# A Leaky RLS Algorithm: Its Optimality and Implementation

# A Leaky RLS Algorithm: Its Optimality and Implementation

Eisuke Horita, Keitaro Sumiya, Hiroyuki Urakami, and Satoru Mitsuishi

***Abstract*—A leaky recursive least squares (LRLS) algorithm obtained by a criterion of the ridge regression with the exponential weighting factor was recently proposed by one of the authors. On the other hand, an optimization criterion for improving the method of total least squares (TLS) has been proposed by Chandrasekaran *et al.* In this work, it is expressed that there is a case where the equation obtained by the criterion of the LRLS algorithm is identical to one obtained by the extended criterion of Chandrasekaran *et al.* In addition, some implementations of the LRLS filter by using the method for updating the eigendecomposition of rank-one matrix updates, or by using the leaky least mean square (LLMS) algorithm, are introduced to decrease the computational complexity of the LRLS algorithm. Moreover, by means of computer experiments, it is shown that the LRLS and the LLMS algorithms yield more precise estimation parameters than the RLS algorithm when the method of Chandrasekaran *et al.* is more useful than that of LS and TLS. Besides, it is demonstrated that the LLMS algorithm can be effectively introduced into a noise reduction system for noisy speech signals to support the theoretical results in this work.**

***Index Terms*—Adaptive filters, computational complexity, parameter estimation.**

## I. Introduction

Recursive least squares (RLS) algorithms are widely used adaptive filters. Fast RLS algorithms based on QR decomposition using Givens rotations are known to be numerically robust and to own regular structures, which can lead to efficient implementations [1]–[3].

Alternative optimization criteria, however, have been proposed including, among others, regularized least-squares, ridge regression, and total least squares (TLS) [4]–[7]. In [8] and [9], an adaptive ridge regression algorithm with the exponential weighting factor, i.e., the leaky RLS (LRLS) algorithm, has been proposed. The LRLS algorithm is numerically stable for time-varying signals even in the worst case, although the estimated parameters by using the LRLS algorithm do not converge to the optimum values in the LS sense because of the ridge parameter [8], [9].

On the other hand, in contrast to the standard LS problem, the TLS formulation allows for errors in the data matrix. However, it still shows certain drawbacks that degrade its performance in practical situations. More explicitly, assume that $\boldsymbol{A} \in R^{k\times n}$ is a given full rank matrix with $k \geq n$, $\boldsymbol{b} \in R^k$ is a given vector, and consider the problem of solving the inconsistent linear system $\boldsymbol{A}\hat{\boldsymbol{x}}(k) \approx \boldsymbol{b}$ in the LS sense. The TLS solution assumes data uncertainties in $\boldsymbol{A}$ and proceeds to correct $\boldsymbol{A}$ and $\boldsymbol{b}$ by replacing them by their projections $\hat{\boldsymbol{A}}$ and $\hat{\boldsymbol{b}}$ onto a specific subspace and by solving the consistent linear system of equations $\hat{\boldsymbol{A}}\hat{\boldsymbol{x}}(k) = \hat{\boldsymbol{b}}$. The spectral norm of the correction $(\boldsymbol{A} - \hat{\boldsymbol{A}})$ in the TLS solution is bounded by the smallest singular value of $[\boldsymbol{A}\,\boldsymbol{b}]$. Although this norm might be small for vectors $\boldsymbol{b}$ that are close enough to the range space of $\boldsymbol{A}$, it need not always be so. In other words, the TLS solution may lead to situations in which the correction term is unnecessarily large. In [5]–[7], an optimization criterion for improving the method of TLS has been proposed by Chandrasekaran *et al.* The method of Chandrasekaran *et al.* is more useful than that of TLS when we consider a situation in which the uncertainties in $\boldsymbol{A}$ are very small, and $\boldsymbol{b}$ is far from the column space of $\boldsymbol{A}$ [7]. It has also been reported in [7] that there are cases where the equations obtained by the criterion of Chandrasekaran *et al.* are identical to ones obtained by the criterion of the ridge regression.

In this work, we express that there is a case where the equation obtained by the extended criterion of Chandrasekaran *et al.* with the exponential weighting factor is identical to one obtained by the criterion of the LRLS algorithm. In other words, we explain that it is possible for the LRLS algorithm to give more accurate estimation parameters than the RLS algorithm. In addition, some implementations of the LRLS filter by using the method for updating the eigendecomposition of rank-one matrix updates, or by using the leaky LMS (LLMS) algorithm, are introduced to decrease the computational complexity of the LRLS algorithm. Moreover, by means of computer experiments, we show that the LRLS and the LLMS algorithms yield more precise estimation parameters than the RLS algorithm when the method of Chandrasekaran *et al.* is more useful than that of LS and TLS. Besides, we demonstrate that the LLMS algorithm can be effectively introduced into the noise reduction system for noisy speech signals proposed in [10] and [11] to support the theoretical results in this work.

## II. Problem Formulation and Its Adaptive Solution

### A. Standard RLS and LRLS Algorithms

Let $\boldsymbol{A} \in R^{k\times n}$ be a given matrix with $k \geq n$ and $\boldsymbol{b} \in R^k$ a given vector, both of which are defined, respectively, by

$$\boldsymbol{u}^T(i) = [u(i) \quad u(i-1) \quad \ldots \quad u(i-n+1)] \tag{1}$$

$$\boldsymbol{A}^T = [\boldsymbol{u}(1) \quad \boldsymbol{u}(2) \quad \ldots \quad \boldsymbol{u}(k)] \tag{2}$$

$$\boldsymbol{b}^T = [d(1) \quad d(2) \quad \ldots \quad d(k)] \tag{3}$$

where the vectors $\boldsymbol{u}(i), i = 1, 2, \ldots, k$ consist of the inputs of an adaptive filter, and $d(i), i = 1, 2, \ldots, k$ denote the desired response.

We then define a criterion of the RLS algorithm with the initial condition discussed in [2] as

$$\min_{\hat{\boldsymbol{x}}(k)} \|\boldsymbol{W}(\boldsymbol{A}\hat{\boldsymbol{x}}(k) - \boldsymbol{b})\|_2^2 + \delta\lambda^k \|\hat{\boldsymbol{x}}(k)\|_2^2 \tag{4}$$

$$\boldsymbol{W} = \mathrm{diag}(\sqrt{\lambda^{k-1}}, \sqrt{\lambda^{k-2}}, \ldots, 1), (0 < \lambda \leq 1) \tag{5}$$

where $\lambda$ is the exponential weighting factor for processing time-varying signals, and $\delta$ is a small positive constant. Its solution vector $\hat{\boldsymbol{x}}(k)$ satisfies

$$(\boldsymbol{A}^T\boldsymbol{W}^2\boldsymbol{A} + \delta\lambda^k\boldsymbol{I})\hat{\boldsymbol{x}}(k) = \boldsymbol{A}^T\boldsymbol{W}^2\boldsymbol{b}. \tag{6}$$

The standard RLS algorithm is obtained by solving (6) adaptively and accurately as follows [1]:

[RLS algorithm]

Initialize the algorithm by setting

$$\boldsymbol{P}(0) = \delta^{-1}\boldsymbol{I}, \quad \hat{\boldsymbol{x}}(0) = \boldsymbol{O}. \tag{7}$$

Manuscript received January 6, 2002; revised October 4, 2003. The associate editor coordinating the review of this paper and approving it for publication was Dr. Joseph Tabrikian.

The authors are with the Faculty of Engineering, Kanazawa University, Kanazawa-shi, Japan (e-mail: e.horita@ieee.org).

Digital Object Identifier 10.1109/TSP.2004.834212

For each instant of time $k = 1, 2, \ldots$, compute

$$\nu(k) = d(k) - \boldsymbol{u}^T(k)\hat{\boldsymbol{x}}(k-1) \tag{8}$$
$$\boldsymbol{g}(k) = \boldsymbol{P}(k-1)\boldsymbol{u}(k)(\lambda + \boldsymbol{u}^T(k)\boldsymbol{P}(k-1)\boldsymbol{u}(k))^{-1} \tag{9}$$
$$\hat{\boldsymbol{x}}(k) = \hat{\boldsymbol{x}}(k-1) + \boldsymbol{g}(k)\nu(k) \tag{10}$$
$$\boldsymbol{P}(k) = \lambda^{-1}\{\boldsymbol{P}(k-1) - \boldsymbol{g}(k)\boldsymbol{u}^T(k)\boldsymbol{P}(k-1)\} \tag{11}$$

where

$$\boldsymbol{P}^{-1}(k) = \boldsymbol{A}^T\boldsymbol{W}^2\boldsymbol{A} + \delta\lambda^k\boldsymbol{I} = \sum_{i=1}^{k}\lambda^{k-i}\boldsymbol{u}(i)\boldsymbol{u}^T(i) + \delta\lambda^k\boldsymbol{I}. \tag{12}$$

In the above algorithm, its computational complexity is on the order of $n^2$ since the matrix inversion lemma can be applied to (12) [1]. In addition, fast and numerically stable RLS algorithms whose computational complexity is on the order of $n$ have been proposed [2], [3].

On the other hand, a criterion of the ridge regression with the exponential weighting factor discussed in [8] and [9] is defined as

$$\min_{\hat{\boldsymbol{x}}(k)}\|\boldsymbol{W}(\boldsymbol{A}\hat{\boldsymbol{x}}(k) - \boldsymbol{b})\|_2^2 + \alpha\|\hat{\boldsymbol{x}}(k)\|_2^2 \tag{13}$$

where $\alpha$ is a positive constant. Its solution vector $\hat{\boldsymbol{x}}(k)$ satisfies

$$(\boldsymbol{A}^T\boldsymbol{W}^2\boldsymbol{A} + \alpha\boldsymbol{I})\hat{\boldsymbol{x}}(k) = \boldsymbol{A}^T\boldsymbol{W}^2\boldsymbol{b}. \tag{14}$$

In order to estimate $\hat{\boldsymbol{x}}(k)$ adaptively, (14) is solved iteratively as

$$\boldsymbol{\Phi}(k)\hat{\boldsymbol{x}}(k) = \boldsymbol{\theta}(k) \tag{15}$$
$$\boldsymbol{\Phi}(k) = \boldsymbol{A}^T\boldsymbol{W}^2\boldsymbol{A} + \alpha\boldsymbol{I} = \sum_{i=1}^{k}\lambda^{k-i}\boldsymbol{u}(i)\boldsymbol{u}^T(i) + \alpha\boldsymbol{I}. \tag{16}$$
$$\boldsymbol{\theta}(k) = \boldsymbol{A}^T\boldsymbol{W}^2\boldsymbol{b} = \sum_{i=1}^{k}\lambda^{k-i}d(i)\boldsymbol{u}(i) \tag{17}$$

where

$$\hat{\boldsymbol{x}}(k) = \boldsymbol{\Phi}^{-1}(k)[\lambda\boldsymbol{\Phi}(k-1)\hat{\boldsymbol{x}}(k-1) + d(k)\boldsymbol{u}(k)] \tag{18}$$

is obtained by substituting (17) into (15). On the other hand, since (16) can be expressed as

$$\lambda\boldsymbol{\Phi}(k-1) = \boldsymbol{\Phi}(k) - \boldsymbol{u}(k)\boldsymbol{u}^T(k) - \alpha(1-\lambda)\boldsymbol{I}$$

and substitued into (18) to obtain an equation that can be used to update the parameter vector

$$\hat{\boldsymbol{x}}(k) = [\boldsymbol{I} - \alpha(1-\lambda)\boldsymbol{\Phi}^{-1}(k)]\hat{\boldsymbol{x}}(k-1) + \boldsymbol{\Phi}^{-1}(k)\boldsymbol{u}(k)\nu(k) \tag{19}$$

the LRLS algorithm is obtained as follows [8], [9]:

[LRLS algorithm]

Initialize the algorithm by setting

$$\boldsymbol{\Phi}(0) = \alpha\boldsymbol{I}, \hat{\boldsymbol{x}}(0) = \boldsymbol{O}. \tag{20}$$

For each instant of time $k = 1, 2, \ldots$, compute

$$\boldsymbol{\Phi}(k) = \lambda\boldsymbol{\Phi}(k-1) + \boldsymbol{u}(k)\boldsymbol{u}^T(k) + \alpha(1-\lambda)\boldsymbol{I} \tag{21}$$
$$\nu(k) = d(k) - \boldsymbol{u}^T(k)\hat{\boldsymbol{x}}(k-1) \tag{22}$$
$$\hat{\boldsymbol{x}}(k) = [\boldsymbol{I} - \alpha(1-\lambda)\boldsymbol{\Phi}^{-1}(k)]\hat{\boldsymbol{x}}(k-1) + \boldsymbol{\Phi}^{-1}(k)\boldsymbol{u}(k)\nu(k). \tag{23}$$

The LRLS algorithm is numerically stable for time-varying signals even in the worst case, although the estimated parameters by using the LRLS algorithm do not converge to the optimum values in the LS sense because of the ridge parameter $\alpha$ [8], [9]. Moreover, its computational complexity is on the order of $n^3$ for $0 < \lambda < 1$ since direct computation of the matrix inverse at each time is required.

### *B. Extended Criterion of Chandrasekaran et al. and LRLS Algorithm*

We consider a situation in which the uncertainties in $\boldsymbol{A}$ are very small, say, $\boldsymbol{A}$ is almost known exactly. We assume further that $\boldsymbol{b}$ is far from the column space of $\boldsymbol{A}$. In this case, it is not difficult to imagine that the TLS solution will need to rotate $(\boldsymbol{A}, \boldsymbol{b})$ into $(\hat{\boldsymbol{A}}, \hat{\boldsymbol{b}})$ and may therefore end up with an overly corrected approximation for $\boldsymbol{A}$, despite the fact that $\boldsymbol{A}$ is almost exact. These facts lead to a motivation that Chandrasekaran *et al.* have introduced a new parameter estimation formulation with prior bounds on the size of the allowable corrections to the data [7].

The criterion of Chandrasekaran *et al.* is expressed as

$$\min_{\hat{\boldsymbol{x}}(k)}\max[\|(\boldsymbol{A} + \delta\boldsymbol{A})\hat{\boldsymbol{x}}(k) - (\boldsymbol{b} + \delta\boldsymbol{b})\|_2 : \|\delta\boldsymbol{A}\|_2 \le \eta, \|\delta\boldsymbol{b}\|_2 \le \eta_b] \tag{24}$$

where $\boldsymbol{A}$ and $\boldsymbol{b}$ are defined by (2) and (3), respectively. It is assumed that the true coefficient matrix is $\boldsymbol{A} + \delta\boldsymbol{A}$ and that the upper bound $\eta$ on the 2-induced norm of the perturbation $\delta\boldsymbol{A}$ is known. Likewise, it is assumed that the true observation vector is $\boldsymbol{b} + \delta\boldsymbol{b}$ and that the upper bound $\eta_b$ on the Euclidean norm of the perturbation $\delta\boldsymbol{b}$ is known. We then pose the problem (24) of finding an estimate that performs well for any allowed perturbation $(\delta\boldsymbol{A}, \delta\boldsymbol{b})$. We note that if $\eta = 0 = \eta_b$, then the problem (24) reduces to a standard least-squares problem. Therefore, we will assume throughout that $\eta > 0$ [7].

Accordingly, we may define the extended criterion (25) of Chandrasekaran *et al.* with the exponential weighting factor to find a relationship between the equation obtained by the extended criterion of Chandrasekaran *et al.* and the equation from which the LRLS algorithm is derived.

$$\min_{\hat{\boldsymbol{x}}(k)}\max[\|\boldsymbol{W}\{(\boldsymbol{A} + \delta\boldsymbol{A})\hat{\boldsymbol{x}}(k) - (\boldsymbol{b} + \delta\boldsymbol{b})\}\|_2 : \|\delta\boldsymbol{A}\|_2 \le \eta, \|\delta\boldsymbol{b}\|_2 \le \eta_b]. \tag{25}$$

In (25), the matrix $\boldsymbol{W}$ is defined by (5). We show how to reduce the min-max problem (25) to reduce it to a standard minimization problem. To begin with, we note that

$$\begin{aligned}&\|\boldsymbol{W}\{(\boldsymbol{A} + \delta\boldsymbol{A})\hat{\boldsymbol{x}}(k) - (\boldsymbol{b} + \delta\boldsymbol{b})\}\|_2\\ &\quad\le \|\boldsymbol{W}(\boldsymbol{A}\hat{\boldsymbol{x}}(k) - \boldsymbol{b})\|_2 + \|\boldsymbol{W}\|_2\|\delta\boldsymbol{A}\|_2\|\hat{\boldsymbol{x}}(k)\|_2 + \|\boldsymbol{W}\|_2\|\delta\boldsymbol{b}\|_2\\ &\quad\le \|\boldsymbol{W}(\boldsymbol{A}\hat{\boldsymbol{x}}(k) - \boldsymbol{b})\|_2 + \eta\|\hat{\boldsymbol{x}}(k)\|_2 + \eta_b\end{aligned}$$

which provides an upper bound for $\|\boldsymbol{W}\{(\boldsymbol{A} + \delta\boldsymbol{A})\hat{\boldsymbol{x}}(k) - (\boldsymbol{b} + \delta\boldsymbol{b})\}\|_2$ because of $\|\boldsymbol{W}\|_2 = 1$. However, this upper bound is in fact achievable, i.e., there exist $(\delta\boldsymbol{A}, \delta\boldsymbol{b})$ for which

$$\|\boldsymbol{W}\{(\boldsymbol{A} + \delta\boldsymbol{A})\hat{\boldsymbol{x}}(k) - (\boldsymbol{b} + \delta\boldsymbol{b})\}\|_2 = \|\boldsymbol{W}(\boldsymbol{A}\hat{\boldsymbol{x}}(k) - \boldsymbol{b})\|_2 + \eta\|\hat{\boldsymbol{x}}(k)\|_2 + \eta_b.$$

To see that this is indeed the case, we choose $\delta\boldsymbol{A}$ as the rank one matrix

$$\delta\boldsymbol{A}^o = \frac{(\boldsymbol{A}\hat{\boldsymbol{x}}(k) - \boldsymbol{b})}{\|\boldsymbol{W}(\boldsymbol{A}\hat{\boldsymbol{x}}(k) - \boldsymbol{b})\|_2}\frac{\hat{\boldsymbol{x}}^T(k)}{\|\hat{\boldsymbol{x}}(k)\|_2}\eta \tag{26}$$

and choose $\delta\boldsymbol{b}$ as the vector

$$\delta\boldsymbol{b}^o = -\frac{(\boldsymbol{A}\hat{\boldsymbol{x}}(k) - \boldsymbol{b})}{\|\boldsymbol{W}(\boldsymbol{A}\hat{\boldsymbol{x}}(k) - \boldsymbol{b})\|_2}\eta_b. \tag{27}$$

For these choices of perturbations in $\boldsymbol{A}$ and $\boldsymbol{b}$, it follows that

$$\boldsymbol{W}(\boldsymbol{A}\hat{\boldsymbol{x}}(k) - \boldsymbol{b}), \quad \boldsymbol{W}\delta\boldsymbol{A}^o\hat{\boldsymbol{x}}(k), \text{and } \boldsymbol{W}\delta\boldsymbol{b}^o$$

are collinear vectors that point in the same direction. Hence

$$\begin{aligned}&\|\boldsymbol{W}\{(\boldsymbol{A}+\delta\boldsymbol{A}^o)\hat{\boldsymbol{x}}(k)-(\boldsymbol{b}+\delta\boldsymbol{b}^o)\}\|_2\\&=\|\boldsymbol{W}(\boldsymbol{A}\hat{\boldsymbol{x}}(k)-\boldsymbol{b})+\boldsymbol{W}\delta\boldsymbol{A}^o\hat{\boldsymbol{x}}(k)-\boldsymbol{W}\delta\boldsymbol{b}^o\|_2\\&=\|\boldsymbol{W}(\boldsymbol{A}\hat{\boldsymbol{x}}(k)-\boldsymbol{b})\|_2+\|\boldsymbol{W}\delta\boldsymbol{A}^o\hat{\boldsymbol{x}}(k)\|_2+\|\boldsymbol{W}\delta\boldsymbol{b}^o\|_2\\&=\|\boldsymbol{W}(\boldsymbol{A}\hat{\boldsymbol{x}}(k)-\boldsymbol{b})\|_2+\eta\|\hat{\boldsymbol{x}}(k)\|_2+\eta_b\end{aligned}$$

which is the desired upper bound. We therefore conclude that

$$\begin{aligned}\max_{\|\delta\boldsymbol{A}\|_2\le\eta,\|\delta\boldsymbol{b}\|_2\le\eta_b}&\|\boldsymbol{W}\{(\boldsymbol{A}+\delta\boldsymbol{A})\hat{\boldsymbol{x}}(k)-(\boldsymbol{b}+\delta\boldsymbol{b})\}\|_2\\&=\|\boldsymbol{W}(\boldsymbol{A}\hat{\boldsymbol{x}}(k)-\boldsymbol{b})\|_2+\eta\|\hat{\boldsymbol{x}}(k)\|_2+\eta_b\end{aligned}\quad(28)$$

which establishes the following result.

*Lemma 2.1:* The min-max problem (25) is equivalent to the following minimization problem. Given $\boldsymbol{A}\in R^{k\times n}$, with $k\ge n$, $\boldsymbol{b}\in R^k$, $\boldsymbol{W}$ in (5), and non-negative real numbers $(\eta,\eta_b)$, determine, if possible, an $\hat{\boldsymbol{x}}(k)$ that solves

$$\min_{\hat{\boldsymbol{x}}(k)}(\|\boldsymbol{W}(\boldsymbol{A}\hat{\boldsymbol{x}}(k)-\boldsymbol{b})\|_2+\eta\|\hat{\boldsymbol{x}}(k)\|_2+\eta_b).\quad(29)$$

Lemma 2.1 with the forgetting factor $\lambda=1$, i.e., $\boldsymbol{W}=\boldsymbol{I}$, coincides with [7, Lemma 3.1]. Moreover, when we introduce a matrix $\boldsymbol{A}_W=\boldsymbol{W}\boldsymbol{A}$ and a vector $\boldsymbol{b}_W=\boldsymbol{W}\boldsymbol{b}$, we can rewrite the problem (29) as

$$\min_{\hat{\boldsymbol{x}}(k)}(\|\boldsymbol{A}_W\hat{\boldsymbol{x}}(k)-\boldsymbol{b}_W\|_2+\eta\|\hat{\boldsymbol{x}}(k)\|_2+\eta_b).\quad(30)$$

Since the solutions of the problem (30) can be derived in the same way by which the solutions of the problem (30) with $\boldsymbol{W}=\boldsymbol{I}$ have been expressed in [7], we can utilize [7, Th. 3.6.] to solve (30) as follows.

- Introduce the SVD of $\boldsymbol{A}_W$

$$\boldsymbol{A}_W=\boldsymbol{U}\begin{bmatrix}\boldsymbol{\Sigma}\\\boldsymbol{O}\end{bmatrix}\boldsymbol{V}^T\quad(31)$$

  where $\boldsymbol{U}\in R^{k\times k}$ and $\boldsymbol{V}\in R^{n\times n}$ are orthogonal, and $\boldsymbol{\Sigma}=\mathrm{diag}(\sigma_1,\ldots,\sigma_n)$ is diagonal, where

$$\sigma_1\ge\sigma_2\ge\cdots\ge\sigma_n>0$$

  are the singular values of $\boldsymbol{A}_W$.
- Partition the vector $\boldsymbol{U}^T\boldsymbol{b}_W$ into

$$\begin{bmatrix}\boldsymbol{b}_{W,1}\\\boldsymbol{b}_{W,2}\end{bmatrix}=\boldsymbol{U}^T\boldsymbol{b}_W\quad(32)$$

  where $\boldsymbol{b}_{W,1}\in R^n$ and $\boldsymbol{b}_{W,2}\in R^{k-n}$.
- Introduce the secular function

$$\mathcal{G}(\alpha)=\boldsymbol{b}_{W,1}^T(\boldsymbol{\Sigma}^2-\eta^2\boldsymbol{I})(\boldsymbol{\Sigma}^2+\alpha\boldsymbol{I})^{-2}\boldsymbol{b}_{W,1}-\frac{\eta^2}{\alpha^2}\|\boldsymbol{b}_{W,2}\|_2^2.\quad(33)$$

- Define

$$\tau=\frac{\|\boldsymbol{A}_W^T\boldsymbol{b}_W\|_2}{\|\boldsymbol{b}_W\|_2}.$$

  Assume that $\boldsymbol{b}_W$ does not belong to the column span of $\boldsymbol{A}_W$. If $\eta<\tau$, then the unique solution is

$$\hat{\boldsymbol{x}}(k)=\left(\boldsymbol{A}_W^T\boldsymbol{A}_W+\alpha\boldsymbol{I}\right)^{-1}\boldsymbol{A}_W^T\boldsymbol{b}_W\quad(34)$$

  where $\alpha$ is the unique positive root of the secular equation $\mathcal{G}(\alpha)=0$.[1]

[1]The solutions in the other cases were given in [7].

Thus, when $\boldsymbol{b}_W$ does not belong to the column span of $\boldsymbol{A}_W$ and $\eta$ is smaller than $\tau$, the solution (34) to (25) is equivalent to the LRLS solution (14) obtaind by the criterion (13).

Accordingly, when we consider a situation in which $\boldsymbol{A}$ is almost known exactly and $\boldsymbol{b}$ is far from the column space of $\boldsymbol{A}$, in addition to the above case, we may conclude that the LRLS algorithm gives more accurate estimation parameters than the RLS algorithm.

### C. Some Implementations of LRLS Algorithm

The computational comlexity of the LRLS filter is on the order of $n^3$ since it requires the inversion of the matrix $\boldsymbol{\Phi}(k)$ at every instant of time. Thus, we introduce an $O(n^2)$ algorithm to update the eigendecomposition of rank-one matrix updates [12].[2] The algorithm developed in [12] employs a method to update the eigendecomposition of rank-one matrix updates [13], [14] and the fast multipole method (FMM) [15] to update each eigenvector. Therefore, the computational complexity of the LRLS algorithm may be reduced to $O(n^2)$ by using the method discussed in [12] as follows.

Let $\boldsymbol{Q}_{k-1}\boldsymbol{D}_{k-1}\boldsymbol{Q}_{k-1}^T$ denote the eigendecomposition of $\boldsymbol{\Phi}(k-1)$. In addition, let $\boldsymbol{Q}'_{k-1}\boldsymbol{D}'_{k-1}\boldsymbol{Q}'^T_{k-1}$ denote the eigendecomposition of the rank-one update $\boldsymbol{\Phi}(k-1)+\lambda^{-1}\boldsymbol{u}(k)\boldsymbol{u}^T(k)$. The method developd in [12] allows us to update $\boldsymbol{D}_{k-1}$ to $\boldsymbol{D}'_{k-1}$ [13], [14] and $\boldsymbol{Q}_{k-1}$ to $\boldsymbol{Q}'_{k-1}$ [12], [13] in $O(n^2)$. Then, we can recognize that

$$\boldsymbol{Q}_k=\boldsymbol{Q}'_{k-1},\boldsymbol{D}_k=\lambda\boldsymbol{D}'_{k-1}+\alpha(1-\lambda)\boldsymbol{I}.\quad(35)$$

This leads us to update $\hat{\boldsymbol{x}}(k-1)$ to $\hat{\boldsymbol{x}}(k)$ in (19) as

$$\begin{aligned}\hat{\boldsymbol{x}}(k)=\boldsymbol{Q}_k\left[\boldsymbol{I}-\alpha(1-\lambda)\boldsymbol{D}_k^{-1}\right]\boldsymbol{Q}_k^T\hat{\boldsymbol{x}}(k-1)\\+\boldsymbol{Q}_k\boldsymbol{D}_k^{-1}\boldsymbol{Q}_k^T\boldsymbol{u}(k)\nu(k).\end{aligned}\quad(36)$$

Therefore, the method discussed in [12] allows us to update $\hat{\boldsymbol{x}}(k-1)$ to $\hat{\boldsymbol{x}}(k)$ in $O(n^2)$ in this way.

Moreover, we express an $O(n)$ algorithm to update the estimation parameter vector $\hat{\boldsymbol{x}}(k)$ approximately. The discussed algorithm is the leaky least-mean-square (LLMS) filter that further stabilizes the digital implementation of the LMS algorithm. In the LLMS algorithm, the criterion

$$J(k)=\{d(k)-\boldsymbol{u}^T(k)\hat{\boldsymbol{x}}(k)\}^2+\alpha_{\mathrm{LLMS}}\|\hat{\boldsymbol{x}}(k)\|_2^2\quad(37)$$

is minimized with respect to the estimation parameter vector $\hat{\boldsymbol{x}}(k)$, where $\alpha_{\mathrm{LLMS}}$ is a positive control parameter. The minimization on (37) yields the following time update for the estimation parameter vector $\hat{\boldsymbol{x}}(k)$:

[LLMS algorithm]

Initialize the algorithm by setting

$$\hat{\boldsymbol{x}}(0)=\boldsymbol{O}.\quad(38)$$

For each instant of time, $k=1,2,\ldots$, compute

$$e(k)=d(k)-\boldsymbol{u}^T(k)\hat{\boldsymbol{x}}(k-1)\quad(39)$$

$$\hat{\boldsymbol{x}}(k)=(1-\mu\alpha_{\mathrm{LLMS}})\hat{\boldsymbol{x}}(k-1)+\mu e(k)\boldsymbol{u}(k).\quad(40)$$

where $\mu$ is the step-size parameter. The leakage factor $(1-\mu\alpha_{\mathrm{LLMS}})$ associated with the first term on the right side of (40) prevents the occurence of overflow in a limited-precision environment by providing a compromise between minimizing the mean-squared error and containing the energy in the estimation parameters [1]. Although the estimation parameter vector $\hat{\boldsymbol{x}}(k)$ of the LLMS algorithm does not converge to the LS solution, we can expect that the vector $\hat{\boldsymbol{x}}(k)$ of (40) ap-

[2]This method has already been indicated in [6].

proximately converges to the solution of the LRLS algorithm, as shown below.

Taking the mathematical expectation of both sides of (15), we get

$$E[\boldsymbol{\Phi}(k)\hat{\boldsymbol{x}}(k)] = E[\boldsymbol{\theta}(k)]. \tag{41}$$

For the right side of (41), we introduce an assumption that the filter input $u(i)$ and the desired response $d(i)$ are single realizations of jointly wide-sense stationary stochastic processes, both with zero mean, to rewrite this term as

$$\begin{aligned} E[\boldsymbol{\theta}(k)] &= \sum_{i=1}^{k} \lambda^{k-i} E[d(i)\boldsymbol{u}(i)] \\ &= \sum_{i=1}^{k} \lambda^{k-i} \boldsymbol{p} \end{aligned} \tag{42}$$

where $\boldsymbol{p}$ denotes the cross-correlation vector between the filter input and the desired response. For the left side of (41), we introduce an assumption that the vector $\hat{\boldsymbol{x}}(k)$ is statistically independent from $\boldsymbol{\Phi}(k)$ to rewrite this term as

$$\begin{aligned} E[\boldsymbol{\Phi}(k)\hat{\boldsymbol{x}}(k)] &= E[\boldsymbol{\Phi}(k)]E[\hat{\boldsymbol{x}}(k)] \\ &= \left(\sum_{i=1}^{k} \lambda^{k-i} E[\boldsymbol{u}(i)\boldsymbol{u}^T(i)] + \alpha \boldsymbol{I}\right) \boldsymbol{x}_{o,\mathrm{LRLS}} \\ &= \left(\sum_{i=1}^{k} \lambda^{k-i} \boldsymbol{R} + \alpha \boldsymbol{I}\right) \boldsymbol{x}_{o,\mathrm{LRLS}} \end{aligned} \tag{43}$$

where $\boldsymbol{R}$ denotes the correlation matrix of the wide-sense stationary stochastic process $u(i)$. Thus, we obtain the following equation for the mean value $\boldsymbol{x}_{o,\mathrm{LRLS}} = E[\hat{\boldsymbol{x}}(k)]$ with $0 < \lambda < 1$:

$$\left(\frac{1}{1-\lambda}\boldsymbol{R} + \alpha \boldsymbol{I}\right) \boldsymbol{x}_{o,\mathrm{LRLS}} = \frac{1}{1-\lambda}\boldsymbol{p}, as\ k \to \infty. \tag{44}$$

Moreover, we get the similar equation to (44) for the LLMS algorithm as

$$(\boldsymbol{R} + \alpha_{\mathrm{LLMS}} \boldsymbol{I}) \boldsymbol{x}_{o,\mathrm{LLMS}} = \boldsymbol{p}. \tag{45}$$

For $\boldsymbol{x}_{o,\mathrm{LRLS}} = \boldsymbol{x}_{o,\mathrm{LLMS}}$ in (44) and (45), we obtain

$$\alpha_{\mathrm{LLMS}} = (1-\lambda)\alpha, \quad 0 < \lambda < 1. \tag{46}$$

Therefore, we may regard (46) as a guide for the estimation parameter vector of the LLMS algorithm to converge to that of the LRLS algorithm with $0 < \lambda < 1$ in the mean sense.

To get a condition about the step-size parameter $\mu$ under which the mean value $E[\hat{\boldsymbol{x}}(k)]$ of the estimation parameter vector $\hat{\boldsymbol{x}}(k)$ of the LLMS algorithm converges to the solution $\boldsymbol{x}_{o,\mathrm{LLMS}}$, we decompose the symmetric matrix $\boldsymbol{R}$ into $\boldsymbol{R} = \boldsymbol{Q}\boldsymbol{D}\boldsymbol{Q}^T$, where $\boldsymbol{D} = \mathrm{diag}(\rho_1, \rho_2, \cdots, \rho_n), \rho_1 \le \rho_2 \le \ldots \le \rho_n$ is the matrix of the eigenvalues, and $\boldsymbol{Q}$ is the matrix of the eigenvectors of $\boldsymbol{R}$. Introducing the parameter error vector $\boldsymbol{e}_x(k) = \hat{\boldsymbol{x}}(k) - \boldsymbol{x}_{o,\mathrm{LLMS}}$, and defining the rotated vectors

$$\boldsymbol{e}'_x(k) = \boldsymbol{Q}^T \boldsymbol{e}_x(k), \quad \boldsymbol{u}'(k) = \boldsymbol{Q}^T \boldsymbol{u}(k), \quad \boldsymbol{x}'_{o,\mathrm{LLMS}} = \boldsymbol{Q}^T \boldsymbol{x}_{o,\mathrm{LLMS}}$$

then using the relation

$$d(k) = \boldsymbol{u}^T(k)\boldsymbol{x}_{o,\mathrm{LLMS}} + e_o(k)$$

(40) can be rewritten as

$$\begin{aligned} \boldsymbol{e}'_x(k) = [\boldsymbol{I} - \mu(\alpha_{\mathrm{LLMS}}\boldsymbol{I} + \boldsymbol{u}'(k)\boldsymbol{u}'^T(k))]\boldsymbol{e}'_x(k-1) \\ - \mu\alpha_{\mathrm{LLMS}}\boldsymbol{x}'_{o,\mathrm{LLMS}} + \mu e_o(k)\boldsymbol{u}'(k). \end{aligned} \tag{47}$$

If we take the expected value of both sides of (47) with the common independence assumption of $\boldsymbol{e}'_x(k)$ and $\boldsymbol{u}'(k)$, and the relation

$$E[\boldsymbol{u}(k)e_o(k)] = \alpha_{\mathrm{LLMS}}\boldsymbol{x}_{o,\mathrm{LLMS}}$$

obtained by minimizing the expected value of (37) with respect to $\boldsymbol{x}_{o,\mathrm{LLMS}} = E[\hat{\boldsymbol{x}}(k)]$, (47) can be rewritten as

$$E[\boldsymbol{e}'_x(k)] = [\boldsymbol{I} - \mu(\alpha_{\mathrm{LLMS}}\boldsymbol{I} + \boldsymbol{D})]E[\boldsymbol{e}'_x(k-1)]. \tag{48}$$

Clearly, from (48), the boundedness of the expected value of all modes is guaranteed by the following condition on the step-size $\mu$:

$$0 < \mu < \frac{2}{\rho_n + \alpha_{\mathrm{LLMS}}}. \tag{49}$$

Relation (49) is identical to the conventional condition for the mean value $E[\hat{\boldsymbol{x}}(k)]$ of the estimation parameter vector $\hat{\boldsymbol{x}}(k)$ of the LLMS algorithm to converge to the Wiener solution $\boldsymbol{x}_W = \boldsymbol{R}^{-1}\boldsymbol{p}$, although a nonzero leakage factor $\alpha_{\mathrm{LLMS}}$ results in some nonzero steady-state coefficient bias [16].

Accordingly, we can use the LLMS algorithm as an $O(n)$ algorithm to track the estimation parameter vector of the LRLS algorithm approximately. However, we see, from (48), that when the eigenvalues of the correlation matrix $\boldsymbol{R}$ are widely spread with $\rho_1 + \alpha_{\mathrm{LLMS}} \ll \rho_n + \alpha_{\mathrm{LLMS}}$, the time taken by the average parameter vector to converge is primarily limited by the smallest eigenvalues and $\alpha_{\mathrm{LLMS}}$.

## III. Experimental Results

In this section, by means of computer experiments with MATLAB, we show that the LRLS and the LLMS algorithms give more accurate estimation parameters than the RLS algorithm when the method of Chandrasekaran *et al.* is more useful than that of LS. In addition, we demonstrate that the LLMS algorithm can be effectively introduced into the noise reduction system for noisy speech signals proposed in [10] and [11].

### *A. Fundamental Example 1: Batch Processing, Eigenvalue Spread $\chi(\boldsymbol{R}) = 1$*

In this experiment, we consider a system identification setup illustrated in Fig. 1 to support the theoretical results of the previous section. The unknown system was a lowpass filter of order $n = 29$ whose passband was from the normalized angular frequency $\omega = 0$ to $\omega = 0.4\pi$ and whose stopband was from $\omega = 0.5\pi$ to $\omega = \pi$. Table I lists the coefficients of the lowpass filter.

The input signal to the unknown system $u_0(i), i = 1, 2, \ldots, k$ and the observed input signal $u(i), i = 1, 2, \ldots, k$ were defined as

$$u(i) = u_0(i) + v_u(i) \tag{50}$$

where $v_u(i)$ denoted a white noise process of zero mean and variance $\sigma_u^2$. The output signal from the unknown system $d_0(i), i = 1, 2, \ldots, k$ and the observed output signal $d(i), i = 1, 2, \ldots, k$ were defined, respectively, as

$$d_0(i) = \boldsymbol{x}^T \boldsymbol{u}_0(i) = \sum_{l=1}^{n} x_l u_0(i-l+1), \tag{51}$$

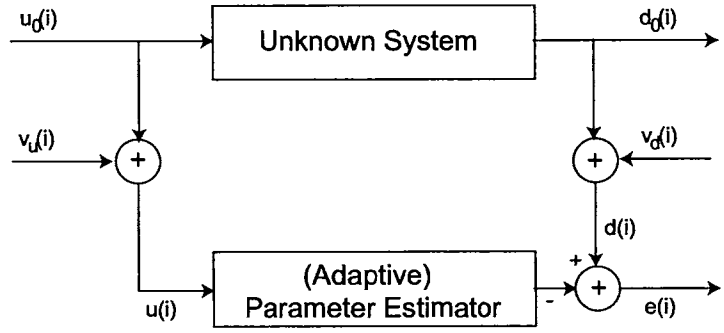


Fig. 1. System identification setup.

TABLE I
COEFFICIENTS OF THE LOWPASS FILTER

| | |
|---|---|
| $x_1 = x_{29}$ | -0.00555489638749 |
| $x_2 = x_{28}$ | -0.02442499477047 |
| $x_3 = x_{27}$ | -0.01760414047874 |
| $x_4 = x_{26}$ | 0.00757896430410 |
| $x_5 = x_{25}$ | 0.02238212553468 |
| $x_6 = x_{24}$ | -0.00208414909154 |
| $x_7 = x_{23}$ | -0.03175784245076 |
| $x_8 = x_{22}$ | -0.01072978865583 |
| $x_9 = x_{21}$ | 0.04181657535078 |
| $x_{10} = x_{20}$ | 0.03540281323908 |
| $x_{11} = x_{19}$ | -0.05049812267987 |
| $x_{12} = x_{18}$ | -0.08762989569786 |
| $x_{13} = x_{17}$ | 0.05635575263757 |
| $x_{14} = x_{16}$ | 0.31188616494361 |
| $x_{15}$ | 0.44157150419284 |

$$d(i) = d_0(i) + v_d(i) \tag{52}$$

where $v_d(i)$ denoted another white noise process of zero mean and variance $\sigma_d^2$. The signal $u_0(i)$ was a white noise process of zero mean and unity variance for the correlation matrix $\boldsymbol{R}$ to have the smallest 2-induced norm condition number[3] $\chi(\boldsymbol{R}) = (\rho_n)/(\rho_1) = (1+\sigma_u^2)/(1+\sigma_u^2) = 1$.

The SNRs of the input to and the output from the unknown system were given by

$$\mathrm{SNR_{in}} = 10\log_{10}\frac{\sum_{i=1}^{k} u_0^2(i)}{\sum_{i=1}^{k} v_u^2(i)}$$
$$\mathrm{SNR_{out}} = 10\log_{10}\frac{\sum_{i=1}^{k} d_0^2(i)}{\sum_{i=1}^{k} v_d^2(i)}.$$

The values of $\mathrm{SNR_{in}}$ and $\mathrm{SNR_{out}}$ were 30 and 0 dB, respectively, to satisfy the condition where the solution of Chandrasekaran *et al.* was more accurate than the solutions of both LS and TLS, i.e., where the matrix $\boldsymbol{A}$ was almost known exactly, and the vector $\boldsymbol{b}$ was far from the column space of $\boldsymbol{A}$.

In addition, since the matrices $\boldsymbol{A}$ and $\boldsymbol{b}$ include the noises $v_u(i)$ and $v_d(i)$, the matrices $\delta\boldsymbol{A}$ and $\delta\boldsymbol{b}$ of the true matrices $\boldsymbol{A}+\delta\boldsymbol{A}$ and $\boldsymbol{b}+\delta\boldsymbol{b}$ in (25) may be assumed to be

$$\boldsymbol{v}_u^T(i) = [v_u(i), v_u(i-1), \ldots, v_u(i-n+1)] \tag{53}$$
$$\delta\boldsymbol{A}^T = -[\boldsymbol{v}_u(1), \boldsymbol{v}_u(2), \ldots, \boldsymbol{v}_u(k)] \tag{54}$$

[3]The correlation matrix $\boldsymbol{R}$ is ill conditioned if the condition number $\chi(\boldsymbol{R}) = \rho_n/\rho_1$ or the eigenvalue spread is large [1], [4].

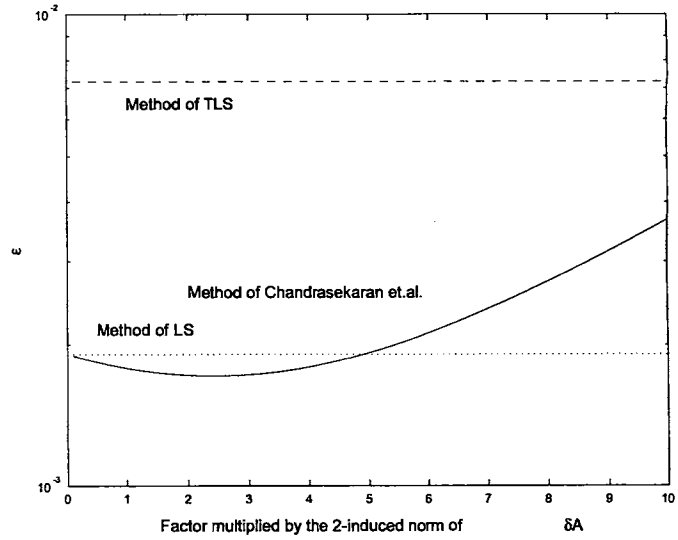


Fig. 2. Comparison of the solution (34) with LS and TLS solutions.

$$\delta\boldsymbol{b}^T = -[v_d(1), v_d(2), \ldots, v_d(k)]. \tag{55}$$

Moreover, we assumed the norms $\|\delta\boldsymbol{A}\|_2$ and $\|\delta\boldsymbol{b}\|_2$ to be estimates of $\eta$ and $\eta_b$ in (25), although the matrices $\delta\boldsymbol{A}$ in (54) and $\delta\boldsymbol{b}$ in (55) were different from the optimum matrices $\delta\boldsymbol{A}^o$ and $\delta\boldsymbol{b}^o$ in (26) and (27), respectively.

In the experiment, the solutions (34) with $\lambda = 1$ of the criterion of Chandrasekaran *et al.*, where the estimation value $\hat{\eta}$ of $\eta$ in (25) varied from $10^{-1}\|\delta\boldsymbol{A}\|_2$ to $10\|\delta\boldsymbol{A}\|_2$, were compared with LS and TLS solutions. An index parameter defined by

$$\epsilon = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{x}_i)^2 \tag{56}$$

has been used as the measure of the estimation accuracy. The ensemble averaging has been performed over 100 independent trials of the experiment.

When we used the observed input $u(i)$, which included the noise $v_u(i)$, the LS solution was not the optimum solution due to the criterion of Chandrasekaran *et al.* [7]. In addition, the used signals satisfied the conditions for which (34) was the solution of (25). Fig. 2 indicates that the solutions (34) with appropriate $\hat{\eta}$s perform better than the solutions of LS and TLS.

### *B. Fundamental Example 2: Adaptive Filter, Eigenvalue Spread $\chi(\boldsymbol{R}) \geq 1$*

In this experiment, we consider the same system identification setup illustrated in Fig. 1. Only the input signal to the unknown system $u_0(i)$

TABLE II
USED CONDITIONS IN THE EXPERIMENT

| $b_1$ | 0 | 0.4 | 0.6 |
|---|---|---|---|
| $\rho_{1,u_0}$ | 1 | 0.3644 | 0.1666 |
| $\rho_{n,u_0}$ | 1 | 1.956 | 2.553 |
| $\sigma_u^2$ | 0.001035 | 0.001211 | 0.001350 |
| $\rho_1 = \rho_{1,u_0} + \sigma_u^2$ | 1.001 | 0.3656 | 0.1679 |
| $\rho_n = \rho_{n,u_0} + \sigma_u^2$ | 1.001 | 1.957 | 2.555 |
| $\chi(\boldsymbol{R}) = \rho_n/\rho_1$ | 1 | 5.352 | 15.21 |
| $\lambda$ | 0.995 | 0.995 | 0.995 |
| $\bar{\alpha}$ | 72.5 | 79.4 | 93.6 |
| $\bar{\alpha}_{LLMS}$ | 0.363 | 0.397 | 0.468 |
| $\mu$ | 0.0035 | 0.0035 | 0.0033 |

and $\mathrm{SNR}_{\mathrm{out}}$ of $-10$ dB were primarily different from that in Section III-A. The input signal $u_0(i)$ was generated by

$$u_0(i) = w(i) - b_1 w(i-1) \tag{57}$$

where $w(i)$ denoted a white noise process of zero mean and unity variance. From (57), the autocorrelation $r_{u_0}(k) = E[u_0(i)u_0(i-k)]$ is given by

$$r_{u_0}(0) = 1 + b_1^2, r_{u_0}(1) = -b_1, r_{u_0}(k) = 0 (k \geq 2). \tag{58}$$

Thus, we can obtain the eigenvalue spread

$$\chi(\boldsymbol{R}) = \chi(\boldsymbol{R}_{u_0} + \boldsymbol{R}_{v_u}) \geq 1$$

where the correlation matrices $\boldsymbol{R}_{u_0}$ and $\boldsymbol{R}_{v_u}$ are $(r_{u_0}(|i-j|))$ and $\sigma_u^2 \boldsymbol{I}$, respectively.

The LRLS and the LLMS algorithms have been numerically compared with the RLS algorithm for each of three different eigenvalue spreads. The used LRLS filter was the algorithm in Section II-A. The estimation value $\hat{\eta}$ was set to $\|\delta \boldsymbol{A}\|_2$ for the LRLS and the LLMS algorithms. The constant $\alpha_{\mathrm{LLMS}}$ was given by (46) with $\lambda$ and $\alpha$ used in the LRLS algorithm. The initial value $\delta$ for the RLS algorithm was set to $\alpha$ of the LRLS algorithm for both the initial conditions to coincide. Table II lists the other conditions in the experiment. An index parameter $\epsilon_1(j)$ defined by

$$\epsilon(j) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{x}_i(j))^2, \quad j = 1, 2, \ldots, k$$

$$\epsilon_1(j) = \frac{1}{T}\sum_{i=0}^{T-1}\epsilon(j-i), \quad j = 1, 2, \ldots, k$$

has been used with $T = 50$ as the measure of the smoothed estimation accuracy. Another index parameter $\xi_1(j)$ defined by

$$\xi(j) = (d(j) - \boldsymbol{u}^T(j)\hat{\boldsymbol{x}}(j))^2, \quad j = 1, 2, \ldots, k$$

$$\xi_1(j) = \frac{1}{T}\sum_{i=0}^{T-1}\xi(j-i), \quad j = 1, 2, \ldots, k$$

has also been used with $T = 50$ to obtain another characteristic of the algorithms. The ensemble averaging has been performed over 100 independent trials of the experiment.

When the number of iterations $k$ increases, the factor $\delta\lambda^k$ in (6) converges to 0 because of $0 < \lambda < 1$. Thus, the solution of the RLS algorithm comes apart from the optimum value in the criterion of Chandrasekaran *et al.* in this process, although the initial value $\delta$ is $\alpha$. Figs. 3–5 show that the LRLS algorithm gives more accurate estimates than the RLS algorithm. Moreover, Fig. 3 demonstrates that the LLMS algorithm performs as well as the LRLS algorithm since $(1-\mu(\rho_1+\alpha_{\mathrm{LLMS}}))^k$ of (48) is 0.0084 with $k = 1000$ and the values of $\mu$, $\rho_1$ and $\alpha_{\mathrm{LLMS}}$ in Table II. In cases of $b_1 = 0.4, 0.6$, $(1-\mu(\rho_1+\alpha_{\mathrm{LLMS}}))^k$ are 0.069 and 0.12 with $k = 1000$ and the values in Table II, respectively. Figs. 3–5 and the values of $(1-\mu(\rho_1+\alpha_{\mathrm{LLMS}}))^k$ for $b_1 = 0, 0.4, 0.6$ indicate that the rate of convergence of the LLMS algorithm slows down if the eigenvalues of the correlation matrix $\boldsymbol{R}$ are widely spread with $\rho_1 + \alpha_{\mathrm{LLMS}} \ll \rho_n + \alpha_{\mathrm{LLMS}}$.

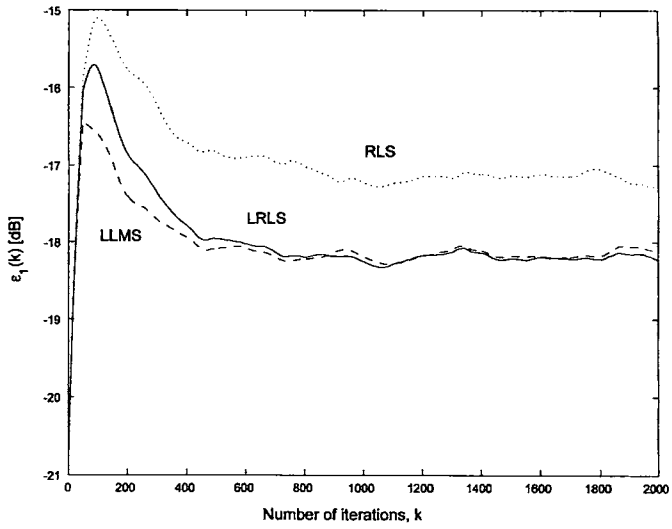


Fig. 3. Comparison of the experimental result $\epsilon_1(k)$ of the LRLS and the LLMS algorithms with the RLS algorithm of the same initial condition for $b_1 = 0$.

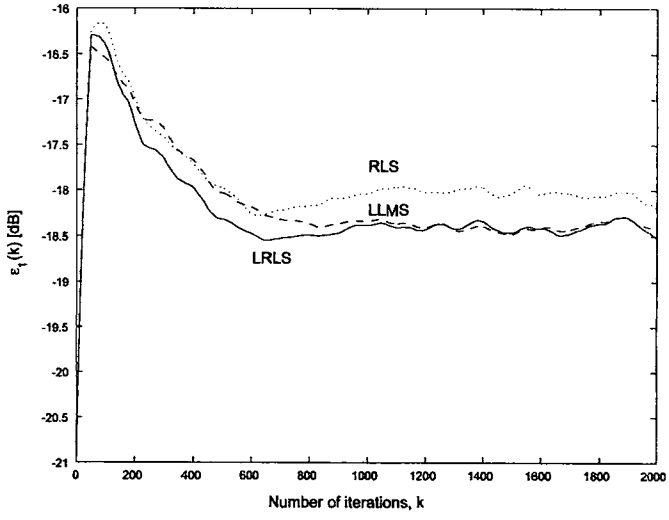


Fig. 4. Comparison of the experimental result $\epsilon_1(k)$ of the LRLS and the LLMS algorithms with the RLS algorithm of the same initial condition for $b_1 = 0.4$.

On the other hand, Fig. 6 shows that the average errors $\xi_1(k)$ of the LRLS and the LLMS algorithms are larger than those of the RLS algorithm since the LRLS and the LLMS algorithms do not minimize $\sum_{i=1}^{k}\xi(i)$ because of (13) and (37).

Accordingly, we can conclude that the LRLS and the LLMS algorithms give more accurate estimation parameters than the RLS algorithm in applications dealing with system identification where the noise

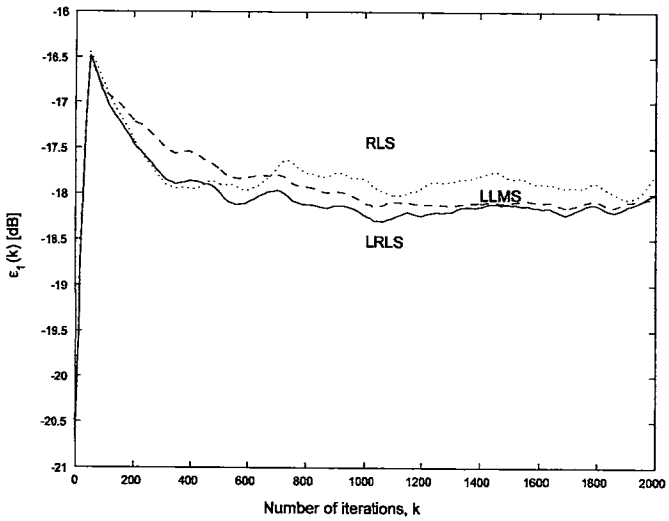


Fig. 5. Comparison of the experimental result $\epsilon_1(k)$ of the LRLS and the LLMS algorithms with the RLS algorithm of the same initial condition for $b_1 = 0.6$.

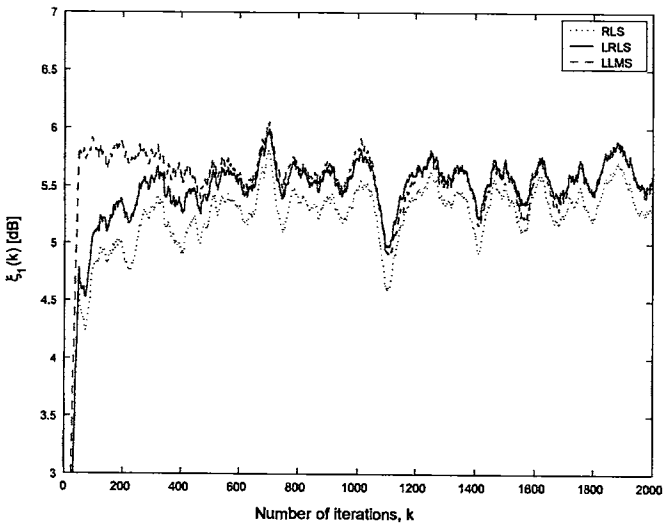


Fig. 6. Comparison of the experimental result $\xi_1(k)$ of the LRLS and the LLMS algorithms with the RLS algorithm of the same initial condition for $b_1 = 0$.

$v_u(i)$ in $\boldsymbol{A}$ is much smaller than the signal $u_0(i)$ and the noise $v_d(i)$ in $\boldsymbol{b}$ is large for $\boldsymbol{b}$ to be far from the column space of $\boldsymbol{A}$.

### C. Actual Example: Noise Reduction for Noisy Speech Signals

In this experiment, we consider a noise-reduction system [10], [11] as an actual example illustrated in Fig. 7 to reduce a background noise in a noisy speech. This system uses a noise reconstruction method based on a linear prediction, a system identification, and an adaptive line enhancer (ALE) [1], although the method proposed in [10] does not utilize the ALE.

The noisy speech is represented as

$$d(i) = s(i) + e_s(i)$$

where $s(i)$ and $e_s(i)$ are a clean speech signal and a background noise, respectively. The signals $\hat{w}(i)$, $\hat{e}_s(i)$, and $\hat{s}(i)$ are the output of a linear prediction error filter (LPEF) [1], the reconstructed noise, and the enhanced speech signal, respectively. $H_{\mathrm{LPEF}}(z)$ and $H_{\mathrm{NRF}}(z)$ denote the transfer functions of the LPEF and the noise reconstruction filter (NRF), respectively. The system proposed in [11] includes the ALE[4] for the input $\hat{w}(i)$ to the NRF to have only few speech components. Since a speech signal input to the LPEF is known as the stationary signal in a short time interval, most of it can be predicted by the linear predictor. On the other hand, the noise becomes the white signal by the LPEF. If the background noise is assumed to be generated by exciting a linear system with the white noise, it can be reconstructed from the whitened noise by estimating the transfer function $H_{\mathrm{NRF}}(z)$ of the noise-generating system. This method does not require the prior estimation of the noise spectrum. In addition, the enhanced signal does not involve the musical tones [10], [11].

In the system [10], [11] of Fig. 7, the LMS or the normalized LMS algorithm [1] is used for the NRF. Thus, the power of the enhanced speech signal $\hat{s}^2(i)$ may become small since the NRF with the LMS algorithm operates for $\hat{s}^2(i)$ to decrease. Clearly, from the previous sections, $\hat{s}^2(i)$ of the LLMS algorithm is generally larger than that of the LMS algorithm because of $\alpha_{\mathrm{LLMS}} > 0$. In addition, we can expect that the LLMS algorithm will give more accurate estimation parameters of the transfer function $H_{\mathrm{NRF}}(z)$ when the error in the input $\hat{w}(i)$ is much smaller than the true input $w(i)$ and the clean speech signal $s(i)$ is larger than the background noise $e_s(i)$.

The noise-reduction system was tested under artificially noise condition. All sound data prepared in simulations were sampled by 8 kHz. As the speech signal, a Japanese sentence pronounced by a male was used. The evaluation indexes for the noise reduction ability are $\mathrm{SNR}'_{\mathrm{in}}$ and $\mathrm{SNR}'_{\mathrm{out}}$, which are given by

$$\mathrm{SNR}'_{\mathrm{in}} = 10\log_{10}\left[\frac{\sum_{i=1}^{k} s^2(i)}{\sum_{i=1}^{k} e_s^2(i)}\right]$$

$$\mathrm{SNR}'_{\mathrm{out}} = 10\log_{10}\left[\frac{\sum_{i=1}^{k} s^2(i)}{\sum_{i=1}^{k} \{s(i) - \hat{s}(i)\}^2}\right].$$

The noise $e_s(i)$ was generated as the response of the filter with the following transfer function:

$$N(z) = \frac{1}{1 - 2\gamma\cos\theta z^{-1} + \gamma^2 z^{-2}}$$

to white noise $w(i)$, where $\gamma = 0.9$, and $\theta = \pi/4$. Incidentally, the noise $e_s(i)$ is modeled on the tunnel noise of an expressway. The adaptive algorithm for the ALE and the LPEF was the LMS algorithm, which was given by

$$\hat{\boldsymbol{x}}(i) = \hat{\boldsymbol{x}}(i-1) + \mu e(i)\boldsymbol{u}(i)$$

where

$$\hat{\boldsymbol{x}}(i) = [\hat{h}_1(i), \hat{h}_2(i), \ldots, \hat{h}_S(i)]^T$$
$$\boldsymbol{u}(i) = [d(i-\Delta), d(i-\Delta-1), \ldots, d(i-\Delta-S+1)]^T$$
$$e(i) = d(i) - \boldsymbol{u}^T(i)\hat{\boldsymbol{x}}(i-1)$$

for the ALE

$$\hat{\boldsymbol{x}}(i) = [\hat{h}'_1(i), \hat{h}'_2(i), \ldots, \hat{h}'_L(i)]^T$$
$$\boldsymbol{u}(i) = [d'(i-1), d'(i-2), \ldots, d'(i-L)]^T$$
$$e(i) = d'(i) - \boldsymbol{u}^T(i)\hat{\boldsymbol{x}}(i-1)$$

[4]In this system, the ALE works to estimate the speech $s(i)$ by suppressing the noise $e_s(i)$ [11].

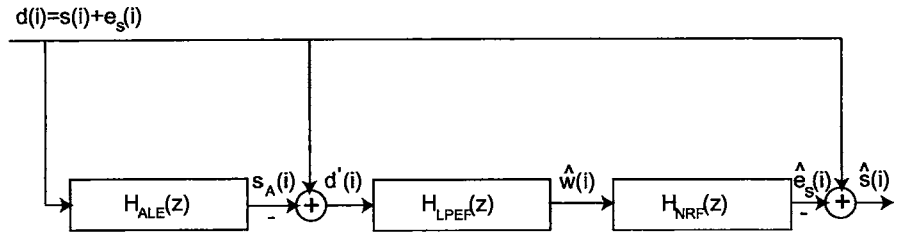


Fig. 7. Simulated noise reduction system.

TABLE III
USED PARAMETERS IN THE EXPERIMENT

| | | |
|---|---|---|
| ALE | Decorrelation parameter $\Delta$ | 40 |
| | Number of tap coefficients S | 50 |
| | Step size $\mu$ | 0.1 |
| LPEF | Number of tap coefficients L | 128 |
| | Step size $\mu$ | 0.01 |
| NRF | Number of tap coefficients M | 128 |
| | Step size $\mu$ | 0.02 |

TABLE IV
SIMULATION RESULT OF THE NOISE REDUCTION. (a) NRF IS THE LMS ALGORITHM. (b) NRF IS THE LLMS ALGORITHM OF $\alpha_{\mathrm{LLMS}} = 0.001$

| | $SNR_{out}[dB]$ | |
|---|---|---|
| $SNR_{in}[dB]$ | (A) | (B) |
| -5.0 | 0.27 | 0.18 |
| -4.0 | 1.08 | 1.00 |
| -3.0 | 1.68 | 1.61 |
| -2.0 | 2.37 | 2.34 |
| -1.0 | 3.02 | 3.00 |
| 0.0 | 3.60 | 3.62 |
| 1.0 | 4.06 | 4.12 |
| 2.0 | 4.63 | 4.74 |
| 3.0 | 5.05 | 5.21 |
| 4.0 | 5.61 | 5.82 |
| 5.0 | 5.95 | 6.20 |

for the LPEF [10], [11]. The adaptive algorithm for the NRF was the LLMS algorithm,[5] which is given by

$$\hat{\boldsymbol{x}}(i) = (1 - \mu\alpha_{\mathrm{LLMS}})\hat{\boldsymbol{x}}(i-1) + \mu e(i)\boldsymbol{u}(i)$$

where

$$\hat{\boldsymbol{x}}(i) = [\hat{x}_1(i), \hat{x}_2(i), \ldots, \hat{x}_M(i)]^T$$
$$\boldsymbol{u}(i) = [\hat{w}(i), \hat{w}(i-1), \ldots, \hat{w}(i-M+1)]^T$$
$$e(i) = d(i) - \boldsymbol{u}^T(i)\hat{\boldsymbol{x}}(i-1).$$

The difference between $\hat{w}(i)$ and $w(i)$ is unknown in the system of Fig. 7. Thus, the parameter $\alpha_{\mathrm{LLMS}}$ of the LLMS algorithm was set to 0.001 from some experiments since the upper bound $\eta$ on $\|\delta A\|_2$ of (25) was unknown. Table III lists the each parameter in the experiment. Table IV and Fig. 8 show the simulation result. We see from Table IV that the $\mathrm{SNR}'_{\mathrm{out}}$ with the LLMS algorithm as the NRF becomes larger than that with the LMS algorithm as the NRF, as the $\mathrm{SNR}'_{\mathrm{in}}$ gets larger than 0. In addition, Fig. 8 shows that the power of the enhanced speech signal with the LLMS algorithm as the NRF is somewhat larger than

[5]The LMS algorithm for the NRF was the LLMS algorithm with $\alpha_{\mathrm{LLMS}} = 0$ in the experiment.

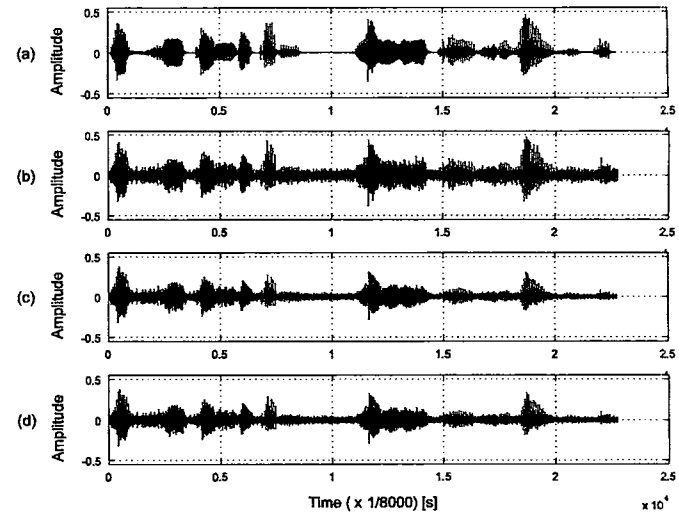


Fig. 8. Waveforms of the simulation result. (a) Clean speech. (b) Noisy speech ($\mathrm{SNR}'_{\mathrm{in}} = 5$ dB). (c) Enhanced speech with the LMS algorithm as the NRF ($\mathrm{SNR}'_{\mathrm{out}} = 5.95$ dB). (d) Enhanced speech with the LLMS algorithm of $\alpha_{\mathrm{LLMS}} = 0.001$ as the NRF ($\mathrm{SNR}'_{\mathrm{out}} = 6.20$ dB).

that with the LMS algorithm as the NRF. We can conclude that these results support the theoretical results in the previous sections in the main.

## IV. CONCLUSION

In this work, we have expressed that there is a case where the equation obtained by the extended criterion of Chandrasekaran *et al.* with the exponential weighting factor is identical to one obtained by the criterion of the LRLS algorithm. In other words, we have explained that it is possible for the LRLS algorithm to give more accurate estimation parameters than the RLS algorithm. In addition, some implementations of the LRLS filter by using the method for updating the eigendecomposition of rank-one matrix updates, or by using the LLMS algorithm, have been introduced to decrease the computational complexity of the LRLS algorithm.

Moreover, by means of computer experiments, we have shown that the LRLS and the LLMS algorithms yield more accurate estimation parameters than the RLS algorithm when the method of Chandrasekaran *et al.* is more useful than that of LS and TLS. Besides, we have demonstrated that the LLMS algorithm can be effectively introduced into the noise-reduction system for noisy speech signals proposed in [10] and [11] to support the theoretical results in this work in the main.

Future issues involve a stochastic interpretation of the experimental results shown in this work and developments of simple and fast LRLS-like filters whose computational complexity is smaller than on the order of $n^2$ and whose rate of convergence is faster than that of the LLMS algorithm for correlated inputs whose correlation matrix $\boldsymbol{R}$ has the wide eigenvalue spread with $\rho_1 + \alpha_{\mathrm{LLMS}} \ll \rho_n + \alpha_{\mathrm{LLMS}}$.

## ACKNOWLEDGMENT

The authors thank Associate Editor Prof. J. Tabrikian and the anonymous reviewers for their constructive suggestions that helped in improving this manuscript.

## REFERENCES

[1] S. Haykin, *Adaptive Filter Theory*, Second ed. Englewood Cliffs, NJ: Prentice-Hall, 1991.

[2] A. H. Sayed and T. Kailath, "A state-space approach to adaptive RLS filtering," *IEEE Signal Process. Mag.*, vol. 11, pp. 18–60, July 1994.

[3] J. A. Apolinário Jr. and P. S. R. Diniz, "A new fast QR algorithm based on a priori errors," *IEEE Signal Processing Lett.*, vol. 4, pp. 307–309, Nov. 1997.

[4] G. H. Golub and C. F. V. Loan, *Matrix Computations*, Third ed. Baltimore, MD: Johns Hopkins Univ. Press, 1996.

[5] S. Chandrasekaran, G. H. Golub, M. Gu, and A. H. Sayed, "Parameter estimation in the presence of bounded modeling errors," *IEEE Signal Processing Lett.*, vol. 4, pp. 195–197, July 1997.

[6] A. H. Sayed, A. Garulli, and S. Chandrasekaran, "A fast iterative solution for worst-case parameter estimation with bounded model uncertainties," in *Proc. Amer. Contr. Conf.*, Albuquerque, NM, June 1997, pp. 1499–1503.

[7] S. Chandrasekaran, G. H. Golub, M. Gu, and A. H. Sayed, "Parameter estimation in the presence of bounded data uncertainties," in *SIAM J. Matrix Anal. Appl.*, vol. 19, Jan. 1998, pp. 235–252.

[8] E. Horita and Y. Miyanaga, "Numerically stable RLS algorithms for time-varying signals,", Tech. Rep. IEICE, DSP97-32, June 1997. In Japanese.

[9] ——, "Numerically stable RLS algorithms for time-varying signals," *Scripta Technica, Electron. Commun. Jpn., Pt. 3*, vol. 82, no. 4, pp. 26–37, Apr. 1999. Translated from IEICE Trans. Fundamentals, vol. J81-A, no. 1, pp. 33–43, Jan. 1998..

[10] A. Kawamura, K. Fujii, Y. Itoh, and Y. Fukui, "A new noise reduction method using estimated noise spectrum," in *IEICE Trans. Fundamentals*, vol. E85-A, Apr. 2002, pp. 784–789.

[11] N. Sasaoka, A. Kawamura, Y. Itoh, and K. Fujii, "Noise reduction method based on noise reconstruction system with adaptive line enhancer," in *Proc. IEEE ISPACS*, Dec. 2003. in print.

[12] M. Gu and S. C. Eisenstat, "A divide-and-conquer algorithm for the symmetric tridiagonal eigenproblem," in *SIAM J. Matrix Anal. Appl.*, vol. 16, Jan. 1995, pp. 172–191.

[13] J. R. Bunch, C. P. Nielsen, and D. C. Sorensen, "Rank-one modification of the symmetric eigenproblem," *Numer. Math.*, vol. 31, pp. 31–48, 1978.

[14] J. J. Dongarra and D. C. Sorensen, "A fully parallel algorithm for the symmetric eigenvalue problem," in *SIAM J. Sci. Stat. Comput.*, vol. 8, Mar. 1987, pp. s.139–s.154.

[15] L. Greengard and V. Rokhlin, "A fast algorithm for particle simulations," *J. Comput. Phys.*, vol. 73, pp. 325–348, 1987.

[16] K. Mayyas and T. Aboulnasr, "Leaky LMS algorithm: MSE analysis for Gaussian data," *IEEE Trans. Signal Processsing*, vol. 45, pp. 927–934, Apr. 1997.