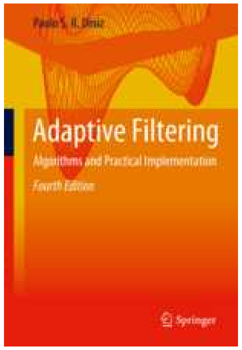
Conventional RLS Algorithm

Paulo S. R. Diniz - *diniz@lps.ufrj.br*



SIGNAL PROCESSING LABORATORY

COPPE/Poli - Federal University of Rio de Janeiro

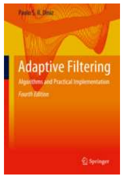
October 2012

slidecn54ed.tex

Conventional RLS Algorithm

Contents

The RLS Algorithm



The least squares objective function is

$$\begin{aligned}\xi^d(k) &= \sum_{i=0}^{k} \lambda^{k-i} \varepsilon^2(i) \\ &= \sum_{i=0}^{k} \lambda^{k-i} \left[d(i) - \mathbf{x}^T(i)\mathbf{w}(k) \right]^2 \qquad (1)\end{aligned}$$

λ is a weighting factor with $0 \ll \lambda < 1$. This parameter is called forgetting factor.

The RLS Algorithm

By differentiating $\xi^d(k)$ with respect to $\mathbf{w}(k)$

$$\frac{\partial \xi^d(k)}{\partial \mathbf{w}(k)} = -2 \sum_{i=0}^{k} \lambda^{k-i} \mathbf{x}(i)[d(i) - \mathbf{x}^T(i)\mathbf{w}(k)] \tag{2}$$

The RLS Algorithm

The optimal vector $\mathbf{w}(k)$ is

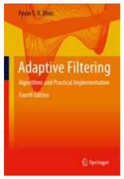
$$-\sum_{i=0}^{k} \lambda^{k-i}\mathbf{x}(i)\mathbf{x}^T(i)\mathbf{w}(k) + \sum_{i=0}^{k} \lambda^{k-i}\mathbf{x}(i)d(i) = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

The RLS Algorithm

$$\begin{aligned}\mathbf{w}(k) &= [\textstyle\sum_{i=0}^{k} \lambda^{k-i}\mathbf{x}(i)\mathbf{x}^T(i)]^{-1} \textstyle\sum_{i=0}^{k} \lambda^{k-i}\mathbf{x}(i)d(i) \\ &= \mathbf{R}_D^{-1}(k)\mathbf{p}_D(k)\end{aligned} \tag{3}$$

where $\mathbf{w}(k)$ in (3) is the optimal coefficient vector, and $\mathbf{R}_D(k)$ and $\mathbf{p}_D(k)$ are called deterministic correlation matrix and deterministic cross-correlation vector respectively.

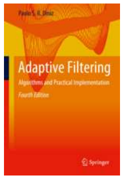
The RLS Algorithm



The computation of the inverse matrix is avoided through the use of the matrix inversion lemma, where

$$\mathbf{S}_D(k) = \mathbf{R}_D^{-1}(k) = \frac{1}{\lambda}[\mathbf{S}_D(k-1) - \frac{\mathbf{S}_D(k-1)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{S}_D(k-1)}{\lambda + \mathbf{x}^T(k)\mathbf{S}_D(k-1)\mathbf{x}(k)}] \tag{4}$$

The RLS Algorithm



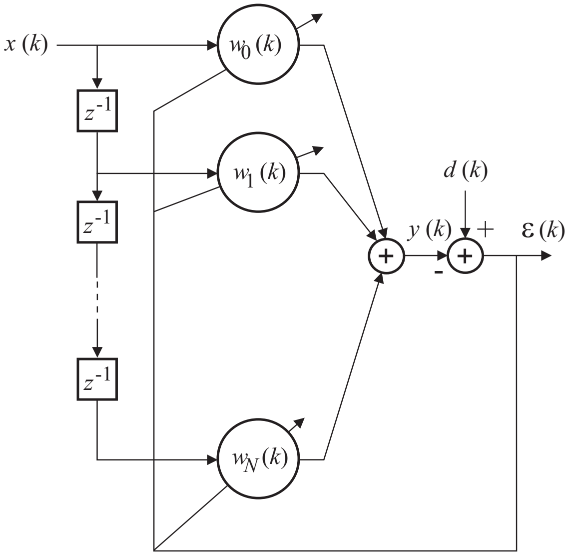


Figure 1: Adaptive FIR filter.

The RLS Algorithm

Algorithm 5.1
Conventional RLS Algorithm

Initialization

$$\mathbf{S}_D(-1) = \delta \mathbf{I}$$

where δ can be the inverse of the input-signal power estimate

$$\mathbf{p}_D(-1) = \mathbf{x}(-1) = [0\ 0\ \ldots\ 0]^T$$

The RLS Algorithm

Do for $k \geq 0$:

$$\mathbf{S}_D(k) = \frac{1}{\lambda}[\mathbf{S}_D(k-1) - \frac{\mathbf{S}_D(k-1)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{S}_D(k-1)}{\lambda+\mathbf{x}^T(k)\mathbf{S}_D(k-1)\mathbf{x}(k)}]$$

$$\mathbf{p}_D(k) = \lambda \mathbf{p}_D(k-1) + d(k)\mathbf{x}(k)$$

$$\mathbf{w}(k) = \mathbf{S}_D(k)\mathbf{p}_D(k)$$

If necessary compute

$$y(k) = \mathbf{w}^T(k)\mathbf{x}(k)$$

$$\varepsilon(k) = d(k) - y(k)$$

The RLS Algorithm

Alternative RLS algorithm is obtained if (3) is rewritten as

$$[\sum_{i=0}^{k} \lambda^{k-i}\mathbf{x}(i)\mathbf{x}^T(i)]\mathbf{w}(k) = \lambda[\sum_{i=0}^{k-1} \lambda^{k-i-1}\mathbf{x}(i)d(i)] + d(k)\mathbf{x}(k) \tag{5}$$

The RLS Algorithm

Considering that $\mathbf{R}_D(k-1)\mathbf{w}(k-1) = \mathbf{p}_D(k-1)$

$$
\begin{aligned}
&[\sum_{i=0}^{k} \lambda^{k-i}\mathbf{x}(i)\mathbf{x}^T(i)]\mathbf{w}(k) = \\
&[\sum_{i=0}^{k} \lambda^{k-i}\mathbf{x}(i)\mathbf{x}^T(i) - \mathbf{x}(k)\mathbf{x}^T(k)]\mathbf{w}(k-1) + \mathbf{x}(k)d(k)
\end{aligned}
\tag{6}
$$

The RLS Algorithm

Defining the *a priori* error as

$$e(k) = d(k) - \mathbf{x}^T(k)\mathbf{w}(k-1) \tag{7}$$

and substituting $d(k)$ in equation(6) then

$$\mathbf{w}(k) = \mathbf{w}(k-1) + e(k)\mathbf{S}_D(k)\mathbf{x}(k) \tag{8}$$

The RLS Algorithm

Algorithm 5.2
Alternative RLS Algorithm

Initialization

$$\mathbf{S}_D(-1) = \delta\mathbf{I}$$

where δ can be the inverse of an estimate of the input signal power

$$\mathbf{x}(-1) = \mathbf{w}(-1) = [0 \; 0 \; \ldots \; 0]^T$$

The RLS Algorithm

Do for $k \geq 0$

$$e(k) = d(k) - \mathbf{x}^T(k)\mathbf{w}(k-1) \quad \boldsymbol{\psi}(k) = \mathbf{S}_D(k-1)\mathbf{x}(k)$$

$$\mathbf{S}_D(k) = \frac{1}{\lambda}[\mathbf{S}_D(k-1) - \frac{\boldsymbol{\psi}(k)\boldsymbol{\psi}^T(k)}{\lambda+\boldsymbol{\psi}^T(k)\mathbf{x}(k)}]$$

$$\mathbf{w}(k) = \mathbf{w}(k-1) + e(k)\mathbf{S}_D(k)\mathbf{x}(k)$$

If necessary compute

$$y(k) = \mathbf{w}^T(k)\mathbf{x}(k) \quad \varepsilon(k) = d(k) - y(k)$$

The RLS Algorithm

No. of divisions is reduced using $\boldsymbol{\phi}(k) = \frac{\psi^T(k)}{\lambda + \psi^T(k)\mathbf{X}(k)}$ that can be used as follows

$$\mathbf{S}_D(k) = \frac{1}{\lambda}[\mathbf{S}_D(k-1) - \psi(k)\boldsymbol{\phi}^T(k)] \tag{9}$$

Properties Of The LS Solutions

Orthogonality Principle

Define the matrices $\mathbf{X}(k)$ and $\mathbf{d}(k)$ that contains all the information

$$\mathbf{X}(k) =$$

$$\begin{bmatrix} x(k) & \lambda^{1/2}x(k-1) & \cdots & \lambda^{k/2}x(0) \\ x(k-1) & \lambda^{1/2}x(k-2) & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ x(k-N) & \lambda^{1/2}x(k-N-1) & \cdots & 0 \end{bmatrix}$$

$$= [\mathbf{x}(k)\ \lambda^{1/2}\mathbf{x}(k-1)\ \ldots\ \lambda^{k/2}\mathbf{x}(0)] \tag{10}$$

$$\mathbf{d}(k) = [d(k)\ \lambda^{1/2}d(k-1)\ \ldots\ \lambda^{k/2}d(0)]^T \tag{11}$$

where $\mathbf{X}(k)$ is $(N+1) \times (k+1)$ and $\mathbf{d}(k)$ is $(k+1) \times (1)$.

The RLS Algorithm

By using the matrices defined above

$$\mathbf{X}(k)\mathbf{X}^T(k)\mathbf{w}(k) = \mathbf{X}(k)\mathbf{d}(k) \tag{12}$$

Hence defining

$$\begin{aligned} \mathbf{y}(k) &= \mathbf{X}^T(k)\mathbf{w}(k) \\ &= [y(k)\ \lambda^{1/2}y(k-1)\ \cdots\ \lambda^{k/2}y(0)]^T \end{aligned} \tag{13}$$

it follows from (12) that

$$\begin{aligned} &\mathbf{X}(k)\mathbf{X}^T(k)\mathbf{w}(k) - \mathbf{X}(k)\mathbf{d}(k) \\ &= \mathbf{X}(k)[\mathbf{y}(k) - \mathbf{d}(k)] = \mathbf{0} \end{aligned} \tag{14}$$

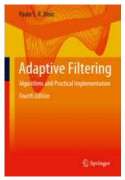
The RLS Algorithm

This relation means that the weighted error vector given by

$$\boldsymbol{\varepsilon}(k) = \begin{bmatrix} \varepsilon(k) \\ \lambda^{1/2}\varepsilon(k-1) \\ \vdots \\ \lambda^{k/2}\varepsilon(0) \end{bmatrix} = \mathbf{d}(k) - \mathbf{y}(k) \tag{15}$$

is in the null space of $\mathbf{X}(k)$.

The RLS Algorithm



Example 1

Suppose $\lambda = 1$ and that

$$\mathbf{d}(1) = \begin{bmatrix} 0.5 \\ 1.5 \end{bmatrix} \quad \mathbf{X}(1) = [1 \quad -2] \quad \mathbf{w}(1) = -\frac{2.5}{5}$$

The RLS Algorithm

The output of the adaptive filter is

$$\mathbf{y}(1) = \begin{bmatrix} -\frac{2.5}{5} \\ 1 \end{bmatrix}$$

Note that

$$[1 \quad -2][\mathbf{y}(1) - \mathbf{d}(1)] = [1 \quad -2] \begin{bmatrix} -1 \\ -0.5 \end{bmatrix} = 0$$

The RLS Algorithm

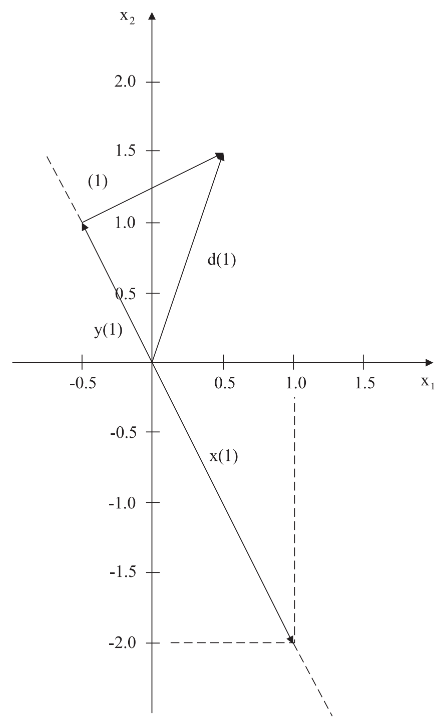


Figure 2: Geometric interpretation of least-square solution.

Conventional RLS Algorithm

Relation Between LS and Wiener Solution

When $\lambda = 1$ the matrix $\frac{1}{k+1}\mathbf{R}_D(k)$ for large k is a consistent estimate of $\mathbf{R}$, if the process from which the input signal was taken is ergodic. The same is valid for $\frac{1}{k+1}\mathbf{p}_D(k)$ related to $\mathbf{p}$ if the desired signal is ergodic.

Conventional RLS Algorithm

In this case

$$\mathbf{R} = \lim_{k\to\infty} \frac{1}{k+1} \sum_{i=0}^{k} \mathbf{x}(i)\mathbf{x}^T(i) = \lim_{k\to\infty} \frac{1}{k+1} \mathbf{R}_D(k) \qquad (16)$$

and

$$\mathbf{p} = \lim_{k\to\infty} \frac{1}{k+1} \sum_{i=0}^{k} \mathbf{x}(i)d(i) = \lim_{k\to\infty} \frac{1}{k+1} \mathbf{p}_D(k) \qquad (17)$$

It can then be shown that

$$\mathbf{w}(k) = \mathbf{R}_D^{-1}(k)\mathbf{p}_D(k) = \mathbf{R}^{-1}\mathbf{p} = \mathbf{w}_o \qquad (18)$$

for large k.

Conventional RLS Algorithm

Influence of Initialization

The initialization of $\mathbf{S}_D(-1) = \delta \mathbf{I}$ causes a bias

$$
\begin{aligned}
\sum_{i=-1}^{k} \lambda^{k-i} \mathbf{x}(i) \mathbf{x}^T(i) \mathbf{w}(k) &= [\sum_{i=0}^{k} \lambda^{k-i} \mathbf{x}(i) \mathbf{x}^T(i) + \frac{\lambda^{k+1}}{\delta} \mathbf{I}] \mathbf{w}(k) \\
&= \mathbf{p}_D(k) \qquad (19)
\end{aligned}
$$

Conventional RLS Algorithm

By multiplying $\mathbf{R}_D^{-1}(k)$ on both sides for $k \rightarrow \infty$

$$\mathbf{w}(k) + \frac{\lambda^{k+1}}{\delta}\mathbf{R}_D^{-1}(k)\mathbf{w}(k) = \mathbf{w}_o \tag{20}$$

The bias is

$$\mathbf{w}(k) - \mathbf{w}_o \approx -\frac{\lambda^{k+1}}{\delta}\mathbf{S}_D(k)\mathbf{w}_o \tag{21}$$

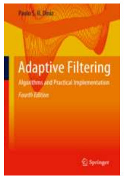
For $\lambda < 1$ the bias tends to zero. For $\lambda = 1$, $\mathbf{S}_D(k)$ approaches a null matrix for large k.

Conventional RLS Algorithm

Behavior of the Coefficient Vector

The input vectors $\mathbf{x}(k)$, for $k = 0, 1, \ldots$, are considered known, while the coefficients $w_i(k)$ for $i = 0, 1, \ldots N$, are considered as stochastic processes. In this case for $k \geq N$.

Conventional RLS Algorithm



$$
\begin{aligned}
E[\mathbf{w}(k)] &= E\left\{[\mathbf{X}(k)\mathbf{X}^T(k)]^{-1}\mathbf{X}(k)d(k)\right\} \\
&= E\left[[\mathbf{X}(k)\mathbf{X}^T(k)]^{-1}[\mathbf{X}(k)\mathbf{X}^T(k)\mathbf{w}_o + \mathbf{n}(k)]\right] \\
&= E[[\mathbf{X}(k)\mathbf{X}^T(k)]^{-1}\mathbf{X}(k)\mathbf{X}^T(k)\mathbf{w}_o] = \mathbf{w}_o
\end{aligned}
\tag{22}
$$

The equation shows that the estimate is unbiased when $\lambda \leq 1$.

Conventional RLS Algorithm

Transient Behavior

Define

$$\Delta \mathbf{w}(k) = \mathbf{w}(k) - \mathbf{w}_o \tag{23}$$

Rewriting eq. (6) in matrix form it follows

$$\mathbf{R}_D(k)\mathbf{w}(k) = \lambda \mathbf{R}_D(k-1)\mathbf{w}(k-1) + \mathbf{x}(k)d(k) \tag{24}$$

Conventional RLS Algorithm

Defining the minimum output error as

$$e_o(k) = d(k) - \mathbf{w}_o^T \mathbf{x}(k) \tag{25}$$

and replacing $d(k)$ in eq. (24)

$$\mathbf{R}_D(k)\Delta\mathbf{w}(k) = \lambda \mathbf{R}_D(k-1)\Delta\mathbf{w}(k-1) + \mathbf{x}(k)e_o(k) \tag{26}$$

Conventional RLS Algorithm

where it was used the following relation

$$\mathbf{R}_D(k) = \lambda \mathbf{R}_D(k-1) + \mathbf{x}(k)\mathbf{x}^T(k) \tag{27}$$

The solution of eq. (26) is given by

$$\Delta\mathbf{w}(k) \;=\; \lambda^{k+1}\mathbf{R}_D^{-1}(k)\mathbf{R}_D(-1)\Delta\mathbf{w}(-1) + \mathbf{R}_D^{-1}(k)\sum_{i=0}^{k}\lambda^{k-i}\mathbf{x}(i)e_o(i) \tag{28}$$

Conventional RLS Algorithm

By replacing $\mathbf{R}_D(-1)$ by $\frac{1}{\delta}\,\mathbf{I}$ and taking the expected value

$$E[\Delta\mathbf{w}(k)] \;=\; \frac{\lambda^{k+1}}{\delta}E[\mathbf{S}_D(k)]\Delta\mathbf{w}(-1) + E[\mathbf{S}_D(k)\sum_{i=0}^{k}\lambda^{k-i}\mathbf{x}(i)e_o(i)] \tag{29}$$

Conventional RLS Algorithm

Since $\mathbf{S}_D(k)$ is dependent on all past input vectors it can be considered independent of an individual $\mathbf{x}(i)$. Also, due to the orthogonality principle $e_o(i)$ can be considered uncorrelated to all elements of $\mathbf{x}(i)$. The last vector of eq. (29) cannot have large element values. The first vector of equation (29) has large element values only during the initial convergence, since as $k \rightarrow \infty, \lambda^{k+1} \rightarrow 0$ and $\mathbf{S}_D(k)$ has a nonincreasing behavior ,i.e. $\mathbf{R}_D(k)$ is assumed to remain positive definite as $k \rightarrow \infty$.

Conventional RLS Algorithm

We conclude that the coefficients tend to $\mathbf{w}_o$ almost independently

from any eigenvalue spread of $\mathbf{R}$. In the simple case of $\lambda = 1$, consider the worst case which we replace $E[\mathbf{S}_D(k)]$ in (29) by

$$\mathbf{S}_{D_{\max}} = \frac{\mathbf{q}_{\min}\mathbf{q}_{\min}^T}{\lambda_{\min}(k+1)} \tag{30}$$

where $\mathbf{q}_{\min}$ is the eigenvector of $\lambda_{\min}$. The value of the minimum eigenvalue affects the convergence only in the first few iterations.

Conventional RLS Algorithm

Coefficient-Error-Vector Cov. Matrix

The coefficient-error-vector covariance matrix is given by

$$\begin{aligned}\mathrm{cov}\,[\Delta\mathbf{w}(k)] &= E[(\mathbf{w}(k)-\mathbf{w}_o)(\mathbf{w}(k)-\mathbf{w}_o)^T] \\ &= \sigma_n^2 E[\mathbf{S}_D(k)] \qquad (31)\end{aligned}$$

for $\lambda = 1$.

Conventional RLS Algorithm

Proof

$$
\begin{aligned}
\mathbf{w}(k) - \mathbf{w}_o &= \mathbf{S}_D(k)\mathbf{p}_D(k) - \mathbf{S}_D(k)\mathbf{S}_D^{-1}(k)\mathbf{w}_o \\
&= (\mathbf{X}(k)\mathbf{X}^T(k))^{-1}\mathbf{X}(k)(\mathbf{d}(k) - \mathbf{X}^T(k)\mathbf{w}_o) \\
&= (\mathbf{X}(k)\mathbf{X}^T(k))^{-1}\mathbf{X}(k)\mathbf{n}(k)
\end{aligned}
$$

where $\mathbf{n}(k) = [n(k)\ \lambda^{1/2}n(k-1)\ \lambda n(k-2)\ \ldots\ \lambda^{k/2}n(0)]^T$.

Conventional RLS Algorithm

It then follows that

$$
\begin{aligned}
\operatorname{cov}[\Delta \mathbf{w}(k)] &= E[(\mathbf{X}(k)\mathbf{X}^T(k))^{-1}\mathbf{X}(k)E[\mathbf{n}(k)\mathbf{n}^T(k)]\mathbf{X}^T(k)(\mathbf{X}(k)\mathbf{X}^T(k))^{-1}] \\
&= \sigma_n^2 E[\mathbf{S}_D(k)\mathbf{X}(k)\Lambda\mathbf{X}^T(k)\mathbf{S}_D(k)]
\end{aligned}
$$

where $\mathbf{X}(k)$ was considered known and

$$
\Lambda = \begin{bmatrix} 1 & & & & \\ & \lambda & & 0 & \\ & & \lambda^2 & & \\ & 0 & & \ddots & \\ & & & & \lambda^k \end{bmatrix}
$$

Conventional RLS Algorithm

where for $\lambda = 1$,

$$\begin{aligned}
\operatorname{cov}[\Delta\mathbf{w}(k)] &= \sigma_n^2 \mathbf{S}_D(k) E[\mathbf{X}(k)\mathbf{X}^T(k)\mathbf{S}_D(k)] \\
&= \sigma_n^2 E[\mathbf{S}_D(k)\mathbf{R}_D(k)\mathbf{S}_D(k)] \\
&= \sigma_n^2 E[\mathbf{S}_D(k)]
\end{aligned}$$

$\diamond$

Conventional RLS Algorithm

Behavior of the Error Signal

With measurement noise, the error signal is given by

$$e(k) = d'(k) - \mathbf{w}^T(k-1)\mathbf{x}(k) + n(k) \tag{32}$$

then

$$\begin{aligned} E[e(k)] &= E[d'(k)] - E[\mathbf{w}^T(k-1)]\mathbf{x}(k) + E[n(k)] \\ &= E[d'(k)] - \mathbf{w}_o^T\mathbf{x}(k) + E[n(k)] \\ &= E[n(k)] \end{aligned} \tag{33}$$

assuming the filter order is suf. to model the desired signal.

Conventional RLS Algorithm

If the noise signal has zero mean then

$$E[e(k)] = 0$$

$\boxed{\textbf{The Minimum Mean Square Error}}$

The minimum MSE in the presence of external uncorrelated noise is given by

$$\xi_{\min} = E[e^2(k)] = E[n^2(k)] = \sigma_n^2 \qquad (34)$$

If the additive noise is correlated to the input and the desired signal, a more complicated expression for the MSE results.

Conventional RLS Algorithm

When employing the *a posteriori* error the minimum MSE, denoted by $\xi_{\mathrm{min,p}}$, differs from the corresponding value related to the *a priori* error. The following relation can be verified

$$\Delta\mathbf{w}(k) = \mathbf{S}_D(k)\mathbf{X}(k)\mathbf{n}(k) \tag{35}$$

When a measurement noise is present, the *a posteriori* error signal is given by

$$\varepsilon(k) = d'(k) - \mathbf{w}^T(k)\mathbf{x}(k) + n(k) = -\Delta\mathbf{w}^T(k)\mathbf{x}(k) + e_o(k) \tag{36}$$

Conventional RLS Algorithm

The expression for the MSE related to the *a posteriori* error is then given by

$$\begin{aligned}\xi(k) &= E[\varepsilon^2(k)] \\ &= E[e_o^2(k)] - 2E[\mathbf{x}^T(k)\Delta\mathbf{w}(k)e_o(k)] \\ &\quad + E[\Delta\mathbf{w}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\Delta\mathbf{w}(k)] \qquad (37)\end{aligned}$$

Conventional RLS Algorithm

By replacing the expression (35) in equation (37) above

$$
\begin{aligned}
\xi(k) &= E[e_o^2(k)] - 2E[\mathbf{x}^T(k)\mathbf{S}_D(k)\mathbf{X}(k)\mathbf{n}(k)e_o(k)] \\
&\quad + E[\Delta\mathbf{w}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\Delta\mathbf{w}(k)] \\
&= \sigma_n^2 - 2E[\mathbf{x}^T(k)\mathbf{S}_D(k)\mathbf{X}(k)] \begin{bmatrix} \sigma_n^2 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \\
&\quad + E[\Delta\mathbf{w}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\Delta\mathbf{w}(k)]
\end{aligned}
$$

Conventional RLS Algorithm

$$
\begin{aligned}
&= \sigma_n^2 - 2E[\mathbf{x}^T(k)\mathbf{S}_D(k)\mathbf{x}(k)]\sigma_n^2 + E[\Delta\mathbf{w}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\Delta\mathbf{w}(k)] \\
&= \xi_{\mathrm{min,p}} + E[\Delta\mathbf{w}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\Delta\mathbf{w}(k)]
\end{aligned}
\tag{38}
$$

where in the second equality the additional noise is uncorrelated with the input signal and that $e_o(k) = n(k)$. This equality occurs when the adaptive filter has sufficient order to identify the unknown system.

$\xi_{\mathrm{min,p}}$ related to the *a posteriori* error in eq. (38) is not the same as minimum MSE of the *a priori* error ξ_{min}. The term $E[\Delta\mathbf{w}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\Delta\mathbf{w}(k)]$ in eq. (38) determines the excess MSE of the RLS algorithm.

Conventional RLS Algorithm

It is possible to verify that the following expressions for $\xi_{\mathrm{min,p}}$ are accurate approximations

$$
\begin{aligned}
\xi_{\mathrm{min,p}} &= \{1 - 2E[\mathbf{x}^T(k)\mathbf{S}_D(k)\mathbf{x}(k)]\}\sigma_n^2 \\
&= \left\{1 - 2\mathrm{tr}\left[E\left(\mathbf{S}_D(k)\mathbf{x}(k)\mathbf{x}^T(k)\right)\right]\right\}\sigma_n^2 \\
&= \left\{1 - 2\mathrm{tr}\left[\frac{1-\lambda}{1-\lambda^{k+1}}\mathbf{I}\right]\right\}\sigma_n^2 \\
&= \left\{1 - 2(N+1)\left[\frac{1-\lambda}{1-\lambda^{k+1}}\right]\right\}\sigma_n^2 \\
&= \left\{1 - 2(N+1)\left[\frac{1}{1+\lambda+\lambda^2+\cdots+\lambda^k}\right]\right\}\sigma_n^2 \qquad (39)
\end{aligned}
$$

Conventional RLS Algorithm

Example 5.2

Repeat the equalization problem of Example 3.1 of Chapter 3 using the RLS algorithm.

(a) Using $\lambda = 0.99$, run the algorithm and save the matrix $\mathbf{S}_D(k)$ at the iteration 500 and compare with the inverse of the input signal correlation matrix.

(b) Plot the convergence path for the RLS algorithm on the MSE surface.

Conventional RLS Algorithm

Solution:

(a) The inverse of matrix $\mathbf{R}$, as computed in the Example 3.1, is given by

$$\mathbf{R}^{-1} = 0.45106 \begin{bmatrix} 1.6873 & 0.7937 \\ 0.7937 & 1.6873 \end{bmatrix}$$
$$= \begin{bmatrix} 0.7611 & 0.3580 \\ 0.3580 & 0.7611 \end{bmatrix}$$

Conventional RLS Algorithm

The initialization matrix $\mathbf{S}_D(-1)$ was a diagonal matrix with the diagonal elements equal to 0.1. The matrix $\mathbf{S}_D(k)$ at the $500th$ iteration, obtained by averaging the results of 30 experiments, was

$$\mathbf{S}_D(500) \quad = \quad \begin{bmatrix} 0.0078 & 0.0037 \\ 0.0037 & 0.0078 \end{bmatrix}$$

Also, the obtained values of the deterministic cross-correlation vector was

$$\mathbf{p}_D(500) \quad = \quad \begin{bmatrix} 95.05 \\ 46.21 \end{bmatrix}$$

Conventional RLS Algorithm

Now if we divide each element of the matrix $\mathbf{R}^{-1}$ by

$$\frac{1-\lambda^{k+1}}{1-\lambda} = 99.34$$

The resulting matrix is

$$\frac{1}{99.34}\mathbf{R}^{-1} \quad = \quad \begin{bmatrix} 0.0077 & 0.0036 \\ 0.0036 & 0.0077 \end{bmatrix}$$

as can be noted the values of the elements of the above matrix are close to the average values of the corresponding elements of matrix $\mathbf{S}_D(500)$.

Conventional RLS Algorithm

Similarly, if we multiply the cross-correlation vector $\mathbf{p}$ by 99.34, the result is

$$99.34\mathbf{p} = \begin{bmatrix} 94.61 \\ 47.31 \end{bmatrix}$$

The values of the elements of this vector are also close to the corresponding elements of $\mathbf{p}_D(500)$.

Conventional RLS Algorithm

(b) The convergence path of the RLS algorithm on the MSE surface is depicted in Fig. 5.3. The reader should notice that the RLS algorithm approaches the minimum using large steps when the coefficients of the adaptive filter are far away from the optimum solution.

Conventional RLS Algorithm

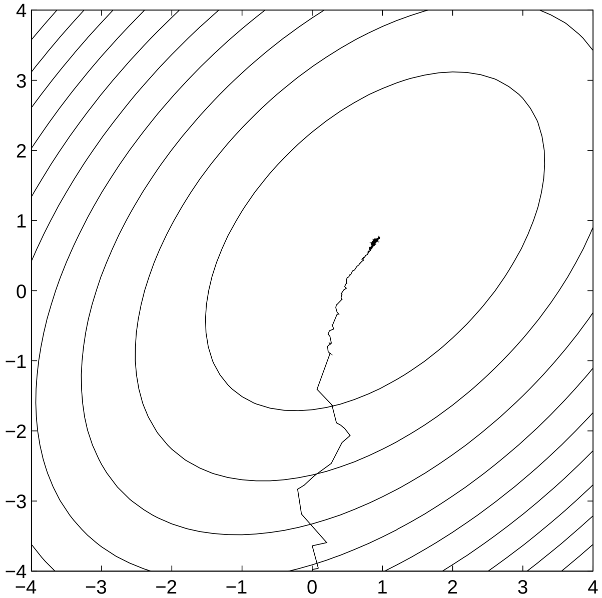


Figure 3: Convergence path of the RLS adaptive filter.

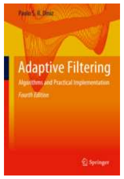
Conventional RLS Algorithm

Excess MSE and Misadjustment

In the real implementation there is always an excess of MSE at the output caused by $\Delta\mathbf{w}(k) = \mathbf{w}(k) - \mathbf{w}_o$. The MSE for a given coefficient vector is

$$\begin{aligned} \xi(k) &= \xi_{\min,p} + E[(\mathbf{w}(k) - \mathbf{w}_o)^T \mathbf{R} (\mathbf{w}(k) - \mathbf{w}_o)] \\ &= \xi_{min,p} + E[\Delta\mathbf{w}^T(k) \mathbf{R} \Delta\mathbf{w}(k)] \end{aligned} \tag{40}$$

Conventional RLS Algorithm



Now considering that $\Delta w_j(k)$ for $j = 0 \ldots N$ are random variables with zero mean and independent of $\mathbf{x}(k)$, then by employing eqs. (31) and (17), the ensemble average of the MSE can be calculated as

$$\begin{aligned} E[\xi(k)] &= \xi_{min,p} + E[\Delta \mathbf{w}^T(k) \mathbf{R} \Delta \mathbf{w}(k)] \\ &= \xi_{min,p} + E[\mathrm{tr}\,(\mathbf{R} \Delta \mathbf{w}(k) \Delta \mathbf{w}^T(k))] \end{aligned}$$

Conventional RLS Algorithm

$$
\begin{aligned}
&= \xi_{min,p} + \mathrm{tr}\left(\mathbf{R}E[\Delta\mathbf{w}(k)\Delta\mathbf{w}^T(k)]\right) \\
&= \xi_{min,p} + \sigma_n^2 \mathrm{tr}\left(\mathbf{R}\mathbf{S}_D(k)\right) \\
&= \sigma_n^2(1 + \mathrm{tr}\,(\mathbf{R}\frac{\mathbf{R}^{-1}}{k+1}))\ for\ \ k \rightarrow \infty \\
&= \sigma_n^2(1 + \frac{N+1}{k+1})
\end{aligned}
$$

where $\lambda = 1$ and $\xi_{min,p} = \xi_{min} = \sigma_n^2$. The minimum MSE is reached after the algorithm has operated a number of samples larger than the filter order.

Conventional RLS Algorithm

The expected excess in the MSE is defined by

$$\Delta\xi(k) = E[\Delta\mathbf{w}^T(k)\mathbf{R}\Delta\mathbf{w}(k)] \tag{41}$$

For $\lambda < 1$. From equation (26) one can show that

$$\Delta\mathbf{w}(k) = \lambda\mathbf{R}_D^{-1}(k)\mathbf{R}_D(k-1)\Delta\mathbf{w}(k-1) + \mathbf{R}_D^{-1}(k)\mathbf{x}(k)e_o(k) \tag{42}$$

Conventional RLS Algorithm

By applying equation (42) to (41) it follows that

$$E[\Delta \mathbf{w}^T(k)\mathbf{R}\Delta \mathbf{w}(k)] = \rho_1 + \rho_2 + \rho_3 + \rho_4 \tag{43}$$

where

$$\rho_1 = \lambda^2 E[\Delta \mathbf{w}^T(k-1)\mathbf{R}\Delta \mathbf{w}(k-1)] \tag{44}$$

Conventional RLS Algorithm

$$
\begin{aligned}
\rho_2 &\approx \lambda(1-\lambda)E[\Delta\mathbf{w}^T(k-1)]E[\mathbf{x}(k)e_o(k)] \\
&= 0
\end{aligned} \tag{45}
$$

Following a similar approach it can be shown that $\rho_3 = 0$.

Conventional RLS Algorithm

3- Evaluation of ρ_4

$$\begin{aligned}\rho_4 &= E[\mathbf{x}^T(k)\mathbf{R}_D^{-1}(k)\mathbf{R}\mathbf{R}_D^{-1}(k)\mathbf{R}\mathbf{R}^{-1}\mathbf{x}(k)e_o^2(k)] \\ &\approx (1-\lambda)^2 E[\mathbf{x}^T(k)(\mathbf{I}-\mathbf{\Delta I}(k))^2\mathbf{R}^{-1}\mathbf{x}(k)]\xi_{\min}\end{aligned} \tag{46}$$

Eq. (46) can be simplified to

$$\rho_4 = (1-\lambda)^2 \mathrm{tr}\left\{\mathbf{I} + E[\mathbf{\Delta I}^2(k)]\right\}\xi_{\min} \tag{47}$$

where tr[$\cdot$] means trace of [$\cdot$].

Conventional RLS Algorithm

By using (44), (45) and (47), it follows that

$$\begin{aligned} E[\Delta\mathbf{w}^T(k)\mathbf{R}\Delta\mathbf{w}(k)] &= \lambda^2 E[\Delta\mathbf{w}^T(k-1)\mathbf{R}\Delta\mathbf{w}(k-1)] \\ &\quad +(1-\lambda)^2 \mathrm{tr}\left\{\mathbf{I} + E[\Delta\mathbf{I}^2(k)]\right\}\xi_{\min} \qquad (48) \end{aligned}$$

Asymptotically, the solution of the equation above is

$$\xi_{\mathrm{exc}} = \frac{1-\lambda}{1+\lambda}\mathrm{tr}\left\{\mathbf{I} + E[\Delta\mathbf{I}^2(k)]\right\}\xi_{\min} \qquad (49)$$

The term $E[\Delta\mathbf{I}^2(k)]$ is dependent on fourth-order statistics of the input signal. Assume diagonal dominance of $\mathbf{R}$ and $\Delta\mathbf{R}$, and use the definition of $\Delta\mathbf{I}(k)$

Conventional RLS Algorithm

$$E[\Delta \mathbf{I}_{ii}^2(k)] = (1-\lambda)\frac{E[\Delta r_{ii}^2(k)]}{[\sigma_x^2]^2} \tag{50}$$

where σ_x^2 is variance of $x(k)$.

$$\Delta r_{ii}(k) = \lambda \Delta r_{ii}(k-1) + x(k-i)x(k-i) - r_{ii} \tag{51}$$

and using the independence between $\Delta r_{ii}(k)$ and $x(k)$ it follows asymptotically

$$E[\Delta r_{ii}^2(k)] = \frac{1}{1-\lambda^2}\sigma_{x^2(k-i)}^2 = \frac{1}{1-\lambda^2}\sigma_{x^2}^2 \tag{52}$$

Conventional RLS Algorithm

By substituting eq. (52) in (50)

$$E[\Delta \mathbf{I}_{ii}^2] = \frac{1-\lambda}{1+\lambda} \frac{\sigma_{x^2}^2}{\sigma_x^2} = \frac{1-\lambda}{1+\lambda} \mathcal{K} \qquad (53)$$

where $\mathcal{K}$ is dependent on input signal statistics. For Gaussian signals $\mathcal{K} = 2$.

Returning to our main objective, the excess of MSE is

$$\xi_{\text{exc}} = (N+1) \frac{1-\lambda}{1+\lambda} (1 + \frac{1-\lambda}{1+\lambda} \mathcal{K}) \xi_{\text{min}} \qquad (54)$$

Conventional RLS Algorithm

If $\lambda \approx 1$ and $\mathcal{K}$ is not very large then

$$\xi_{\text{exc}} = (N+1)\frac{1-\lambda}{1+\lambda}\xi_{\text{min}} \tag{55}$$

The misadjustment can be deduced from (54)

$$M = (N+1)\frac{1-\lambda}{1+\lambda}(1+\frac{1-\lambda}{1+\lambda}\mathcal{K}) \tag{56}$$

Conventional RLS Algorithm

Behavior in Nonstat. Environ.

If the input signal and/or the desired signal are nonstationary, the optimal coefficients are time variant $\mathbf{w}_o(k)$. That autocorrelation matrix $\mathbf{R}(k)$ and/or the crosscorrelation vector $\mathbf{p}(k)$ are time variant.

Conventional RLS Algorithm

Recall that

$$\mathbf{w}(k) = \mathbf{w}(k-1) + \mathbf{S}_D(k)\mathbf{x}(k)(d(k) - \mathbf{x}^T(k)\mathbf{w}(k-1)) \qquad (57)$$

and

$$d(k) = \mathbf{x}^T(k)\mathbf{w}_o(k-1) + e'_o(k) \qquad (58)$$

Conventional RLS Algorithm

one can replace eq. (58) in (57) then

$$\begin{aligned}\mathbf{w}(k) &= \mathbf{w}(k-1) + \mathbf{S}_D(k)\mathbf{x}(k)\mathbf{x}^T(k)[\mathbf{w}_o(k-1) - \mathbf{w}(k-1)] \\ &\quad + \mathbf{S}_D(k)\mathbf{x}(k)e'_o(k) \qquad (59)\end{aligned}$$

Consider that $\mathbf{x}(k)$ and $e'_o(k)$ are approximately orthogonal and $\mathbf{w}(k-1)$ is independent of $\mathbf{x}(k)$, then

$$\begin{aligned}E[\mathbf{w}(k)] &= E[\mathbf{w}(k-1)] + E[\mathbf{S}_D(k)\mathbf{x}(k)\mathbf{x}^T(k)] \\ &\quad (\mathbf{w}_o(k-1) - E[\mathbf{w}(k-1)]) \qquad (60)\end{aligned}$$

Conventional RLS Algorithm

It is needed to compute $E[\mathbf{S}_D(k)\mathbf{x}(k)\mathbf{x}^T(k)]$, for nonstationary environment. In this case

$$\mathbf{R}_D(k) = \sum_{l=0}^{k} \lambda^{k-l}\mathbf{R}(l) + \Delta\mathbf{R}(k) \tag{61}$$

since $E[\mathbf{R}_D(k)] = \sum_{l=0}^{k}\lambda^{k-l}\mathbf{R}(l)$.

If the environment is varying in a slower pace than the memory of the adaptive RLS algorithm then

$$\mathbf{R}_D(k) \approx \frac{1}{1-\lambda}\mathbf{R}(k) + \Delta\mathbf{R}(k) \tag{62}$$

Conventional RLS Algorithm

Considering that $(1-\lambda)||\mathbf{R}^{-1}(k)\Delta\mathbf{R}(k)|| < 1$ then

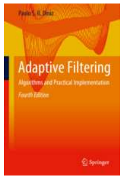
$$\mathbf{S}_D(k) \approx (1-\lambda)\mathbf{R}^{-1}(k) - (1-\lambda)^2\mathbf{R}^{-1}(k)\Delta\mathbf{R}(k)\mathbf{R}^{-1}(k) \tag{63}$$

it then follows that

$$\begin{aligned} E[\mathbf{w}(k)] &= E[\mathbf{w}(k-1)] + \{(1-\lambda)E[\mathbf{R}^{-1}(k)\mathbf{x}(k)\mathbf{x}^T(k)] \\ &+(1-\lambda)^2E[\mathbf{R}^{-1}(k)\Delta\mathbf{R}(k)\mathbf{R}^{-1}(k)\mathbf{x}(k)\mathbf{x}^T(k)]\} \\ &(\mathbf{w}_o(k-1) - E[\mathbf{w}(k-1)]) \\ &\approx E[\mathbf{w}(k-1)] + (1-\lambda)(\mathbf{w}_o(k-1) - E[\mathbf{w}(k-1)]) \end{aligned} \tag{64}$$

where $\Delta\mathbf{R}(k)$ is independent of $\mathbf{x}(k)$ and has zero expected value.

Conventional RLS Algorithm



Now defining the lag error vector in the coefficients as

$$\mathbf{l}_{\mathbf{w}}(k) = E[\mathbf{w}(k)] - \mathbf{w}_o(k) \tag{65}$$

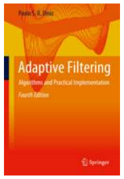
from equation (64) it can be concluded that

$$\mathbf{l}_{\mathbf{w}}(k) = \lambda \mathbf{l}_{\mathbf{w}}(k-1) - \mathbf{w}_o(k) + \mathbf{w}_o(k-1) \tag{66}$$

This eq. is equivalent to pass the optimal value $\mathbf{w}_o(k)$ through a first order filter

$$L_i(z) = -\frac{z-1}{z-\lambda} W_{oi}(z) \tag{67}$$

Conventional RLS Algorithm



The filter transient response converges with a time constant

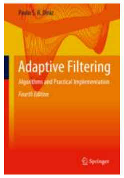
$$\tau = \frac{1}{1-\lambda} \tag{68}$$

To calculate the MSE suppose that the coefficients values are a first order Markov process

$$\mathbf{w}_o(k) = \lambda_{\mathbf{w}} \mathbf{w}_o(k-1) + \mathbf{n}_{\mathbf{w}}(k) \tag{69}$$

where the elements of $\mathbf{n}_{\mathbf{w}}(k)$ are zero mean Gaussian noise processes with variance $\sigma^2_{\mathbf{w}}$, and $\lambda_{\mathbf{w}} < 1$.

Conventional RLS Algorithm

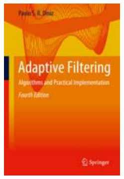


$\lambda < \lambda_{\mathbf{w}} < 1$, since the optimal coefficients must vary slower than the filter tracking speed, i.e. $\frac{1}{1-\lambda} << \frac{1}{1-\lambda_{\mathbf{w}}}$.

The excess of MSE due to lag is

$$\begin{aligned} \xi_{\text{lag}} &= E[\mathbf{l}_{\mathbf{w}}^T(k)\mathbf{R}\mathbf{l}_{\mathbf{w}}(k)] \\ &= E[\text{tr}\left(\mathbf{R}\mathbf{l}_{\mathbf{w}}(k)\mathbf{l}_{\mathbf{w}}^T(k)\right)] \\ &= \text{tr}\left(\mathbf{R}E[\mathbf{l}_{\mathbf{w}}(k)\mathbf{l}_{\mathbf{w}}^T(k)]\right) \end{aligned} \tag{70}$$

Conventional RLS Algorithm



If $\lambda_{\mathbf{w}} \approx 1$ then from (67) and (69)

$$\begin{aligned} \xi_{\mathrm{lag}} &\approx \operatorname{tr}\left[\mathbf{R} \cdot \frac{\sigma_{\mathbf{w}}^2}{1-\lambda^2}\mathbf{I}\right] = \frac{\sigma_{\mathbf{w}}^2}{1-\lambda^2}\operatorname{tr}\mathbf{R} \\ &= \frac{(N+1)\sigma_{\mathbf{w}}^2\sigma_x^2}{1-\lambda^2} \end{aligned} \tag{71}$$

Conventional RLS Algorithm

For general $\lambda_{\mathbf{w}}$ it can be shown that

$$\xi_{\text{lag}} \approx \frac{(N+1)\sigma_{\mathbf{w}}^2\sigma_x^2}{\lambda_{\mathbf{w}}(1+\lambda^2)-\lambda(1+\lambda_{\mathbf{w}}^2)}\left(\frac{1-\lambda}{1+\lambda}-\frac{1-\lambda_{\mathbf{w}}}{1+\lambda_{\mathbf{w}}}\right) \tag{72}$$

If $\lambda = 1$ and $\lambda_{\mathbf{w}} \approx 1$, the MSE due to lag tends to infinity indicating that the RLS cannot track any change in the environment.

Conventional RLS Algorithm

Since $\lambda \approx 1$ we can rewrite (72) as

$$\xi_{\text{lag}} \approx (N+1)\frac{\sigma_{\mathbf{w}}^2}{2(1-\lambda)}\sigma_x^2 \tag{73}$$

The total excess MSE accounting for the lag and finite memory is

$$\xi_{\text{total}} \approx (N+1)[\frac{1-\lambda}{1+\lambda}\xi_{\text{min}} + \frac{\sigma_{\mathbf{w}}^2\sigma_x^2}{2(1-\lambda)}] \tag{74}$$

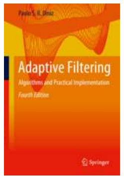
Conventional RLS Algorithm

By differentiating with respect to λ, setting the result to zero, an optimum value for λ can be found

$$\lambda_{\mathrm{opt}} = \frac{1 - \frac{\sigma_{\mathbf{w}} \sigma_x}{2\sigma_n}}{1 + \frac{\sigma_{\mathbf{w}} \sigma_x}{2\sigma_n}} \tag{75}$$

It can be used when it falls in an acceptable range of values for λ.

Complex RLS Algorithm



In the complex data case the objective function is

$$\xi^d(k) = \sum_{i=0}^{k} \lambda^{k-i}|\varepsilon(i)|^2 = \sum_{i=0}^{k} \lambda^{k-i}|d(i) - \mathbf{w}^H(i)\mathbf{x}(k)|^2$$

$$= \sum_{i=0}^{k} \lambda^{k-i} \left[d(i) - \mathbf{w}^H(i)\mathbf{x}(k)\right] \left[d^*(i) - \mathbf{w}^T(i)\mathbf{x}^*(k)\right] \quad (76)$$

Differentiating $\xi^d(k)$ with respect to the complex coefficient $\mathbf{w}(k)$ leads to

$$\frac{\partial \xi^d(k)}{\partial \mathbf{w}(k)} = -\sum_{i=0}^{k} \lambda^{k-i}\mathbf{x}^*(i)[d(i) - \mathbf{w}^H(i)\mathbf{x}(k)] \quad (77)$$

The optimal vector is computed by equating the equation above to zero that is

$$-\sum_{i=0}^{k}\lambda^{k-i}\mathbf{x}(i)\mathbf{x}^{H}(i)\mathbf{w}(k)+\sum_{i=0}^{k}\lambda^{k-i}\mathbf{x}(i)d^{*}(i)=\begin{bmatrix}0\\0\\\vdots\\0\end{bmatrix}$$

leading to the following expression

$$\begin{aligned}\mathbf{w}(k) &= \left[\sum_{i=0}^{k}\lambda^{k-i}\mathbf{x}(i)\mathbf{x}^{H}(i)\right]^{-1}\sum_{i=0}^{k}\lambda^{k-i}\mathbf{x}(i)d^{*}(i)\\ &= \mathbf{R}_D^{-1}(k)\mathbf{p}_D(k)\end{aligned} \tag{78}$$

Complex RLS Algorithm

The matrix inversion lemma to the case of complex data is given by

$$\mathbf{S}_D(k) = \mathbf{R}_D^{-1}(k) = \frac{1}{\lambda}\left[\mathbf{S}_D(k-1) - \frac{\mathbf{S}_D(k-1)\mathbf{x}(k)\mathbf{x}^H(k)\mathbf{S}_D(k-1)}{\lambda + \mathbf{x}^H(k)\mathbf{S}_D(k-1)\mathbf{x}(k)}\right] \tag{79}$$

Complex RLS Algorithm

Algorithm 5.3
Conventional Complex RLS Algorithm

Initialization

$$\mathbf{S}_D(-1) = \delta \mathbf{I}$$

where δ can be the inverse of the input signal power estimate

$$\mathbf{p}_D(-1) = \mathbf{x}(-1) = [0\,0\ldots0]^T$$

Complex RLS Algorithm

Do for $k \geq 0$:

$$\mathbf{S}_D(k) = \frac{1}{\lambda}[\mathbf{S}_D(k-1) - \frac{\mathbf{S}_D(k-1)\mathbf{x}(k)\mathbf{x}^H(k)\mathbf{S}_D(k-1)}{\lambda+\mathbf{x}^H(k)\mathbf{S}_D(k-1)\mathbf{x}(k)}]$$

$$\mathbf{p}_D(k) = \lambda\mathbf{p}_D(k-1) + d^*(k)\mathbf{x}(k)$$

$$\mathbf{w}(k) = \mathbf{S}_D(k)\mathbf{p}_D(k)$$

If necessary compute

$$y(k) = \mathbf{w}^H(k)\mathbf{x}(k)$$

$$\varepsilon(k) = d(k) - y(k)$$

Complex RLS Algorithm

An alternative complex RLS algorithm has an updating equation described by

$$\mathbf{w}(k) = \mathbf{w}(k-1) + e^*(k)\mathbf{S}_D(k)\mathbf{x}(k) \tag{80}$$

where

$$e(k) = d(k) - \mathbf{w}^H(k-1)\mathbf{x}(k) \tag{81}$$

Complex RLS Algorithm

Algorithm 5.4
Alternative Complex RLS Algorithm

Initialization

$$\mathbf{S}_D(-1) = \delta\mathbf{I}$$

where δ can be the inverse of an estimate of the input signal power

$$\mathbf{x}(-1) = \mathbf{w}(-1) = [0\,0 \ldots 0]^T$$

Complex RLS Algorithm

Do for $k \geq 0$

$$
\begin{aligned}
e(k) &= d(k) - \mathbf{w}^H(k-1)\mathbf{x}(k) \\
\boldsymbol{\psi}(k) &= \mathbf{S}_D(k-1)\mathbf{x}(k)
\end{aligned}
$$

Complex RLS Algorithm

$$
\mathbf{S}_D(k) = \frac{1}{\lambda}[\mathbf{S}_D(k-1) - \frac{\boldsymbol{\psi}(k)\boldsymbol{\psi}^H(k)}{\lambda+\boldsymbol{\psi}^H(k)\mathbf{x}(k)}]
$$

$$
\mathbf{w}(k) = \mathbf{w}(k-1) + e^*(k)\mathbf{S}_D(k)\mathbf{x}(k)
$$

If necessary compute

$$
y(k) = \mathbf{w}^H(k)\mathbf{x}(k)
$$

$$
\varepsilon(k) = d(k) - y(k)
$$

Analytical Example

Example 5.3

Assume that an adaptive filter of sufficient order is employed to identify an unknown system of order N, and produces a misadjustment of 10%. Assume the input signal is a white Gausssian noise with unit variance and $\sigma_n^2 = 0.001$.

(a) Compute the value of λ required by the RLS algorithm in order to achieve the desired result when $N = 9$.

(b) For values in the range $0.9 < \lambda < 0.99$, which orders should the adaptive filters have?

Analytical Example

Solution:

(a) The desired misadjustment expression is

$$M = 0.1 = (N+1)\frac{1-\lambda}{1+\lambda}\left(1 + \frac{1-\lambda}{1+\lambda}\mathcal{K}\right) = 10a(1+2a)$$

where $a = \frac{1-\lambda}{1+\lambda}$ and $\mathcal{K} = 2$. By solving this equation we obtain

$$a = \frac{-\frac{1}{2} \pm \sqrt{\frac{1}{4} + 0.02}}{2}$$

where the valid solution is

$$a = \frac{1}{4}\left(-1 + \sqrt{1+0.08}\right) = 0.0098076$$

Analytical Example

Then solving for λ

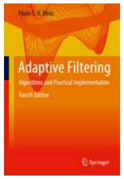
$$\lambda = \frac{1-a}{1+a} = 0.980507$$

By employing the simplest expression we obtain

$$\lambda = \frac{1 - \frac{M}{(N+1)}}{1 + \frac{M}{(N+1)}} = \frac{1-10^{-2}}{1+10^{-2}} = 0.98$$

where M is the misadjustment.

Analytical Example



Solution:

(b) Since

$$\frac{1}{N+1} = \frac{1}{M}\frac{1-\lambda}{1+\lambda}\left(1 + \frac{1-\lambda}{1+\lambda}\mathcal{K}\right) = 10a(1+2a)$$

for $\lambda = 0.90$, $a = 0.052631578$

$$\frac{1}{N+1} = 0.5817$$

so that $N = 0.7190$ and as a result only one coefficient can be employed in the adaptive filter.

Analytical Example

For $\lambda = 0.99$, $a = 0.005025125$,

$$\frac{1}{N+1} = 0.05075$$

so that $N = 18.7$ and as a result 19 coefficients can be employed in the adaptive filter.

Analytical Example

Using the simplest expression for M the results are almost the same since

$$N = M\frac{1+\lambda}{1-\lambda} - 1$$

for $\lambda = 0.90$, $N = 0.9$ meaning that only an adaptive filter with one coefficient would be able to a achieve the desired misadjustment for this value of λ. For $\lambda = 0.99$, $N = 18.9$ meaning that adaptive filters up to order 18 would be able to a achieve the desired misadjustment for this value of λ.

Conventional RLS Algorithm

System Identification Simulation

The conventional RLS algorithm was employed in the identification of the system discussed in the previous Chapter. The forgetting factor was chosen $\lambda = 0.99$.

Conventional RLS Algorithm

The learning curves of the MS *a priori* error are depicted in Fig. 4,

for different values of the eigenvalue spread. Also the measured misadjustment in each example is given in Table 1. From these results we conclude that the RLS algorithm is insensitive to the eigenvalue spread. On Table 1 is given the misadjustment predicted by theory. As can be seen the analytical results are quite accurate.

$$M = (N+1)\frac{1-\lambda}{1+\lambda}(1+\frac{1-\lambda}{1+\lambda}\mathcal{K}) \tag{82}$$

Conventional RLS Algorithm

Table 2 summarizes the results. Note the close agreement between the measurement results and those predicted by the equations

$$E[||\Delta\mathbf{w}(k)_Q||^2] \approx \frac{(1-\lambda)(N+1)}{2\lambda}\,\frac{\sigma_n^2+\sigma_e^2}{\sigma_x^2} + \frac{(N+1)\sigma_{\mathbf{w}}^2}{2\lambda(1-\lambda)} \tag{83}$$

$$\xi(k)_Q \approx \xi_{min}^2 + \sigma_e^2 + \frac{(N+1)\sigma_{\mathbf{w}}^2\sigma_x^2}{2\lambda(1-\lambda)} \tag{84}$$

For the simulations with 12 and 10 bits, the discrepancy between the measured and theoretical estimates of $E[||\Delta\mathbf{w}(k)_Q||^2]$ are caused by the freezing of some tap coefficients.

Conventional RLS Algorithm

The optimal value of λ was too small, $\lambda = 0.99$ was used. The measured excess of MSE was 0.0254, the theoretical value predicted was 0.0418. The theoretical result is not as accurate as before, due to a number of approximations used in the analysis. However, the equation below provides a good indication of what is expected in the practical implementation.

$$\begin{aligned} \xi_{\text{exc}} &\approx (N+1)\frac{1-\lambda}{1+\lambda}(1+\frac{1-\lambda}{1+\lambda}\mathcal{K})\xi_{\text{min}} \\ &+ \frac{(N+1)\sigma_{\mathbf{w}}^2\sigma_x^2}{\lambda_{\mathbf{w}}(1+\lambda^2)-\lambda(1+\lambda_{\mathbf{w}}^2)}(\frac{1-\lambda}{1+\lambda}-\frac{1-\lambda_{\mathbf{w}}}{1+\lambda_{\mathbf{w}}}) \end{aligned} \tag{85}$$

Conventional RLS Algorithm

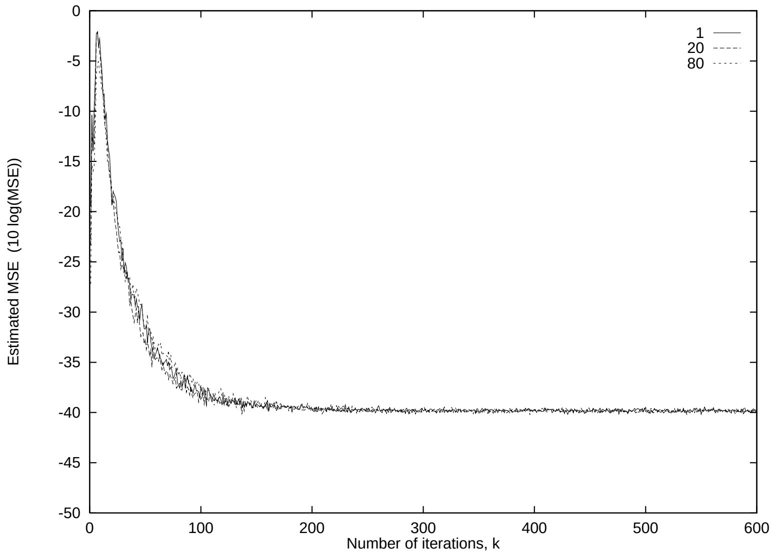


Figure 4: Learning curves for RLS algorithm for eigenvalue spreads: 1, 20, and 80; $\lambda = 0.99$.

Conventional RLS Algorithm

Table 1: Evaluation of the RLS Algorithm

$\frac{\lambda_{max}}{\lambda_{min}}$	Misadjustment	
	Experiment	Theory
1	0.04211	0.04020
20	0.04211	0.04020
80	0.04547	0.04020

Conventional RLS Algorithm

Table 2: Finite Precision Implementation of the RLS Algorithm

No. of bits	$\xi(k)_Q$		$E[\|\|\Delta\mathbf{w}(k)_Q\|\|^2]$	
	Experiment	Theory	Experiment	Theory
16	$1.566\ 10^{-3}$	$1.500\ 10^{-3}$	$6.013\ 10^{-5}$	$6.061\ 10^{-5}$
12	$1.522\ 10^{-3}$	$1.502\ 10^{-3}$	$3.128\ 10^{-5}$	$6.261\ 10^{-5}$
10	$1.566\ 10^{-3}$	$1.532\ 10^{-3}$	$6.979\ 10^{-5}$	$9.272\ 10^{-5}$

Conventional RLS Algorithm

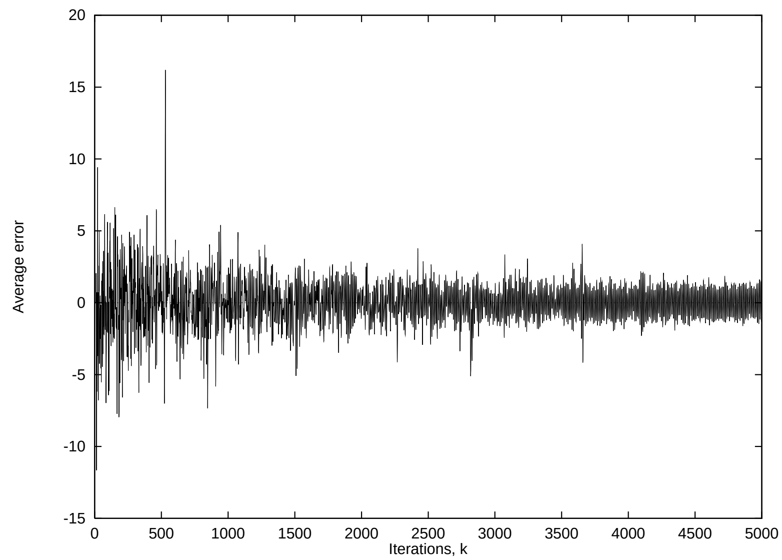
Signal Enhancement Simulations

LMS and RLS Algorithms

We solved the same signal enhancement problem with the conventional RLS and LMS algorithms.

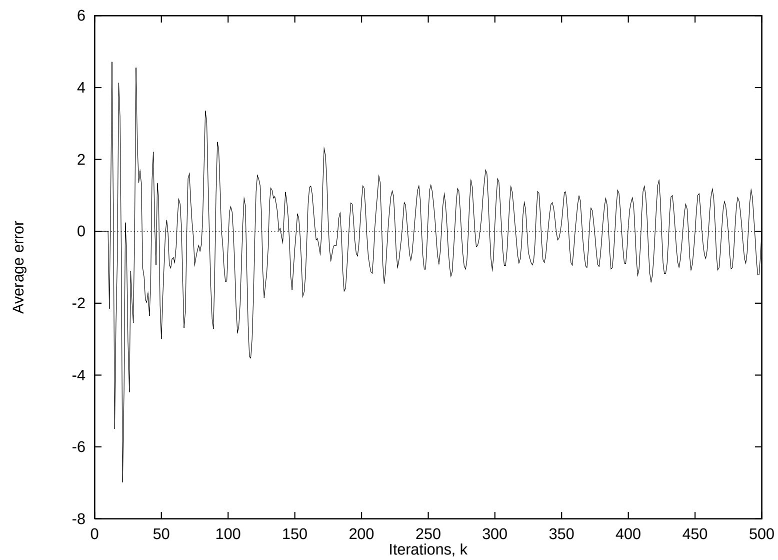
An appropriate value for μ in the LMS case is 0.0204, whereas $\lambda = 1.0$ was used for the RLS. The learning curves for the algorithms are shown in Fig. 5, we see the faster convergence of the RLS. Plotting the output errors after convergence we noted the larger variance of the MSE for the RLS. This result is due to the small signal to noise ratio. Fig. 6 depicts the output error for the RLS.

Conventional RLS Algorithm



(a)

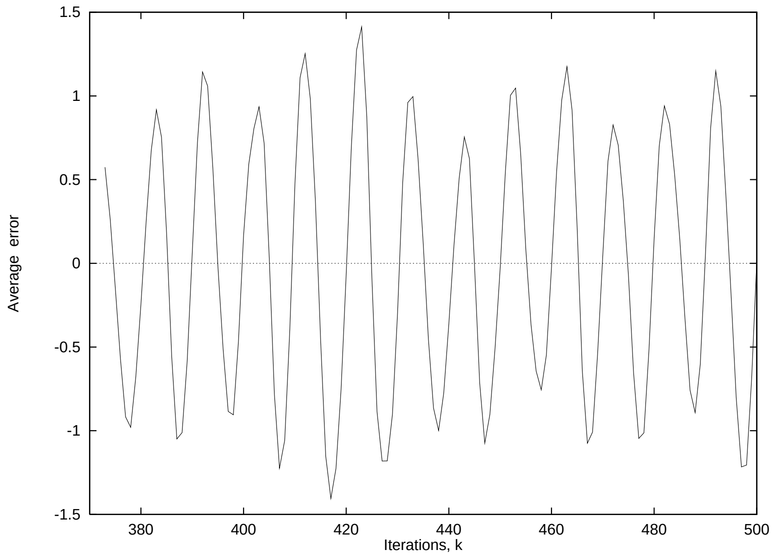
Conventional RLS Algorithm



(b)

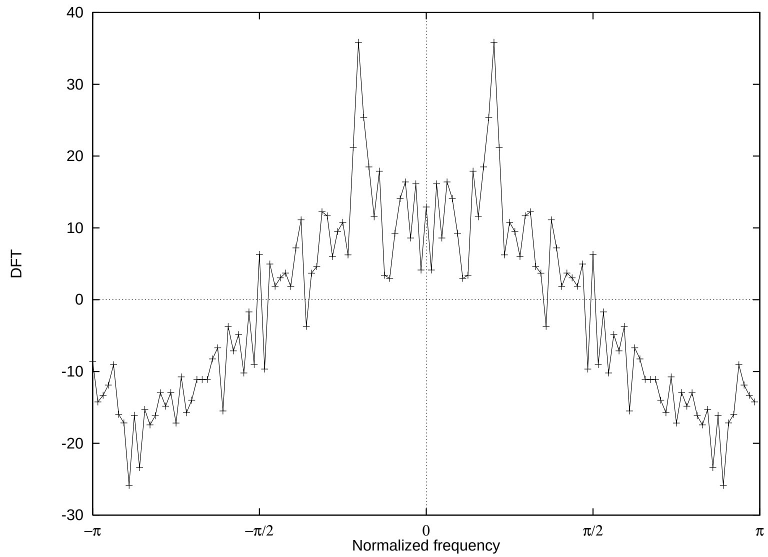
Figure 5: Learning curves for the (a) LMS and (b) RLS algorithms.

Conventional RLS Algorithm



(a)

Conventional RLS Algorithm



(b)

Figure 6: (a) Output error for the RLS algorithm and (b) DFT of the output error.