter, $\gamma_{m+1}$ using a suitable regularization parameter selection method (e.g., Equation (7)).
   • Calculate: $\delta = \gamma_{m+1} - \gamma_m$.
   • Calculate: $\mathbf{P}_{m+1}$ using Equation (26).
   • Calculate: $\hat{\mathbf{x}}_{m+1}$ using Equation (27).
```

5. Simulation Results

The results in this section simulate model (1), where $\mathbf{A}$ is generated independently from a Gaussian distribution with zero-mean and unit variance. The noise, $\mathbf{z}$, and the data vector, $\mathbf{x}$, are zero-mean Gaussian vectors. Having these assumptions on hand suggests the optimal regularization parameter can be obtained from Equation (7). Initially, the matrix $\mathbf{A}$ has 100 columns $(n = 100)$ and 90 rows $(m = 90)$. We simulate three scenarios with uncertainties in calculating the signal-to-noise ratio (SNR). The SNR varies uniformly during the update process within the three intervals $[10.5, 11]$ dB, $[10.25, 11]$ dB and $[10, 11]$ dB, i.e., the uncertainties are 0.5 dB, 0.75 dB, and 1 dB, respectively. We consider two competitive methods: the fixed RRLS and the time-varying RRLS. In the fixed regularization setting, the regularization parameter does not account for this variation in the SNR, while, in the varying regularization parameter, the parameter is changed according to Equation (7).

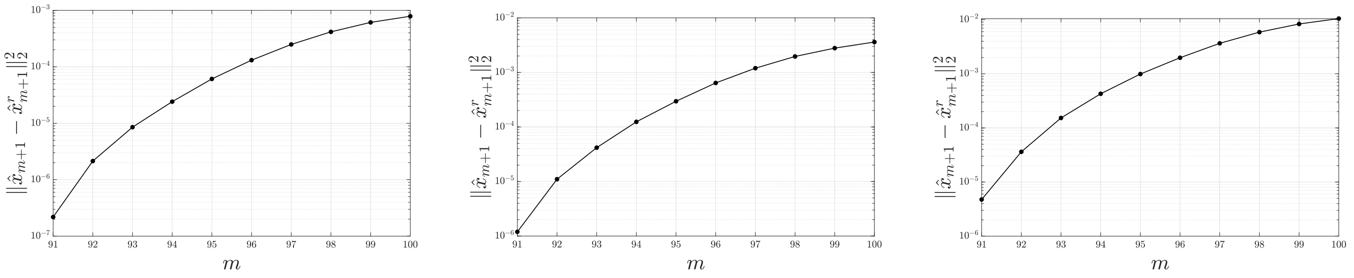
Figure 1 compares the MSE performance for the fixed regularization and the varying regularization parameter methods when the number of observations is increased one at a time. Figure 1a shows the result obtained when the uncertainty is 0.5 dB. When applying the fixed regularized recursive formula at $(m = 91)$, the MSE scores 17.9 square units, while the time-varying RRLS scores 17.8 square units. At $m = 100$, the MSE is 16.4 square units for the fixed RLS and is 15.9 square units for the varying RLS. Figure 1b plots the MSE against m when the uncertainty is increased to 0.75 dB. The MSE of the fixed regularization method is 17.8 square units when $m = 91$ and decreases to 15.9 at $m = 100$. On the other hand, the proposed time-varying RRLS achieves 17.7 square units at $m = 91$ and falls to 15.5 square units when $m = 100$. Finally, Figure 1c compares the performance of the two methods when the uncertainty in SNR is 1 dB. The MSE of both the fixed RRLS and the time-varying RRLS is 17.6 square units at $m = 91$. As m increases, the MSE reaches 15.1 square units for the fixed regularization method and 14.7 square units for the varying regularization method when $m = 100$. Hence, it can be concluded from Figure 1 that the performance of the proposed time-varying RRLS outperforms the fixed RRLS for $m > 90$ most of the time.

Figure 2 plots the error in estimating the data vector recursively using the proposed time-varying RRLS versus the number of rows, m. As expected, the recursive formula's error increases in each iteration because the derived formula relies on some approximations. Figure 2a plots the error curve when the uncertainty in the SNR is 0.5 dB. The minimum error is 9×10^{-6} at $m = 91$ and the maximum error is 2×10^{-3} at $m = 100$. Figure 2b illustrates that the recursive update's minimum and maximum errors are between 1×10^{-6} and 8×10^{-2}, respectively, for 0.75 dB uncertainty. Finally, for 1 dB uncertainty in calculating the SNR in Figure 2c, the minimum error is 7×10^{-5} at $m = 91$ and the maximum error is 1×10^{-2} at $m = 100$.



(**a**) (**b**) (**c**)

Figure 1. Comparison between the MSE performance of the fixed regularization parameter (Algorithm 1) and the varying regularization parameter (Algorithm 2) methods of the least squares solution considering three different uncertainties in estimating the SNR, 0.5 dB, 0.75 dB and 1 dB. (**a**) ΔSNR = 0.5 dB. (**b**) ΔSNR = 0.75 dB. (**c**) ΔSNR = 1 dB.



(**a**) (**b**) (**c**)

Figure 2. The estimated error due to using the proposed recursive formula considering three different uncertainties in estimating the SNR, 0.5 dB, 0.75 dB and 1 dB. (**a**) ΔSNR = 0.5 dB. (**b**) ΔSNR = 0.75 dB. (**c**) ΔSNR = 1 dB.

6. Conclusions

Recursive least-squares algorithms are widely used in many applications. This paper proposed a recursive algorithm for computing a time-varying non-fading regularized RLS. An approximate formula of the regularized RLS was derived, assuming slight deviations in the parameter during the time-update process to cope with noise and data statistics variations. Simulation results provided in this paper considered different uncertainties in calculating the SNR. The results demonstrated the benefit of the proposed approach in providing an efficient means of tracking the regularization parameter changes compared to the fixed RRLS method. Because of the approximation involved in deriving the proposed method, its efficiency is limited to the cases where the change in the regularization parameter is small. Hence, more accurate solutions with efficient implementation are needed for future work.

Author Contributions: Conceptualization, M.A., T.B. and M.M. (Maaz Mahadi); methodology, T.B.; software, M.A.; writing—original draft preparation, M.A.; writing—review and editing, T.B.; supervision, M.M. (Muhammad Moinuddin) and U.M.A.-S. 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.

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

References

1. Gauss, C.F. *Theory of the Motion of the Heavenly Bodies Moving about the Sun in Conic Sections: A Translation of Gauss's "Theoria Motus"*; Little, Brown and Company: Boston, MA, USA, 1857.
2. Kay, S.M. *Fundamentals of Statistical Signal Processing*; Prentice Hall: Hoboken, NJ, USA, 1993.
3. Plackett, R.L. Some theorems in least squares. *Biometrika* **1950**, *37*, 149–157. [CrossRef] [PubMed]
4. Tsakiris, M.C.; Lopes, C.G.; Nascimento, V.H. An array recursive least-squares algorithm with generic nonfading regularization matrix. *IEEE Signal Process. Lett.* **2010**, *17*, 1001–1004. [CrossRef]
5. Golub, G.H.; Van Loan, C.F. *Matrix Computations*; JHU Press: Baltimore, MD, USA, 2013; Volume 3.
6. Tikhonov, A.N. *On the Solution of Ill-Posed Problems and the Method of Regularization*; Russian Academy of Sciences: Moscow, Russia, 1963; Volume 151, pp. 501–504.
7. Tsakiris, M. On the Regularization of the Recursive Least Squares Algorithm. Ph.D. Thesis, Universidade de São Paulo, São Paulo, Brazil, 2010.
8. Horita, E.; Sumiya, K.; Urakami, H.; Mitsuishi, S. A leaky RLS algorithm: Its optimality and implementation. *IEEE Trans. Signal Process.* **2004**, *52*, 2924–2936. [CrossRef]
9. Horita, E. A Computationally Efficient Leaky and Regularized RLS Filter for Its Short Length. *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.* **2017**, *100*, 3045–3048. [CrossRef]

10. Ma, N.; Goh, J.T. Efficient method to determine diagonal loading value. In Proceedings of the 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003, (ICASSP'03), Hong Kong, China, 6–10 April 2003; Volume 5, pp. 5–341.
11. Mahadi, M.; Ballal, T.; Moinuddin, M.; Al-Naffouri, T.Y.; Al-Saggaf, U. Low-Complexity Robust Beamforming for a Moving Source. In Proceedings of the 2020 28th European Signal Processing Conference (EUSIPCO), Amsterdam, The Netherlands, 24–28 August 2020; pp. 1846–1850.
12. Gay, S.L. Dynamically regularized fast RLS with application to echo cancellation. In Proceedings of the 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings, Atlanta, GA, USA, 9 May 1996; Volume 2, pp. 957–960.
13. Comminiello, D.; Scarpiniti, M.; Azpicueta-Ruiz, L.A.; Arenas-García, J.; Uncini, A. Full proportionate functional link adaptive filters for nonlinear acoustic echo cancellation. In Proceedings of the 2017 25th European Signal Processing Conference (EUSIPCO), Kos, Greece, 28 August–2 September 2017; pp. 1145–1149.
14. Challa, D.; Grant, S.L.; Mohammad, A. Variable regularized fast affine projections. In Proceedings of the 2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07, Honolulu, HI, USA, 15–20 April 2007; Volume 1, pp. 1–89.
15. Ali, A.A.; Hoagg, J.B.; Mossberg, M.; Bernstein, D.S. On the stability and convergence of a sliding-window variable-regularization recursive-least-squares algorithm. *Int. J. Adapt. Control. Signal Process.* **2016**, *30*, 715–735. [CrossRef]
16. Goel, A.; Bruce, A.L.; Bernstein, D.S. Recursive Least Squares With Variable-Direction Forgetting: Compensating for the Loss of Persistency [Lecture Notes]. *IEEE Control. Syst. Mag.* **2020**, *40*, 80–102. [CrossRef]
17. Sun, X.; Ji, J.; Ren, B.; Xie, C.; Yan, D. Adaptive forgetting factor recursive least square algorithm for online identification of equivalent circuit model parameters of a lithium-ion battery. *Energies* **2019**, *12*, 2242. [CrossRef]
18. Bruce, A.L.; Goel, A.; Bernstein, D.S. Convergence and consistency of recursive least squares with variable-rate forgetting. *Automatica* **2020**, *119*, 109052. [CrossRef]
19. Tang, X.; Xu, Y.; Chen, R.; Zhou, Y. A Time-Varying Forgetting Factor-Based QRRLS Algorithm for Multichannel Speech Dereverberation. In Proceedings of the 2020 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), Louisville, KY, USA, 9–11 December 2020; pp. 1–6.
20. Sayed, A.H. *Fundamentals of Adaptive Filtering*; John Wiley & Sons: Hoboken, NJ, USA, 2003.
21. Eldar, Y.C.; Oppenheim, A.V. Covariance shaping least-squares estimation. *IEEE Trans. Signal Process.* **2003**, *51*, 686–697. [CrossRef]
22. Hoerl, A.E.; Kennard, R.W. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics* **1970**, *12*, 55–67. [CrossRef]
23. Ballal, T.; Suliman, M.A.; Al-Naffouri, T.Y. Bounded perturbation regularization for linear least squares estimation. *IEEE Access* **2017**, *5*, 27551–27562. [CrossRef]
24. Kress, R. *Numerical Analysis (Graduate Texts in Mathematics)*; Springer: Berlin/Heidelberg, Germany, 1998.
25. Suliman, M.A.; Ballal, T.; Al-Naffouri, T.Y. Perturbation-based regularization for signal estimation in linear discrete ill-posed problems. *Signal Process.* **2018**, *152*, 35–46. [CrossRef]
26. Petersen, K.B.; Pedersen, M.S. *The Matrix Cookbook*; Version 20121115; Technical University of Denmark: Lyngby, Denmark, 2012.