Published in Proc. Allerton Conference on Communication, Control, and Computing, vol. II,

pp. 1175–1183, Allerton, IL, Oct. 2000

An Adaptive Filter Robust to Data Uncertainties*

Tareq Y. Al-Naffouri† and Ali H. Sayed‡

Abstract

This paper considers the problem of adaptive filtering in the presence of uncertainties in the regression data. A recursive procedure is derived that is based on solving local optimization problems that attempt to alleviate the worst-case effect of data uncertainties on filter performance. The resulting procedure turns out to have similarities with leakage-based adaptive filters.

1 Introduction

In real applications there are always discrepancies between an assumed model and the actual data. Such discrepancies or data uncertainties can be due to several factors such as unmodeled dynamics, disturbances, simplifications during the modeling phase, or even finite precision effects. In this paper, we focus on uncertainties in the regression data in the context of adaptive filtering. In particular, we suggest a procedure that attempts to alleviate the worst-case effect of the uncertainties on filter performance. The resulting algorithm includes leakage and has connections with leaky-LMS, as we shall explain later.

Thus consider a reference sequence $\{d(k)\}$ that arises from a linear model of the form

$$d(k) = \boldsymbol{x}_k^T \boldsymbol{w}_o + v(k), \tag{1}$$

where $v(k)$ accounts for measurement noise, $\boldsymbol{x}_k$ is a regression vector (taken as a column vector), and $\boldsymbol{w}_o$ is an unknown column vector of size M that we wish to estimate. Adaptive algorithms for estimating $\boldsymbol{w}_o$ from the data $\{\boldsymbol{x}_k, d(k)\}$ rely on recursive procedures that attempt to match the output of the adaptive filter to the reference signal $\{d(k)\}$ in some optimal manner.

*This work was supported in part by a grant from the National Science Foundation under Award number ECS-9820765.

†T. Y. Al-Naffouri is with the Department of Electrical Engineering, Stanford University, Stanford, CA 94305. E-mail: nafouri@stanford.edu. URL: www.stanford.edu/~nafouri

‡A. H. Sayed is with the Electrical Engineering Department, University of California, Los Angeles, CA 90095. E-mail: sayed@ee.ucla.edu. URL: www.ee.ucla.edu/asl

In this work, we examine the effect of one source of uncertainty that can interfere with the performance of an adaptive filter. It pertains to the fact that at any time instant, k, the regression vector $\boldsymbol{x}_k$ may not be known perfectly but only to within a certain uncertainty. In order to indicate this fact explicitly, we replace the linear model (1) by the alternative expression

$$d(k) = (\boldsymbol{x}_k + \delta\boldsymbol{x}_k)^T \boldsymbol{w}_o + v(k) \tag{2}$$

where $\delta\boldsymbol{x}_k$ denotes the uncertainty in $\boldsymbol{x}_k$. In this context, $\boldsymbol{x}_k$ represents the *nominal* value of the actual regression vector and $\delta\boldsymbol{x}_k$ represents the unknown disturbance to $\boldsymbol{x}_k$.

Motivated by the formulation in [1] for design problems with uncertainties (see also [2]–[4] for related applications), we consider two models for the uncertainty $\{\delta\boldsymbol{x}_k\}$:

1. Uncertainty in factored form. In this first model we assume that the uncertainty lies along a certain direction, i.e., it satisfies a relation of the form

$$\delta\boldsymbol{x}_k \;=\; a\boldsymbol{E}_a \quad \text{for all } k, \tag{3}$$

   where $\boldsymbol{E}_a$ is a known column vector and a is an unknown arbitrary scalar bounded by unity, $|a| \leq 1$. Different choices for $\boldsymbol{E}_a$ would correspond to different modeling assumptions on $\delta\boldsymbol{x}_k$. For example, the choice $\boldsymbol{E}_a = \text{col}\{\eta, \eta, \ldots, \eta\}$ means that each entry of $\delta\boldsymbol{x}_k$ can be regarded as bounded by η, which is useful to model finite precision effects on the data. Similarly, the choice $\boldsymbol{E}_a = \text{col}\{\eta, 0, \ldots, 0\}$, or similar sparse vectors $\boldsymbol{E}_a$, allow us to model the effect of impulsive disturbances. Also, a column vector $\boldsymbol{E}_a$ with distinct entries allows us to handle different levels of uncertainties in the various entries of $\boldsymbol{x}_k$.

2. Bounded uncertainty. In this second model we assume that the uncertainty is simply bounded, say

$$\|\delta\boldsymbol{x}_k\| \leq \eta, \tag{4}$$

   for some known scalar η and for all k. Here the notation $\|\cdot\|$ denotes the Euclidean norm of its argument.

2 Adaptive Filter Design

Given the uncertain model (2), we can proceed to formulate an optimization problem that seeks to estimate the weight vector $\boldsymbol{w}_o$ in some robust manner. By a robust procedure we mean one that attempts to alleviate the effect of the uncertainties on filter performance, especially under worst-case conditions.

Thus assume that we already have a weight-vector estimate at time k, say $\boldsymbol{w}_k$, and let us seek an estimate for time $k+1$, say $\boldsymbol{w}_{k+1}$. We shall obtain $\boldsymbol{w}_{k+1}$ from $\boldsymbol{w}_k$ by solving the following optimization (min-max) problem:

$$\min_{\boldsymbol{w}_{k+1}} \max_{\delta\boldsymbol{x}_k} J(\boldsymbol{w}_{k+1}) \tag{5}$$

where the cost function $J(\boldsymbol{w}_{k+1})$ is quadratic in $\boldsymbol{w}_{k+1}$ and is defined by

$$J(\boldsymbol{w}_{k+1}) = |\; d(k) \;-\; (\boldsymbol{x}_k + \delta\boldsymbol{x}_k)^T \boldsymbol{w}_{k+1} \;|^2 \;+\; \alpha\|\boldsymbol{w}_{k+1} - \boldsymbol{w}_k\|^2, \tag{6}$$

and α is a positive regularization parameter. The maximization in (5) is carried out over any assumed model for the uncertainty $\{\delta \boldsymbol{x}_k\}$, such as the ones indicated by (3) or (4).

The choice of the cost function (6) is motivated by the fact that in the absence of uncertainties, it is known that the classical normalized LMS algorithm can be motivated by regarding it as the solution to a regularized least-squares problem of this kind (see, e.g., [6]). Here, instead of simply minimizing a quadratic cost function, we are minimizing its worst-possible residual in the presence of the uncertainties (which corresponds to solving a min-max problem). To solve this problem we rely on the least-squares estimation theory with uncertain data developed in [1] — see also [2]–[4] and [5] for related discussions and applications.

To proceed, we introduce the quantities,

$$\begin{aligned} \boldsymbol{w} &\triangleq \boldsymbol{w}_{k+1} - \boldsymbol{w}_k \\ e(k) &\triangleq d(k) - \boldsymbol{x}_k^T \boldsymbol{w}_k, \end{aligned}$$

which are defined in terms of the given data $\{\boldsymbol{x}_k, d(k)\}$ and the weight estimates $\{\boldsymbol{w}_k, \boldsymbol{w}_{k+1}\}$, as well as the following quantity, which incorporates the effect of the uncertainty $\delta \boldsymbol{x}_k$:

$$u(k) \triangleq \delta \boldsymbol{x}_k^T (\boldsymbol{w} + \boldsymbol{w}_k). \tag{7}$$

It then follows that we can rewrite the expression for $J(\boldsymbol{w}_{k+1})$ as a function of $\boldsymbol{w}$,

$$J(\boldsymbol{w}) \;=\; |\boldsymbol{x}_k^T \boldsymbol{w} - e(k) + u(k)|^2 \;+\; \alpha \|\boldsymbol{w}\|^2, \tag{8}$$

where $u(k)$ can be interpreted as representing the aggregate model for the uncertainties.

Using (3), we can replace (5) by the optimization problem

$$\min_{\boldsymbol{w}} \max_{|u(k)| \leq \phi(\boldsymbol{w})} J(\boldsymbol{w}), \tag{9}$$

where the function $\phi(\boldsymbol{w})$ is defined by

$$\phi(\boldsymbol{w}) = \|\boldsymbol{E}_a^T (\boldsymbol{w} + \boldsymbol{w}_k)\|, \tag{10}$$

and $\{\boldsymbol{w}_k, \boldsymbol{E}_a\}$ are given. Using (4) instead, we obtain the same optimization problem (9) with the following alternative expression for $\phi(\boldsymbol{w})$:

$$\phi(\boldsymbol{w}) = \eta \|\boldsymbol{w} + \boldsymbol{w}_k\|. \tag{11}$$

Observe that we can regard this second form for $\phi(\boldsymbol{w})$ as a special case of (10) by assuming $\boldsymbol{E}_a = \eta I$. This observation allows us to treat both cases (10) and (11) simultaneously in the sequel: in one case $\boldsymbol{E}_a$ is a column vector and in the other a multiple of the identity matrix.

3 Solving the Optimization Problem

We solve the optimization problem (9) by using the solution method of [1]. To begin with, we rewrite the min-max problem (9) in the following equivalent min-min form:

$$\min_{\lambda \geq 1} \min_{\boldsymbol{w}} \left[C(\boldsymbol{w}, \lambda) + \alpha \|\boldsymbol{w}\|^2 \right], \tag{12}$$

where the two-variable function $C(\boldsymbol{w},\lambda)$ is given by

$$C(\boldsymbol{w},\lambda)=\left(1+(\lambda-1)^{\dagger}\right)\left(\boldsymbol{x}_k^T\boldsymbol{w}-e(k)\right)^2+\lambda\phi^2(\boldsymbol{w}), \tag{13}$$

and

$$(\lambda-1)^{\dagger}=\begin{cases}0 & \text{if } \lambda=1\\ \frac{1}{\lambda-1} & \text{otherwise.}\end{cases}$$

The parameter λ is a positive scalar.

We can now search for the minimum over $\boldsymbol{w}$ of (12) by setting the derivative of the cost function, $C(\boldsymbol{w},\lambda)+\alpha\|\boldsymbol{w}\|^2$, with respect to $\boldsymbol{w}$ equal to zero. This yields the equation

$$\left(\alpha I+\left(1+(\lambda-1)^{\dagger}\right)\boldsymbol{x}_k\boldsymbol{x}_k^T\right)\boldsymbol{w}+\frac{1}{2}\lambda\nabla\phi^2(\boldsymbol{w})=\left(1+(\lambda-1)^{\dagger}\right)\boldsymbol{x}_k e(k), \tag{14}$$

where the notation $\nabla\phi^2(\boldsymbol{w})$ denotes the gradient of $\phi^2(\boldsymbol{w})$ with respect to $\boldsymbol{w}$. We should stress that the optimum (and unique) $\boldsymbol{w}^o$ that results from solving (14) is a function of λ and, accordingly, should be explicitly written as $\boldsymbol{w}^o(\lambda)$. The optimum $\boldsymbol{w}^o$ over the permissible range of λ $(\lambda\geq 1)$ can then be obtained by solving the (one-dimensional) optimization problem:

$$\min_{\lambda\geq 1} G(\lambda) \tag{15}$$

where $G(\lambda)$ is defined by

$$G(\lambda)=C(\boldsymbol{w}^o(\lambda),\lambda)+\alpha\|\boldsymbol{w}^o(\lambda)\|^2. \tag{16}$$

We now invoke the special form of $\phi(\boldsymbol{w})$ to carry out the above calculations. As indicated earlier, $\phi(\boldsymbol{w})$ has the general form:

$$\phi(\boldsymbol{w})=\|\boldsymbol{E}_a^T(\boldsymbol{w}+\boldsymbol{w}_k)\|, \tag{17}$$

where $\boldsymbol{E}_a$ is either a column vector or $\boldsymbol{E}_a=\eta I$. The following calculations are valid regardless of what form $\boldsymbol{E}_a$ takes. In particular, we have

$$\nabla\phi^2(\boldsymbol{w}) \;=\; 2\boldsymbol{E}_a\boldsymbol{E}_a^T(\boldsymbol{w}+\boldsymbol{w}_k). \tag{18}$$

Upon substituting (18) into (14), and solving for $\boldsymbol{w}^o(\lambda)$, we obtain

$$\boldsymbol{w}^o(\lambda)=\left[\alpha I+\left(1+(\lambda-1)^{\dagger}\right)\boldsymbol{x}_k\boldsymbol{x}_k^T+\lambda\boldsymbol{E}_a\boldsymbol{E}_a^T\right]^{-1}\left(\left(1+(\lambda-1)^{\dagger}\right)\boldsymbol{x}_k e(k)-\lambda\boldsymbol{E}_a\boldsymbol{E}_a^T\boldsymbol{w}_k\right) \tag{19}$$

Notice that no use has yet been made of the special structure of our problem. In particular, notice that the only matrix that appears in (19) is the identity, and every other parameter is otherwise a vector or a scalar. This fact can be exploited to get a more explicit expression for $\boldsymbol{w}^o(\lambda)$ since we can invoke the matrix inversion lemma to evaluate the inverse in (19). As such, at this point, we have to treat the two classes of uncertainties we mentioned before separately.

4 The Case of Bounded Uncertainties

For bounded uncertainties (where we regard $\boldsymbol{E}_a = \eta I$), we find by the matrix inversion lemma that

$$\boldsymbol{w}^o(\lambda) = -\gamma \boldsymbol{w}_k + \lambda \frac{\gamma \boldsymbol{x}_k^T \boldsymbol{w}_k + e(k)}{\beta(\lambda - 1) + \lambda \|\boldsymbol{x}_k\|^2} \boldsymbol{x}_k \tag{20}$$

where

$$\beta \triangleq \alpha + \eta^2 \lambda \quad \text{and} \quad \gamma \triangleq \eta^2 \lambda / \beta.$$

Or, since $\boldsymbol{w}^o(\lambda) = \boldsymbol{w}_{k+1} - \boldsymbol{w}_k$ and $e(k) = d(k) - \boldsymbol{x}_k^T \boldsymbol{w}_k$, we get

$$\begin{aligned} \boldsymbol{w}_{k+1} &= (1-\gamma)\boldsymbol{w}_k + \lambda \frac{d(k) - (1-\gamma)\boldsymbol{x}_k^T \boldsymbol{w}_k}{\beta(\lambda - 1) + \lambda \|\boldsymbol{x}_k\|^2} \, \boldsymbol{x}_k & (21) \\ &= (1-\gamma)\boldsymbol{w}_k + \lambda \frac{e^w(k)}{\beta(\lambda - 1) + \lambda \|\boldsymbol{x}_k\|^2} \, \boldsymbol{x}_k, & (22) \end{aligned}$$

where the modified error $e^w(k)$ is defined by

$$e^w(k) \triangleq d(k) - (1-\gamma)\boldsymbol{x}_k^T \boldsymbol{w}_k. \tag{23}$$

Four remarks are in place here:

1. Equation (22) gives the update equation for $\boldsymbol{w}_{k+1}$ in terms of a parameter λ, whose optimal value we still need to determine. Note that the values of $\{\beta, \gamma\}$ also depend on λ. As pointed out earlier (see (15)-(16)), the optimum λ is obtained by minimizing the function $G(\lambda)$ which in the present case is given by

$$G(\lambda) = \begin{cases} \frac{1}{\alpha+\eta^2} \left(\frac{\left(-\alpha e(k) - \eta^2 d(k)\right)^2}{\alpha + \eta^2 + \|\boldsymbol{x}_k\|^2} + \alpha \eta^2 \|\boldsymbol{w}_k\|^2 \right) & \text{if } \lambda = 1 \\ \frac{\lambda}{\beta} \left[\frac{\left(\alpha \boldsymbol{x}_k^T \boldsymbol{w}_k - \beta d(k)\right)^2}{\beta(\lambda-1) + \lambda \|\boldsymbol{x}_k\|^2} + \alpha \eta^2 \|\boldsymbol{w}_k\|^2 \right] & \text{otherwise.} \end{cases} \tag{24}$$

2. Minimizing $G(\lambda)$ requires that we find the roots of its derivative, $dG(\lambda)/d\lambda$, or, equivalently, the roots of a fourth-order polynomial. While this could be done numerically and in closed form as well, we are more interested in deriving a simplified algorithm. For this purpose, we have observed through repeated simulations that the minimum value of the function $G(\lambda)$ generally occurs at a value that is very close to 1. In this case we can simplify the update recursions to:

$$\boldsymbol{w}_{k+1} = (1-\gamma)\boldsymbol{w}_k \; + \; \frac{d(k) - (1-\gamma)\boldsymbol{x}_k^T \boldsymbol{w}_k}{\alpha + \eta^2 + \|\boldsymbol{x}_k\|^2} \, \boldsymbol{x}_k, \tag{25}$$

   where

$$\gamma = \frac{\eta^2}{\alpha + \eta^2} \tag{26}$$

   Table 1 summarizes the adaptive algorithm for the bounded uncertainty case, where we have also incorporated a step-size parameter μ.

Table 1: *Adaptive algorithm for bounded regressor uncertainty.*

DESCRIPTION	EQUATION
Adaptation Equation	$\boldsymbol{w}_{k+1} = (1-\gamma)\boldsymbol{w}_k \; + \; \mu \; \frac{d(k)-(1-\gamma)\boldsymbol{x}_k^T\boldsymbol{w}_k}{\alpha+\eta^2+\|\boldsymbol{x}_k\|^2} \; \boldsymbol{x}_k$
Filtering Equation	$e^w(k) = d(k) - (1-\gamma)\boldsymbol{x}_k^T\boldsymbol{w}_k$
Leakage Parameter	$\gamma = \frac{\eta^2}{\alpha+\eta^2}$
Step-size	μ
Apriori Information	α (regularization parameter) η (uncertainty parameter)

3. By inspecting (25) we notice that it has a form similar to leaky-LMS, except that the leakage factor $(1-\gamma)$ is applied to both terms on the right-hand side. We shall comment further on this fact below.

4. In our development, we could have obtained similar (and more sophisticated) results with an additional uncertainty model on the additive noise $v(k)$ (see (2)). In particular, if we assume that $|v(k)| \leq \eta_v$ for some η_v, then the function $\phi(\boldsymbol{w})$ would have taken the form (compare with (17)):
$$\phi(\boldsymbol{w}) = \|\boldsymbol{E}_a^T(\boldsymbol{w}+\boldsymbol{w}_k)\| + \eta_v. \tag{27}$$
The adaptation equation we arrived at (namely (25)) remains essentially the same.

5 The Case of Factored Uncertainties

Consider (19) again for the case of factored uncertainties. Building on our observation in the second remark above, we set λ in (19) to 1 to get

$$\boldsymbol{w}^o(1) = \left[\alpha I + \boldsymbol{x}_k\boldsymbol{x}_k^T + \boldsymbol{E}_a\boldsymbol{E}_a^T\right]^{-1}\left(e(k) - \boldsymbol{E}_a\boldsymbol{E}_a^T\boldsymbol{w}_k\right). \tag{28}$$

We need to employ the matrix inversion lemma twice this time to evaluate the inverse in (28). We find after tedious but straightforward manipulations that the adaptation equation is given this time by

$$\boldsymbol{w}_{k+1} = \left\{I - \frac{1}{\Delta}\left[\|\boldsymbol{x}_k\|^2\left(I - \frac{\boldsymbol{x}_k\boldsymbol{x}_k^T}{\|\boldsymbol{x}_k\|^2}\right) + \alpha I\right]\boldsymbol{E}_a\boldsymbol{E}_a^T\right\}\boldsymbol{w}_k + \frac{1}{\Delta}\left[\|\boldsymbol{E}_a\|^2\left(I - \frac{\boldsymbol{E}_a\boldsymbol{E}_a^T}{\|\boldsymbol{E}_a\|^2}\right) + \alpha I\right]e(k)\boldsymbol{x}_k, \tag{29}$$

where
$$\Delta \triangleq \left(\alpha + \|\boldsymbol{x}_k\|^2\right)\left(\alpha + \|\boldsymbol{E}_a\|^2\right) - \left(\boldsymbol{E}_a^T\boldsymbol{x}_k\right)^2 \tag{30}$$

6 A Comment on Leakage in Adaptive Filtering

Leakage in LMS is usually employed when the input is noisy or not persistently exciting since it helps avoid the drift problem of LMS. In this case, the adaptation equation is of the form

$$\boldsymbol{w}_{k+1} = (1-\mu\alpha)\boldsymbol{w}_k + \mu\boldsymbol{x}_k e(k), \tag{31}$$

for some α, with $e(k) = d(k) - \boldsymbol{x}_k^T\boldsymbol{w}_k$ and $d(k) = \boldsymbol{x}_k^T\boldsymbol{w}_o + v(k)$.[1] The leakage factor $(1-\mu\alpha)$ applies to the factor $\boldsymbol{w}_k$ in (31) only but not to the $\boldsymbol{w}_k$ that appears in the output error $e(k)$. This distinction relates to the bias problem that is characteristic of leaky-LMS (see also [6, 7] and [8]).

To examine this point further, let us investigate the bias problem when we introduce a leakage factor in both terms of the update equation (31), i.e., let us consider a recursion of the form

$$\boldsymbol{w}_{k+1} = (1-\mu\alpha)\boldsymbol{w}_k + \mu\boldsymbol{x}_k e^w(k), \tag{32}$$

where

$$e^w(k) = d(k) - (1-\alpha)\boldsymbol{x}_k^T\boldsymbol{w}_k. \tag{33}$$

It is easy to verify that this modified algorithm leads to unbiased weight vector estimates for i.i.d. input signals. To this end, we rewrite (32) in terms of the weight error vector, $\tilde{\boldsymbol{w}}_k = \boldsymbol{w}_k - \boldsymbol{w}_o$, by subtracting $\boldsymbol{w}_o$ from both sides of (32):

$$\begin{aligned} \tilde{\boldsymbol{w}}_{k+1} &= (1-\mu\alpha)\tilde{\boldsymbol{w}}_k - \mu\alpha w_o + \mu\boldsymbol{x}_k e^w(k) \\ &= (1-\mu\alpha)\tilde{\boldsymbol{w}}_k + \mu\left(\boldsymbol{x}_k e^w(k) - \alpha\boldsymbol{w}_o\right). \end{aligned} \tag{34}$$

Now the error $e^w(k)$ too can be written in terms of the weight error vector:

$$\begin{aligned} e^w(k) &= d(k) - (1-\alpha)\boldsymbol{x}_k^T\boldsymbol{w}_k & (35) \\ &= \alpha\boldsymbol{x}_k^T\boldsymbol{w}_o - (1-\alpha)\boldsymbol{x}_k^T\tilde{\boldsymbol{w}}_k + v(k), & (36) \end{aligned}$$

so that (34) becomes

$$\tilde{\boldsymbol{w}}_{k+1} = \left(1-\mu\alpha-\mu(1-\alpha)\boldsymbol{x}_k\boldsymbol{x}_k^T\right)\tilde{\boldsymbol{w}}_k + \mu\alpha(\boldsymbol{x}_k\boldsymbol{x}_k^T - I)\boldsymbol{w}_o + \mu\boldsymbol{x}_k v(k). \tag{37}$$

Assuming the input to be i.i.d., of unit variance (i.e., $E\boldsymbol{x}_k\boldsymbol{x}_k^T = I$), and independent of the noise, we get upon averaging the last equation that

$$E[\tilde{\boldsymbol{w}}_{k+1}] = \left(1-\mu\alpha-\mu(1-\alpha)\right)E[\tilde{\boldsymbol{w}}_k],$$

a homogeneous recursion in $E[\tilde{\boldsymbol{w}}_k]$. Thus, for a small enough step-size, $E[\tilde{\boldsymbol{w}}_k]$ will converge to zero and $\boldsymbol{w}_k$ will be an unbiased estimate of $\boldsymbol{w}_o$.

But what if the input signal is correlated? We can treat the general input case by introducing the "whitening" factor $1/\|\boldsymbol{x}_k\|^2$ in the adaptation equation (32), which then becomes

$$\boldsymbol{w}_{k+1} = (1-\mu\alpha)\boldsymbol{w}_k + \mu\boldsymbol{x}_k\,\frac{e^w(k)}{\|\boldsymbol{x}_k\|^2}. \tag{38}$$

[1] In this section we are ignoring uncertainties in the regression vectors.

This "whitening" factor is usually introduced in the LMS algorithm to improve its convergence for a spectrally correlated input. When introduced for the leaky-LMS version (32), this factor serves the additional purpose of helping "ensure" unbiasedness for a correlated input. This gives rise to a hybrid algorithm that is composed of leaky-LMS and NLMS. The question is whether it is possible to justify these modifications and constructions in a more "rigorous" manner.

As can be seen from the discussions in the earlier sections, it turns out that these modifications are similar to what we get when we derive an adaptive algorithm in the presence of uncertainties in the regression data.

7 Simulation Results

In order to demonstrate some of the points raised in the previous section, we compared the performance of the algorithm of Table 1 with the following related leaky-algorithms:

1. The leaky-LMS algorithm.

$$\boldsymbol{w}_{k+1} = (1 - \mu\alpha)\boldsymbol{w}_k + \mu e(k)\boldsymbol{x}_k$$

2. The leaky-NLMS algorithm.

$$\boldsymbol{w}_{k+1} = (1 - \mu\alpha)\boldsymbol{w}_k + \mu \frac{e(k)}{\beta + \|\boldsymbol{x}_k\|^2}\boldsymbol{x}_k$$

In both cases, we have $e(k) = d(k) - \boldsymbol{x}_k^T \boldsymbol{w}_k$.

To ensure a fair comparison, we chose the step size so that all three algorithms would have the same convergence speed. For the same reason, the three algorithms also shared the same leakage factor (i.e., the factor multiplying $\boldsymbol{w}_k$ on the right-hand side). The algorithms were employed in identifying a four tap FIR filter. Both the nominal input and the associated uncertainty were chosen to be uniformly distributed such that

$$\frac{\max \|\delta \boldsymbol{x}_k\|}{\max \|\boldsymbol{x}_k\|} = \frac{0.4}{4} = 10\%.$$

This means that $\eta = .4$ and $\gamma = .096$. The experiment was repeated 10,000 times and the resulting learning curves are shown in Fig. 1. The top curve refers to leaky-LMS, the middle curve to leaky-NLMS, and the bottom curve to the robust-NLMS version of Table 1.

8 Conclusion

In this paper we derived an adaptive algorithm that attempts to alleviate the worst-case of the uncertainties in the regression data on filter performance. Such situations can arise, for example, in channel tracking applications for multicarrier modulation. The resulting algorithm turns out to include leakage, albeit in a manner that applies both to the weight vector $\boldsymbol{w}_k$ and to the estimation error $e^w(k)$, as shown in Table 1.

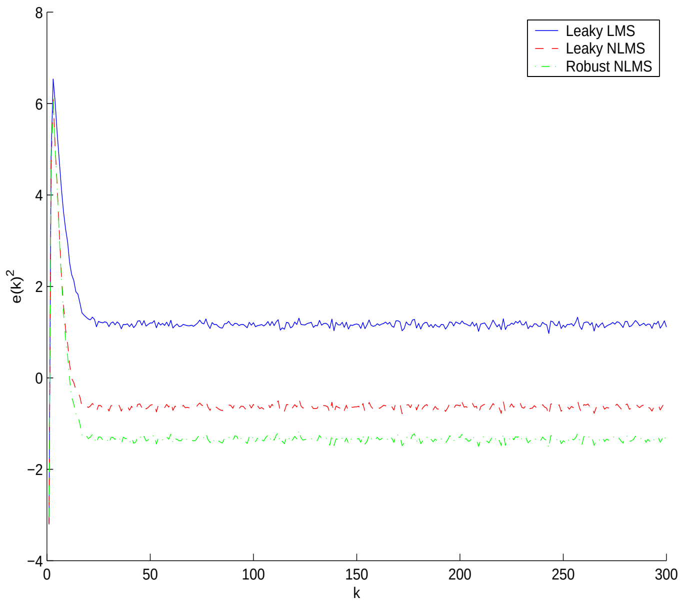

Figure 1: *Learning curves for leaky*-LMS*, leaky*-NLMS*, and the robust version of Table 1.*

References

[1] A. H. Sayed and V. H. Nascimento, "Design criteria for uncertain models with structured and unstructured uncertainties," in *Robustness in Identification and Control*, A. Garulli, A. Tesi, and A. Vicino, editors, vol. 245 of Lecture Notes in Control and Information Sciences, pp. 159–173, Springer Verlag, London, 1999.

[2] A. H. Sayed, V. H. Nascimento, and S. Chandrasekaran, "Estimation and control with bounded data uncertainties," *Linear Algebra and Its Applications*, vol. 284, pp. 259–306, Nov. 1998.

[3] A. H. Sayed and S. Chandrasekaran, "Parameter estimation with multiple sources and levels of uncertainties," *IEEE Transactions on Signal Processing*, vol. 48, no. 3, pp. 680–692, Mar. 2000.

[4] A. H. Sayed, T. Y. Al-Naffouri, and T. Kailath, "Robust estimation for uncertain models in a data fusion scenario," *Proc. IFAC System Identification Symposium,* Santa Barbara, CA, June 2000.

[5] L. E. Ghaoui and H. Hebret, "Robust solutions to least-squares problems with uncertain data," *SIAM J. Matrix Anal. Appl.*, vol. 18, pp. 1035–1064, 1997.

[6] S. Haykin, *Adaptive Filter Theory*, Prentice Hall, 3rd edition, NJ, 1996.

[7] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*, Prentice Hall, NJ, 1985.

[8] V. H. Nascimento and A. H. Sayed, "Unbiased and stable leakage-based adaptive filters," *IEEE Transactions on Signal Processing*, vol. 47, no. 12, pp. 3261–3276, Dec. 1999.