Chapter 16
Quantization Effects in the RLS Algorithm

16.1 Introduction

In this chapter, several aspects of the finite-wordlength effects in the RLS algorithm are discussed for the cases of implementation with fixed- and floating-point arithmetic [1, 3–6, 8, 9].

16.2 Error Description

All the elements of matrices and vectors in the RLS algorithm will deviate from their correct values due to quantization effects. The error generated in any individual quantization is considered to be a zero-mean random variable that is independent of any other error and quantities related to the adaptive-filter algorithm. The variances of these errors depend on the type of quantization and arithmetic that will be applied in the algorithm implementation.

The errors in the quantities related to the conventional RLS algorithm are defined by

$$n_e(k) = e(k) - e(k)_Q \tag{16.1}$$

$$\mathbf{n}_{\boldsymbol{\psi}}(k) = \mathbf{S}_D(k-1)_Q\mathbf{x}(k) - [\mathbf{S}_D(k-1)_Q\mathbf{x}(k)]_Q \tag{16.2}$$

$$\mathbf{N}_{\mathbf{S}_D}(k) = \mathbf{S}_D(k) - \mathbf{S}_D(k)_Q \tag{16.3}$$

$$\mathbf{n}_{\mathbf{w}}(k) = \mathbf{w}(k) - \mathbf{w}(k)_Q \tag{16.4}$$

$$n_y(k) = y(k) - y(k)_Q \tag{16.5}$$

$$n_\varepsilon(k) = \varepsilon(k) - \varepsilon(k)_Q \tag{16.6}$$

where the subscript Q denotes the quantized form of the given matrix, vector, or scalar.

P.S.R. Diniz, *Adaptive Filtering: Algorithms and Practical Implementation*,
DOI 10.1007/978-1-4614-4106-9_16,

It is assumed that the input signal and desired signal suffer no quantization; so only quantizations of internal computations are taken into account. With the above definitions, the following relations describe the computational error in some quantities of interest related to the RLS algorithm:

$$e(k)_Q = d(k) - \mathbf{x}^T(k)\mathbf{w}(k-1)_Q - n_e(k) \tag{16.7}$$

$$\mathbf{w}(k)_Q = \mathbf{w}(k-1)_Q + \mathbf{S}_D(k)_Q\mathbf{x}(k)e(k)_Q - \mathbf{n_W}(k) \tag{16.8}$$

where $n_e(k)$ is the noise sequence due to quantization in the inner product $\mathbf{x}^T(k)\mathbf{w}(k-1)_Q$ and $\mathbf{n_W}(k)$ is a noise vector due to quantization in the product $\mathbf{S}_D(k)_Q\mathbf{x}(k)e(k)_Q$.

The development here is intended to study the algorithm behavior when the internal signals, vectors, and matrices are available in quantized form as happens in a practical implementation. This means that, for example in Algorithm 5.2, all the information needed from the previous time interval $(k-1)$ to update the adaptive filter at instant k are available in quantized form.

Now we can proceed with the analysis of the deviation in the coefficient vector generated by the quantization error. By defining

$$\Delta\mathbf{w}(k)_Q = \mathbf{w}(k)_Q - \mathbf{w}_o \tag{16.9}$$

and considering that

$$d(k) = \mathbf{x}^T(k)\mathbf{w}_o + n(k)$$

then it follows that

$$e(k)_Q = -\mathbf{x}^T(k)\Delta\mathbf{w}(k-1)_Q - n_e(k) + n(k) \tag{16.10}$$

and

$$\begin{aligned}\Delta\mathbf{w}(k)_Q = {} & \Delta\mathbf{w}(k-1)_Q + \mathbf{S}_D(k)_Q\mathbf{x}(k)[-\mathbf{x}^T(k)\Delta\mathbf{w}(k-1)_Q - n_e(k) + n(k)] \\ & -\mathbf{n_W}(k)\end{aligned} \tag{16.11}$$

(16.11) can be rewritten as follows:

$$\Delta\mathbf{w}(k)_Q = [\mathbf{I} - \mathbf{S}_D(k)_Q\mathbf{x}(k)\mathbf{x}^T(k)]\Delta\mathbf{w}(k-1)_Q + \mathbf{n}'_\mathbf{W}(k) \tag{16.12}$$

where

$$\mathbf{n}'_\mathbf{W}(k) = \mathbf{S}_D(k)_Q\mathbf{x}(k)[n(k) - n_e(k)] - \mathbf{n_W}(k) \tag{16.13}$$

Algorithm 16.1 RLS algorithm including quantization

Initialization

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

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

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

Do for $k \geq 0$

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

$\boldsymbol{\psi}(k)_Q = \mathbf{S}_D(k-1)_Q\mathbf{x}(k) - \mathbf{n}_{\boldsymbol{\psi}}(k)$

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

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

If necessary compute

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

$\varepsilon(k)_Q = d(k) - y_Q(k)$

The solution of (16.12) can be calculated as

$$\Delta\mathbf{w}(k)_Q = \prod_{i=0}^{k}\left[\mathbf{I} - \mathbf{S}_D(i)_Q\mathbf{x}(i)\mathbf{x}^T(i)\right]\Delta\mathbf{w}(-1)_Q + \sum_{i=0}^{k}\left\{\prod_{j=i+1}^{k}[\mathbf{I} - \mathbf{S}_D(j)_Q\mathbf{x}(j)\mathbf{x}^T(j)]\right\}\mathbf{n}'_{\mathbf{w}}(i) \qquad (16.14)$$

where in the last term of the above equation for $i = k$, we consider that

$$\prod_{j=k+1}^{k}[\cdot] = 1$$

Now, if we rewrite Algorithm 5.2 taking into account that any calculation in the present updating generates quantization noise, we obtain Algorithm 16.1 that describes the RLS algorithm with quantization and additional noise taken into account. Notice that Algorithm 16.1 is not a new algorithm.

16.3 Error Models for Fixed-Point Arithmetic

In the case of fixed-point arithmetic, with rounding assumed for quantization, the error after each product can be modeled as a zero-mean stochastic process, with variance given by [2, 7]

$$\sigma^2 = \frac{2^{-2b}}{12} \qquad (16.15)$$

where b is the number of bits after the sign bit. Here it is assumed that the number of bits after the sign bit for quantities representing signals and filter coefficients are different, and given by b_d and b_c, respectively. It is also assumed that the internal signals are properly scaled, so that no overflow occurs during the computations, and that the signal values are between -1 and $+1$. If in addition independence between errors is assumed, each element in (16.1)–(16.6) is on average zero. The respective covariance matrices are given by

$$E[n_e^2(k)] = E[n_\varepsilon^2(k)] = \sigma_e^2 \tag{16.16}$$

$$E[\mathbf{N}_{\mathbf{S}_D}(k)\mathbf{N}^T{}_{\mathbf{S}_D}(k)] = \sigma_{\mathbf{S}_D}^2\mathbf{I} \tag{16.17}$$

$$E[\mathbf{n}_{\mathbf{w}}(k)\mathbf{n}_{\mathbf{w}}^T(k)] = \sigma_{\mathbf{w}}^2\mathbf{I} \tag{16.18}$$

$$E[\mathbf{n}_{\boldsymbol{\psi}}(k)\mathbf{n}_{\boldsymbol{\psi}}^T(k)] = \sigma_{\boldsymbol{\psi}}^2\mathbf{I} \tag{16.19}$$

$$E[n_y^2(k)] = \sigma_y^2 \tag{16.20}$$

If distinction is made between data and coefficient wordlengths, the noise variances of data and coefficients are respectively given by

$$\sigma_e^2 = \sigma_y^2 = \gamma\frac{2^{-2b_d}}{12} \tag{16.21}$$

$$\sigma_{\mathbf{w}}^2 = \gamma'\frac{2^{-2b_c}}{12} \tag{16.22}$$

where $\gamma' = \gamma = 1$ if the quantization is performed after addition, i.e., the products are performed in full precision and the quantization is applied only after all the additions in the inner product are finished. For quantization after each product, then $\gamma = N+1$ and $\gamma' = N+2$, since each quantization in the partial product generates an independent noise, and the number of products in the error computation is $N+1$ whereas in the coefficient computation it is $N+2$.

As an illustration, it is shown how to calculate the value of the variance $\sigma_{\mathbf{S}_D}^2$ when making some simplifying assumptions. The value of $\sigma_{\mathbf{S}_D}^2$ depends on how the computations to generate $\mathbf{S}_D(k)$ are performed. Assume the multiplications and divisions are performed with the same wordlength and that the needed divisions are performed once, followed by the corresponding scalar matrix product. Also, assuming the inner product quantizations are performed after the addition, each element of the matrix $\mathbf{S}_D(k)_Q$ requires five multiplications[1] considering that $1/\lambda$ is prestored. The diagonal elements of (16.17) consist of $N+1$ noise autocorrelations, each with variance $5\sigma_\psi^2$. The desired result is then given by

[1]One is due to the inner product at the denominator; one is due to the division; one is due to the product of the division result by $1/\lambda$; one is to calculate the elements of the outer product of the numerator; the other is the result of quantization of the product of the last two terms.

$$\sigma^2_{\mathbf{S}_D} = 5(N+1)\sigma^2_{\psi} \tag{16.23}$$

where σ^2_{ψ} is the variance of each multiplication error.

16.4 Coefficient-Error-Vector Covariance Matrix

Assume that the quantization signals $n_e(k), n(k)$, and the vector $\mathbf{n_W}(k)$ are all independent of the data, filter coefficients, and each other. Also, assuming that these errors are all zero-mean stochastic processes, the covariance matrix of the coefficient-error vector given by $E[\Delta\mathbf{w}(k)_Q \Delta\mathbf{w}^T(k)_Q]$ can be derived from (16.12) and (16.13)

$$\begin{aligned}
\text{cov}\,[\Delta\mathbf{w}(k)_Q] &= E[\Delta\mathbf{w}(k)_Q \Delta\mathbf{w}^T(k)_Q] \\
&= E\left\{[\mathbf{I} - \mathbf{S}_D(k)_Q\mathbf{x}(k)\mathbf{x}^T(k)]\Delta\mathbf{w}(k-1)_Q\Delta\mathbf{w}^T(k-1)_Q\right. \\
&\quad \left.[\mathbf{I} - \mathbf{x}(k)\mathbf{x}^T(k)\mathbf{S}_D(k)_Q]\right\} \\
&\quad + E[\mathbf{S}_D(k)_Q\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{S}_D(k)_Q]E[n^2(k)] \\
&\quad + E[\mathbf{S}_D(k)_Q\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{S}_D(k)_Q]E[n_e^2(k)] \\
&\quad + E[\mathbf{n_W}(k)\mathbf{n_W}^T(k)]
\end{aligned} \tag{16.24}$$

The above equation can be approximated in the steady state, where each term on the right-hand side will be considered separately. It should be noted that during the derivations it is implicitly assumed that the algorithm follows closely the behavior of its infinite-precision counterpart. This assumption can always be considered as true if the wordlengths used are sufficiently long. However, under short-wordlength implementation this assumption might not be true as will be discussed later on.

Term 1: The elements of $\Delta\mathbf{w}(k-1)_Q$ can be considered independent of $\mathbf{S}_D(k)_Q$ and $\mathbf{x}(k)$. In this case, the first term in (16.24) can be expressed as

$$\begin{aligned}
\mathbf{T}_1 = &\ \text{cov}\,[\Delta\mathbf{w}(k-1)_Q] - \text{cov}\,[\Delta\mathbf{w}(k-1)_Q]E[\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{S}_D(k)_Q] \\
&- E[\mathbf{S}_D(k)_Q\mathbf{x}(k)\mathbf{x}^T(k)]\text{cov}\,[\Delta\mathbf{w}(k-1)_Q] \\
&+ E\left\{\mathbf{S}_D(k)_Q\mathbf{x}(k)\mathbf{x}^T(k)\text{cov}\,[\Delta\mathbf{w}(k-1)_Q]\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{S}_D(k)_Q\right\}
\end{aligned} \tag{16.25}$$

If it is recalled that $\mathbf{S}_D(k)_Q$ is the unquantized $\mathbf{S}_D(k)$ matrix disturbed by a noise matrix that is uncorrelated to the input signal vector, then in order to compute the second and third terms of $\mathbf{T}_1$ it suffices to calculate

$$E[\mathbf{S}_D(k)\mathbf{x}(k)\mathbf{x}^T(k)] \approx E\,[\mathbf{S}_D(k)]\,E\left[\mathbf{x}(k)\mathbf{x}^T(k)\right] \tag{16.26}$$

where the approximation is justified by the fact that $\mathbf{S}_D(k)$ is slowly varying as compared to $\mathbf{x}(k)$ when $\lambda \rightarrow 1$. Using (5.55) it follows that

$$E\left[\mathbf{S}_D(k)\mathbf{x}(k)\mathbf{x}^T(k)\right] \approx \frac{1-\lambda}{1-\lambda^{k+1}}\mathbf{I} \tag{16.27}$$

Now we need to use stronger assumptions for $\mathbf{S}_D(k)$ than those considered in the above equation. If the matrix $E[\mathbf{S}_D(k)_Q]$ is assumed to be approximately constant for large k (see the discussions around (5.54)), the last term in $\mathbf{T}_1$ can be approximated by

$$\begin{aligned} &E\left\{\mathbf{S}_D(k)_Q\mathbf{x}(k)\mathbf{x}^T(k)\text{cov}[\Delta\mathbf{w}(k-1)_Q]\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{S}_D(k)_Q\right\} \\ &\quad\approx E[\mathbf{S}_D(k)_Q]E\left\{\mathbf{x}(k)\mathbf{x}^T(k)\text{cov}\left[\Delta\mathbf{w}(k-1)_Q\right]\mathbf{x}(k)\mathbf{x}^T(k)\right\}E[\mathbf{S}_D(k)_Q] \end{aligned} \tag{16.28}$$

If it is further assumed that the elements of the input signal vector are jointly Gaussian, then each element of the middle term in the last equation can be given by

$$\begin{aligned} &E\left\{\mathbf{x}(k)\mathbf{x}^T(k)\text{cov}\left[\Delta\mathbf{w}(k-1)_Q\right]\mathbf{x}(k)\mathbf{x}^T(k)\right\}_{i,j} \\ &\quad= \sum_{m=0}^{N}\sum_{l=0}^{N}\text{cov}\left[\Delta\mathbf{w}(k-1)_Q\right]_{ml}E[x_i(k)x_m(k)x_l(k)x_j(k)] \\ &\quad= 2\{\mathbf{R}\text{cov}\left[\Delta\mathbf{w}(k-1)_Q\right]\mathbf{R}\}_{i,j} + [\mathbf{R}]_{i,j}\,\text{tr}\left\{\mathbf{R}\text{cov}\left[\Delta\mathbf{w}(k-1)_Q\right]\right\} \end{aligned} \tag{16.29}$$

where $[\cdot]_{i,j}$ denotes the ith, jth element of the matrix $[\cdot]$. It then follows that

$$\begin{aligned} &E\left\{\mathbf{x}(k)\mathbf{x}^T(k)\text{cov}\left[\Delta\mathbf{w}(k-1)_Q\right]\mathbf{x}(k)\mathbf{x}^T(k)\right\} \\ &\quad= 2\mathbf{R}\,\text{cov}\left[\Delta\mathbf{w}(k-1)_Q\right]\mathbf{R} + \mathbf{R}\text{tr}\left\{\mathbf{R}\text{cov}\left[\Delta\mathbf{w}(k-1)_Q\right]\right\} \end{aligned} \tag{16.30}$$

The last term of $\mathbf{T}_1$ in (16.25), after simplified, yields

$$\begin{aligned} &2\left(\frac{1-\lambda}{1-\lambda^{k+1}}\right)^2\text{cov}\left[\Delta\mathbf{w}(k-1)_Q\right] + \left(\frac{1-\lambda}{1-\lambda^{k+1}}\right)^2\text{tr}\left\{\mathbf{R}\text{cov}\left[\Delta\mathbf{w}(k-1)_Q\right]\right\}\mathbf{R}^{-1} \\ &\quad+E\left\{\mathbf{N}_{\mathbf{S}_D}(k)\mathbf{x}(k)\mathbf{x}^T(k)\text{cov}\left[\Delta\mathbf{w}(k-1)_Q\right]\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{N}_{\mathbf{S}_D}(k)\right\} \end{aligned} \tag{16.31}$$

After a few manipulations, it can be shown that the third term in the above equation is nondiagonal with $\mathbf{N}_{\mathbf{S}_D}(k)$ being symmetric for the RLS algorithm described in Algorithm 16.1. On the other hand, if the matrix $\mathbf{R}$ is diagonal dominant, that is in general the case, the third term of (16.31) becomes approximately diagonal and given by[2]

[2]The proof is not relevant but following the lines of (16.30) and considering that its last term is the most relevant, the result follows.

$$\mathbf{T}_S(k) \approx \sigma^2_{\mathbf{S}_D} \sigma^4_x \text{tr}\{\text{cov}\,[\Delta\mathbf{w}(k-1)_Q]\}\mathbf{I} \tag{16.32}$$

where σ^2_x is the variance of the input signal. This term, which is proportional to a quantization noise variance, can actually be neglected in the analysis, since it has in general much smaller norm than the remaining terms in $\mathbf{T}_1$.

Terms 2 and 3: Using the same arguments applied before, such as $\mathbf{S}_D(k)$ is almost fixed as $\lambda \to 1$, then the main result required to calculate the terms 2 and 3 of (16.24) is approximately given by

$$\begin{aligned} E[\mathbf{S}_D(k)_Q\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{S}_D(k)_Q] &\approx E[\mathbf{S}_D(k)]\mathbf{R}E[\mathbf{S}_D(k)] + E[\mathbf{N}_{\mathbf{S}_D}(k)\mathbf{R}\mathbf{N}_{\mathbf{S}_D}(k)] \\ &\approx \left(\frac{1-\lambda}{1-\lambda^{k+1}}\right)^2 \mathbf{R}^{-1} \end{aligned} \tag{16.33}$$

where the term $E[\mathbf{N}_{\mathbf{S}_D}(k)\mathbf{R}\mathbf{N}_{\mathbf{S}_D}(k)]$ can be neglected because it is in general much smaller than the remaining term. In addition, it will be multiplied by a small variance when (16.33) is replaced back in (16.24). From (16.24), (16.28), (16.33), (16.16), (16.18), and (16.22), it follows that

$$\begin{aligned} \text{cov}\,[\Delta\mathbf{w}(k)_Q] = &\left[1 - 2\left(\frac{1-\lambda}{1-\lambda^{k+1}}\right) + 2\left(\frac{1-\lambda}{1-\lambda^{k+1}}\right)^2\right] \text{cov}\,[\Delta\mathbf{w}(k-1)_Q] \\ &+ \left(\frac{1-\lambda}{1-\lambda^{k+1}}\right)^2 \text{tr}\,\{\mathbf{R}\text{cov}\,[\Delta\mathbf{w}(k-1)_Q]\}\mathbf{R}^{-1} \\ &+ \left(\frac{1-\lambda}{1-\lambda^{k+1}}\right)^2 (\sigma^2_n + \sigma^2_e)\mathbf{R}^{-1} + \sigma^2_{\mathbf{w}}\mathbf{I} \end{aligned} \tag{16.34}$$

Now, by considering in (16.34) that in the steady state $\text{cov}\,[\Delta\mathbf{w}(k)_Q] \approx \text{cov}\,[\Delta\mathbf{w}(k-1)_Q]$, multiplying the resulting expression by $\mathbf{R}$, and calculating the trace of the final equation, it can be shown that

$$\text{tr}\,\{\mathbf{R}\,\text{cov}\,[\Delta\mathbf{w}(k-1)_Q]\} \approx \frac{(1-\lambda)^2(N+1)(\sigma^2_n+\sigma^2_e) + \sigma^2_{\mathbf{w}}\text{tr}\,(\mathbf{R})}{(1-\lambda)[2\lambda-(1-\lambda)(N+1)]} \tag{16.35}$$

where it was considered that $\lambda^{k+1} \to 0$. Replacing the (16.35) in (16.34), and computing the steady-state solution the following equation results

$$\begin{aligned} \text{cov}\,[\Delta\mathbf{w}(k)_Q] \approx &\frac{(1-\lambda)(\sigma^2_n+\sigma^2_e)}{2\lambda-(1-\lambda)(N+1)}\mathbf{R}^{-1} \\ &+ \frac{(1-\lambda)\text{tr}\,(\mathbf{R})\mathbf{R}^{-1} + [2\lambda-(1-\lambda)(N+1)]\mathbf{I}}{2(1-\lambda)\lambda[2\lambda-(1-\lambda)(N+1)]}\sigma^2_{\mathbf{w}} \end{aligned} \tag{16.36}$$

Finally, if the trace of the above equation is calculated considering that $x(k)$ is a Gaussian white noise with variance σ_x^2, and that $2\lambda \gg (1-\lambda)(N+1)$ for $\lambda \rightarrow 1$, the resulting expected value of $||\Delta\mathbf{w}(k)_Q||^2$ is

$$E[||\Delta\mathbf{w}(k)_Q||^2] \approx \frac{(1-\lambda)(N+1)}{2\lambda}\frac{\sigma_n^2+\sigma_e^2}{\sigma_x^2} + \frac{(N+1)\sigma_{\mathbf{w}}^2}{2\lambda(1-\lambda)} \tag{16.37}$$

As can be noted if the value of λ is very close to one, the square errors in the tap coefficients tend to increase and to become more dependent of the tap coefficient wordlengths. On the other hand, if λ is not close to one, in general for fast tracking purposes, the effects of the additive noise and data wordlength become more disturbing to the coefficient square errors. The optimum value for λ close to 1, as far as quantization effects are concerned, can be derived by calculating the derivative of $E||\Delta\mathbf{w}(k)_Q||^2]$ with respect to λ and setting the result to zero

$$\lambda_{\text{opt}} \approx 1 - \frac{\sigma_{\mathbf{w}}\sigma_x}{\sqrt{\sigma_n^2+\sigma_e^2}} \tag{16.38}$$

where it was assumed that $(2\lambda - 1) \approx 1$.

By noting that $\frac{1-\lambda}{1-\lambda^{k+1}}$ should be replaced by $\frac{1}{k+1}$ when $\lambda = 1$, it can be shown from (16.34) that the algorithm tends to diverge when $\lambda = 1$, since in this case $||\text{cov}\,[\Delta\mathbf{w}(k)_Q]||$ is growing with k.

16.5 Algorithm Stop

In some cases the adaptive-filter tap coefficients may stop adapting due to quantization effects. In particular, the conventional RLS algorithm will freeze when the coefficient updating term is not representable with the available wordlength. This occurs when its modulus is smaller than half the value of the least significant bit, i.e.,

$$|e(k)_Q\mathbf{S}_D(k)_Q\mathbf{x}(k)|_i < 2^{-b_c-1} \tag{16.39}$$

where $|\ |_i$ denotes the modulus of the ith component. Equivalently it can be concluded that updating will be stopped if

$$E[e(k)_Q^2]E[|\mathbf{S}_D(k)_Q\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{S}_D(k)_Q|_{ii}] \approx \left(\frac{1-\lambda}{1-\lambda^{k+1}}\right)^2\frac{\sigma_e^2+\sigma_n^2}{\sigma_x^2} < 2^{-2b_c-2} \tag{16.40}$$

where $x(k)$ was considered a Gaussian white noise with variance σ_x^2, and the following approximation was made: $E[e(k)_Q^2] \approx \sigma_e^2 + \sigma_n^2$.

For a given coefficient wordlength b_c, the algorithm can always be kept updating if

$$\lambda < 1 - 2^{-b_c-1} \frac{\sigma_x}{\sqrt{\sigma_e^2 + \sigma_n^2}} \tag{16.41}$$

On the other hand, if the above condition is not satisfied, it can be expected that the algorithm will stop updating in

$$k \approx \frac{\sqrt{\sigma_e^2 + \sigma_n^2}}{\sigma_x} 2^{b_c+1} - 1 \tag{16.42}$$

iterations for $\lambda = 1$, and

$$k \approx \frac{\ln[(\lambda - 1)\frac{\sqrt{\sigma_e^2+\sigma_n^2}}{\sigma_x} 2^{b_c+1} + 1]}{\ln \lambda} - 1 \tag{16.43}$$

iterations for $\lambda < 1$.

In the case $\lambda = 1$ the algorithm always stops updating. If σ_n^2 and b_c are not large, any steady-state analysis for the RLS algorithm when $\lambda = 1$ does not apply, since the algorithm stops prematurely. Because of that, the norm of the covariance of $\Delta\mathbf{w}(k)_Q$ does not become unbounded.

16.6 Mean-Square Error

The MSE in the conventional RLS algorithm in the presence of quantization noise is given by

$$\xi(k)_Q = E[\varepsilon^2(k)_Q] \tag{16.44}$$

By recalling that $\varepsilon(k)_Q$ can be expressed as

$$\varepsilon(k)_Q = -\mathbf{x}^T(k)\Delta\mathbf{w}(k)_Q - n_e(k) + n(k) \tag{16.45}$$

it then follows that

$$\begin{aligned}
\xi(k)_Q &= E[\mathbf{x}^T(k)\Delta\mathbf{w}(k)_Q\mathbf{x}^T(k)\Delta\mathbf{w}(k)_Q] + \sigma_e^2 + \xi_{\min} \\
&= E\left\{\text{tr}\left[\mathbf{x}(k)\mathbf{x}^T(k)\Delta\mathbf{w}(k)_Q\Delta\mathbf{w}^T(k)_Q\right]\right\} + \sigma_e^2 + \xi_{\min} \\
&= \text{tr}\left\{\mathbf{R}\,\text{cov}\left[\Delta\mathbf{w}(k)_Q\right]\right\} + \sigma_e^2 + \xi_{\min}
\end{aligned} \tag{16.46}$$

By replacing (16.35) in (16.46), it can be concluded that

$$\xi(k)_Q = \frac{(1-\lambda)^2(N+1)(\sigma_n^2 + \sigma_e^2) + \sigma_{\mathbf{w}}^2 \text{tr}\,\mathbf{R}}{(1-\lambda)[2\lambda - (1-\lambda)(N+1)]} + \xi_{\min} + \sigma_e^2 \tag{16.47}$$

If it is again assumed that $x(k)$ is a Gaussian white noise with variance σ_x^2 and that $2\lambda \gg (1-\lambda)(N+1)$ for $\lambda \rightarrow 1$, the MSE expression can be simplified to

$$\xi(k)_Q \approx \xi_{\min} + \sigma_e^2 + \frac{(N+1)\sigma_{\mathbf{w}}^2 \sigma_x^2}{2\lambda(1-\lambda)} \tag{16.48}$$

16.7 Fixed-Point Implementation Issues

The implementation of the conventional RLS algorithm in fixed-point arithmetic must consider the possibility of occurrence of overflow and underflow during the computations. In general, some scaling must be performed in certain quantities of the RLS algorithm to avoid undesired behavior due to overflow and underflow. The scaling procedure must be applied in almost all computations required in the conventional RLS algorithm [5], increasing the computational complexity and/or the implementation control by a large amount. A possible solution is to leave enough room in the integer and fractional parts of the number representation, in order to avoid frequent overflows and underflows and also avoid the use of cumbersome scaling strategies. In other words, a fixed-point implementation does require a reasonable number of bits to represent each quantity.

The error propagation analysis can be performed by studying the behavior of the difference between each quantity of the algorithm calculated in infinite precision and finite precision. This analysis allows the detection of divergence of the algorithm due to quantization error accumulation. The error propagation analysis for the conventional RLS algorithm reveals divergence behavior linked to the fact that $\mathbf{S}_D(k)$ loses the positive definiteness property [5]. The main factors contributing to divergence are:

- Large maximum eigenvalue in the matrix $\mathbf{R}$ that amplifies some terms in propagation error of the $\mathbf{S}_D(k)$ matrix. In this case, $\mathbf{S}_D(k)$ might have a small minimum eigenvalue, being as consequence "almost" singular.
- A small number of bits used in the calculations increases the roundoff noise contributing to divergence.
- The forgetting factor when small turns the memory of the algorithm short, making the matrix $\mathbf{S}_D(k)$ deviate from its expected steady-state value and more likely to lose the positive definiteness property.

Despite these facts, the conventional RLS algorithm can be implemented without possibility of divergence if some special quantization strategies for the internal computations are used [5]. These quantization strategies, along with adaptive scaling strategies, must be used when implementing the conventional RLS algorithm in fixed-point arithmetic with short wordlength.

16.8 Floating-Point Arithmetic Implementation

In this section, a succinct analysis of the quantization effects in the conventional RLS algorithm when implemented in floating-point arithmetic is presented. Most of the derivations are given in Sect. 16.9 and follow closely the procedure of the fixed-point analysis.

In floating-point arithmetic, quantization errors are injected after multiplication and addition operations and are modeled as follows: [10]:

$$\text{fl}[a+b] = a + b - (a+b)n_a \tag{16.49}$$

$$\text{fl}[a \cdot b] = a \cdot b - a \cdot b \cdot n_p \tag{16.50}$$

where n_a and n_p are zero-mean random variables that are independent of any other errors. Their variances are given by

$$\sigma_{n_p}^2 \approx 0.18\, 2^{-2b} \tag{16.51}$$

and

$$\sigma_{n_a}^2 < \sigma_{n_p}^2 \tag{16.52}$$

where b is the number of bits in the mantissa representation.

The quantized error and the quantized coefficient vector are given by

$$e(k)_Q = d'(k) - \mathbf{x}^T(k)\mathbf{w}(k-1)_Q - n_e(k) + n(k) \tag{16.53}$$

$$\mathbf{w}(k)_Q = \mathbf{w}(k-1)_Q + \mathbf{S}_D(k)_Q\mathbf{x}(k)e(k)_Q - \mathbf{n_w}(k) \tag{16.54}$$

where $n_e(k)$ and $\mathbf{n_w}(k)$ represent computational errors and their expressions are given in Sect. 16.9. Since $\mathbf{n_w}(k)$ is a zero-mean vector, it is shown in Sect. 16.9 that on average $\mathbf{w}(k)_Q$ tends to $\mathbf{w}_o$. Also, it can be shown that

$$\begin{aligned}\Delta\mathbf{w}(k)_Q &= [\mathbf{I} - \mathbf{S}_D(k)_Q\mathbf{x}(k)\mathbf{x}^T(k) + \mathbf{N}_{\Delta\mathbf{w}}(k)]\Delta\mathbf{w}(k-1) \\ &\quad + \mathbf{N}'_a(k)\mathbf{w}_o + \mathbf{S}_D(k)_Q\mathbf{x}(k)[n(k) - n_e(k)]\end{aligned} \tag{16.55}$$

where $\mathbf{N}_{\Delta\mathbf{w}}(k)$ combines several quantization noise effects as discussed in Sect. 16.9 and $\mathbf{N}'_a(k)$ is a diagonal noise matrix that models the noise generated in the vector addition required to update $\mathbf{w}(k)_Q$.

The covariance matrix of $\Delta\mathbf{w}(k)_Q$ can be calculated through the same procedure previously used in the fixed-point case, resulting in

$$\begin{aligned}\text{cov}[\Delta\mathbf{w}(k)_Q] &\approx \frac{(1-\lambda)(\sigma_n^2 + \sigma_e^2)\mathbf{R}^{-1}}{2\lambda - (1-\lambda)(N+1)} \\ &+ \frac{(1-\lambda)\mathbf{R}^{-1}\text{tr}\left\{\mathbf{R}\text{diag}[w_{oi}^2]\right\} + [2\lambda - (1-\lambda)(N+1)]\text{diag}[w_{oi}^2]}{2(1-\lambda)\lambda[2\lambda - (1-\lambda)(N+1)]}\sigma_{n'_a}^2\end{aligned} \tag{16.56}$$

where $\mathbf{N}_{\mathbf{S}_D}(k)$ of (16.3) and $\mathbf{N}_{\Delta\mathbf{w}}(k)$ were considered negligible as compared to the remaining matrices multiplying $\Delta\mathbf{w}(k-1)$ in (16.55). The expression of $\sigma^2_{n'_a}$ is given by (16.52). The term diag$[w^2_{oi}]$ represents a diagonal matrix formed with the squared elements of $\mathbf{w}_o$.

The expected value of $||\Delta\mathbf{w}(k)_Q||^2$ in the floating-point case is approximately given by

$$E[||\Delta\mathbf{w}(k)_Q||^2] \approx \frac{(1-\lambda)(N+1)}{2\lambda}\frac{\sigma_n^2+\sigma_e^2}{\sigma_x^2} + \frac{1}{2\lambda(1-\lambda)}||\mathbf{w}_o||^2\sigma^2_{n'_a} \tag{16.57}$$

where it was considered that $x(k)$ is a Gaussian white noise with variance σ_x^2 and that $2\lambda \gg (1-\lambda)(N+1)$ for $\lambda \to 1$. If the value of λ is very close to one, the squared errors in the tap coefficients tend to increase. Notice that the second term on the right-hand side of the above equation turns these errors more dependent on the precision of the vector addition of the taps updating. For λ not very close to one, the effects of the additive noise and data wordlength become more pronounced. In floating-point implementation, the optimal value of λ as far as quantization effects are concerned is given by

$$\lambda_{\text{opt}} = 1 - \frac{\sigma_{n'_a}\sigma_x}{\sqrt{\sigma_n^2+\sigma_e^2}}||\mathbf{w}_o|| \tag{16.58}$$

where this relation was obtained by calculating the derivative of (16.57) with respect to λ, and equalizing the result to zero in order to reach the value of λ that minimizes the $E[||\Delta\mathbf{w}(k)_Q||^2]$. For $\lambda = 1$, like in the fixed-point case, $||\text{cov}\,[\Delta\mathbf{w}(k)_Q]||$ is also a growing function that can make the conventional RLS algorithm diverge.

The algorithm may stop updating if

$$|e(k)_Q\mathbf{S}_D(k)\mathbf{x}(k)|_i < 2^{-b_c-1}w_i(k) \tag{16.59}$$

where $|\ |_i$ is the modulus of the ith component and b_c is the number of bits in the mantissa of the coefficients representation. Following the same procedure to derive (16.40), we can infer that the updating will be stopped if

$$\left(\frac{1-\lambda}{1-\lambda^{k+1}}\right)^2\frac{\sigma_e^2+\sigma_n^2}{\sigma_x^2} < 2^{-2b_c-2}|w_{oi}|^2 \tag{16.60}$$

where w_{oi} is the ith element of $\mathbf{w}_o$.

The updating can be continued indefinitely if

$$\lambda < 1 - 2^{-b_c-1}\frac{\sigma_x|w_{oi}|}{\sqrt{\sigma_e^2+\sigma_n^2}} \tag{16.61}$$

In the case λ does not satisfy the above condition, the algorithm will stop updating the ith tap in approximately

$$k = \frac{\sqrt{\sigma_e^2+\sigma_n^2}}{\sigma_x|w_{oi}|} - 1 \tag{16.62}$$

iterations for $\lambda = 1$, and

$$k \approx \frac{\ln[(\lambda - 1)\frac{\sqrt{\sigma_e^2+\sigma_n^2}}{\sigma_x|w_{oi}|}2^{-b_c-1} + 1]}{\ln \lambda} - 1 \tag{16.63}$$

iterations for $\lambda < 1$.

Following the same procedure as in the fixed-point implementation, it can be shown that the MSE in the floating-point case is given by

$$\begin{aligned}\xi(k)_Q &= \text{tr}\{\mathbf{R}\text{cov}[\Delta\mathbf{w}(k)_Q]\} + \sigma_e^2 + \xi_{\min} \\ &\approx \frac{(1-\lambda)^2(N+1)(\sigma_n^2+\sigma_e^2) + \sigma_{n'_a}^2 \text{tr}\left\{\mathbf{R}\text{diag}[w_{oi}^2]\right\}}{(1-\lambda)[2\lambda - (1-\lambda)(N+1)]} + \sigma_e^2 + \xi_{\min}\end{aligned} \tag{16.64}$$

where σ_ε^2 was considered equal to σ_e^2. If $x(k)$ is a Gaussian white noise with variance σ_x^2 and $2\lambda \gg (1-\lambda)(N+1)$ for $\lambda \to 1$, the MSE can be approximated by

$$\xi(k)_Q \approx \xi_{\min} + \sigma_e^2 + \frac{||\mathbf{w}_o||^2\sigma_{n'_a}^2\sigma_x^2}{2\lambda(1-\lambda)} \tag{16.65}$$

Note that σ_e^2 has a somewhat complicated expression that is given in Sect. 16.9.

Finally, it should be mentioned that in floating-point implementations the matrix $\mathbf{S}_D(k)$ can also lose its positive definite property [11]. In [5], it was mentioned that if no interactions between errors is considered, preserving the symmetry of $\mathbf{S}_D(k)$ is enough to keep it positive definite. However, interactions between errors do exist in practice, so the conventional RLS algorithm can become unstable in floating-point implementations unless some special quantization procedures are employed in the actual implementation. An alternative is to use numerically stable RLS algorithms discussed in Chaps. 7–9.

16.9 Floating-Point Quantization Errors in RLS Algorithm

The error in the a priori output error computation is given by

$$\begin{aligned}n_e(k) \approx& -n_a(k)[d(k) - \mathbf{x}^T(k)\mathbf{w}(k-1)_Q] \\ &-\mathbf{x}^T(k)\begin{bmatrix} n_{p_o}(k) & 0 & 0 \cdots & 0 \\ 0 & n_{p_1}(k) & \cdots\cdots & 0 \\ \vdots & & \ddots & \\ 0 & 0 & \cdots\cdots & n_{p_N}(k)\end{bmatrix}\mathbf{w}(k-1)_Q\end{aligned}$$

$$-[n_{a_1}(k)\, n_{a_2}(k) \ldots n_{a_N}(k)] \begin{bmatrix} \sum_{i=0}^{1} x(k-i)w_i(k-1)_Q \\ \sum_{i=0}^{2} x(k-i)w_i(k-1)_Q \\ \vdots \\ \sum_{i=0}^{N} x(k-i)w_i(k-1)_Q \end{bmatrix}$$

$$= -n_a(k)e(k)_Q - \mathbf{x}^T(k)\mathbf{N}_p(k)\mathbf{w}(k-1)_Q - \mathbf{n}_a(k)\mathbf{s}_i(k)$$

where $n_{p_i}(k)$ accounts for the noise generated in the products $x(k-i)w_i(k-1)_Q$ and $n_{a_i}(k)$ accounts for the noise generated in the additions of the product $\mathbf{x}^T(k)\mathbf{w}(k-1)$. Please note that the error terms of second- and higher-order have been neglected.

Using similar assumptions one can show that

$$\begin{aligned} \mathbf{n}_{\mathbf{W}}(k) = -\big\{&\mathbf{n}_{Sx}(k)e(k)_Q + \mathbf{S}_D(k)_Q\mathbf{N}'_p(k)\mathbf{x}(k)e(k)_Q \\ &+\mathbf{N}''_p(k)\mathbf{S}_D(k)_Q\mathbf{x}(k)e(k)_Q \\ &+ \mathbf{N}'_a(k)[\mathbf{w}(k-1) + \mathbf{S}_D(k)_Q\mathbf{x}(k)e(k)_Q]\big\} \end{aligned}$$

where

$$\mathbf{n}_{Sx}(k) = \begin{bmatrix} \sum_{j=1}^{N} n'_{a_{1,j}}(k) \sum_{i=0}^{j} \mathbf{S}_{D_{1,i}}(k)_Q x(k-i) \\ \vdots \\ \sum_{j=1}^{N} n'_{a_{N+1,j}}(k) \sum_{i=0}^{j} \mathbf{S}_{D_{N+1,i}}(k)_Q x(k-i) \end{bmatrix}$$

$$\mathbf{N}'_a(k) = \begin{bmatrix} n'_{a_o}(k) & 0 & \cdots & 0 \\ 0 & n'_{a_1}(k) & & \vdots \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & \cdots & n'_{a_N}(k) \end{bmatrix}$$

$$\mathbf{N}'_p(k) = \begin{bmatrix} n'_{p_o}(k) & 0 & \cdots & 0 \\ 0 & n'_{p_1}(k) & & \vdots \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & \cdots & n'_{p_N}(k) \end{bmatrix}$$

$$\mathbf{N}''_p(k) = \begin{bmatrix} n''_{p_{1,1}}(k) & n''_{p_{1,2}}(k) & \cdots & n''_{p_{1,N+1}}(k) \\ n''_{p_{2,1}}(k) & n''_{p_{2,2}}(k) & & \vdots \\ \vdots & & \ddots & \vdots \\ n''_{p_{N+1,1}}(k) & \cdots & \cdots & n''_{p_{N+1,N+1}}(k) \end{bmatrix}$$

The vector $\mathbf{n}_{Sx}(k)$ is due to the quantization of additions in the matrix product $\mathbf{S}_D(k)\mathbf{x}(k)$, while the matrix $\mathbf{N}''_p(k)$ accounts for product quantizations in the same operation. The matrix $\mathbf{N}'_a(k)$ models the error in the vector addition to generate $\mathbf{w}(k)_Q$, while $\mathbf{N}'_p(k)$ models the quantization in the product of $e(k)$ by $\mathbf{S}_D(k)_Q\mathbf{x}(k)$.

By replacing $d'(k)$ by $\mathbf{x}^T(k)\mathbf{w}_o$ in the expression of $e(k)_Q$ given in (16.7), it follows that

$$e(k)_Q = -\mathbf{x}^T(k)\Delta\mathbf{w}(k-1)_Q - n'_e(k) + n(k)$$

By using in the above equation the expression of $\mathbf{w}(k)_Q$ of (16.8) (after subtracting $\mathbf{w}_o$ in each side of the equation), and neglecting the second- and higher-order errors, after some manipulations the following equality results:

$$\begin{aligned}\Delta\mathbf{w}(k)_Q = [&\mathbf{I} - \mathbf{S}_D(k)_Q\mathbf{x}(k)\mathbf{x}^T(k) + \mathbf{n}_{Sx}\mathbf{x}^T(k) + \mathbf{S}_D(k)_Q\mathbf{N}'_p(k)\mathbf{x}(k)\mathbf{x}^T(k) \\ &+\mathbf{N}''_p(k)\mathbf{S}_D(k)_Q\mathbf{x}(k)\mathbf{x}^T(k) + \mathbf{N}'_a(k)\mathbf{S}_D(k)_Q\mathbf{x}(k)\mathbf{x}^T(k) \\ &+\mathbf{N}'_a(k)]\Delta\mathbf{w}(k-1)_Q + \mathbf{N}'_a(k)\mathbf{w}_o + \mathbf{S}_D(k)_Q\mathbf{x}(k)[n(k) - n'_e(k)]\end{aligned}$$

Since all the noise components in the above equation have zero mean, on average the tap coefficients will converge to their optimal values because the same dynamic equation describes the evolution of $\Delta\mathbf{w}(k)$ and $\Delta\mathbf{w}(k)_Q$.

Finally, the variance of the a priori error noise can be derived as follows:

$$\begin{aligned}\sigma_e^2 = \sigma_\varepsilon^2 = &\sigma_{n_a}^2\xi(k)_Q + \sigma_{n_p}^2\sum_{i=0}^{N}\mathbf{R}_{i,i}\mathrm{cov}\,[\mathbf{w}(k)_Q]_{i,i} \\ &+\sigma_{n_a}^2\left\{E\left[\left(\sum_{i=0}^{1}x(k-i)w_i(k-1)_Q\right)^2\right]\right. \\ &+E\left[\left(\sum_{i=0}^{2}x(k-i)w_i(k-1)_Q\right)^2\right] \\ &\left.+\cdots+E\left[\left(\sum_{i=0}^{N}x(k-i)w_i(k-1)_Q\right)^2\right]\right\}\end{aligned}$$

where $\sigma^2_{n'_{ai}} = \sigma^2_{n_a}$ was used and $[\ ]_{i,i}$ means diagonal elements of [ ]. The second term in the above equation can be further simplified as follows:

$$\text{tr}\{\mathbf{R}\text{cov}[\mathbf{w}(k)_Q]\} \approx \sum_{i=0}^{N} \mathbf{R}_{i,i} w_{oi}^2 + \sum_{i=0}^{N} \mathbf{R}_{i,i} \text{cov}[\Delta\mathbf{w}(k)]_{i,i}$$
$$+\text{first}- \text{ and higher}-\text{order terms } \cdots$$

Since this term is multiplied by $\sigma^2_{n_p}$, any first- and higher-order terms can be neglected. The first term of σ_e^2 is also small in the steady state. The last term can be rewritten as

$$\sigma^2_{n_a} \left\{ E\left[\left(\sum_{i=0}^{1} x(k-i)w_{oi}\right)^2\right] + E\left[\left(\sum_{i=0}^{2} x(k-i)w_{oi}\right)^2\right] + \cdots \right.$$
$$\left. +E\left[\left(\sum_{i=0}^{N} x(k-i)w_{oi}\right)^2\right]\right\} = \sigma^2_{n_a} \left\{ \sum_{j=1}^{N} \sum_{i=0}^{j} \mathbf{R}_{i,i} [\text{cov}(\Delta\mathbf{w}(k))]_{i,i} \right\}$$

where terms of order higher than one were neglected, $x(k)$ was considered uncorrelated to $\Delta\mathbf{w}(k)$, and $\text{cov}[\Delta\mathbf{w}(k)]$ was considered a diagonal matrix. Actually, if $x(k)$ is considered a zero-mean Gaussian white noise from the proof of (5.36) and (5.55), it can be shown that

$$\text{cov}[\Delta\mathbf{w}(k)] \approx \frac{\sigma_n^2}{\sigma_x^2}\mathbf{I}$$

Since this term will be multiplied by $\sigma^2_{n_a}$ and $\sigma^2_{n_p}$, it can also be disregarded. In conclusion

$$\sigma_e^2 \approx \sigma^2_{n_a} \left\{ E[\sum_{j=1}^{N}(\sum_{i=0}^{j} x(k-i)w_{oi})^2] \right\} + \sigma^2_{n_p} \sum_{i=0}^{N} \mathbf{R}_{i,i} w_{oi}^2$$

This equation can be simplified further when $x(k)$ is as described above and $\sigma^2_{na} = \sigma^2_{n_p} = \sigma^2_d$

$$\sigma_e^2 \approx \sigma_d^2 \left[\sum_{i=1}^{N}(N-i+2)\mathbf{R}_{i,i} w_{oi}^2 - \mathbf{R}_{1,1} w_{o1}^2\right]$$
$$= \sigma_d^2 \sigma_x^2 \left[\sum_{i=1}^{N}(N-i+2) w_{oi}^2 - w_{o1}^2\right]$$

References

1. T. Adali, S.H. Ardalan, Steady state and convergence characteristics of the fixed-point RLS algorithm, in *Proceedings of the IEEE International Symposium on Circuits Systems*, New Orleans, LA, May 1990, pp. 788–791
2. A. Antoniou, *Digital Signal Processing: Signals, Systems, and Filters* (McGraw Hill, New York, 2005)
3. S.H. Ardalan, Floating-point analysis of recursive least-squares and least-mean squares adaptive filters. IEEE Trans. Circ. Syst. **CAS-33**, 1192–1208 (1986)
4. S.H. Ardalan, S.T. Alexander, Fixed-point roundoff error analysis of the exponentially windowed RLS algorithm for time-varying systems. IEEE Trans. Acoust. Speech Signal Process. **ASSP-35**, 770–783 (1983)
5. G.E. Bottomley, S.T. Alexander, A novel approach for stabilizing recursive least squares filters. IEEE Trans. Signal Process. **39**, 1770–1779 (1991)
6. J.M. Cioffi, Limited precision effects in adaptive filtering. IEEE Trans. Circ. Syst. **CAS-34**, 821–833 (1987)
7. P.S.R. Diniz, E.A.B. da Silva, S.L. Netto, *Digital Signal Processing: System Analysis and Design* (Cambridge University Press, Cambridge, 2002)
8. G. Kubin, Stabilization of the RLS algorithm in the absence of persistent excitation, in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, NY, April 1988, pp. 1369–1372
9. F. Ling, J.G. Proakis, Numerical accuracy and stability: Two problems of adaptive estimation algorithms caused by round-off error, in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, San Diego, CA, March 1984, pp. 30.3.1–30.3.4
10. A.B. Spirad, D.L. Snyder, Quantization errors in floating-point arithmetic. IEEE Trans. Acoust. Speech Signal Process. **ASSP-26**, 456–464 (1983)
11. M.H. Verhaegen, Round-off error propagation in four generally applicable, recursive, least squares estimation schemes. Automatica **25**, 437–444 (1989)