Unbiased and Stable Leakage-Based Adaptive Filters

Vítor H. Nascimento and Ali H. Sayed, *Senior Member, IEEE*

***Abstract*—The paper develops a leakage-based adaptive algorithm, refered to as circular-leaky, which in addition to solving the drift problem of the classical least mean squares (LMS) adaptive algorithm, it also avoids the bias problem that is created by the standard leaky LMS solution. These two desirable properties of unbiased and bounded estimates are guaranteed by circular leaky at essentially the same computational cost as LMS. The derivation in the paper relies on results from averaging theory and from Lyapunov stability theory, and the analysis shows that the above properties hold not only in infinite-precision but also in finite-precision arithmetic. The paper further extends the results to a so-called switching-σ algorithm, which is a leakage-based solution used in adaptive control.**

***Index Terms*—Adaptive algorithm, averaging theory, bias, finite precision, leakage, Lyapunov stability, stability.**

I. Introduction

The LEAKY least-mean-squares (leaky LMS) algorithm is a widely used adaptive scheme, having been employed in applications such as fractionally spaced equalizers (FSE's) [1], speech digitization for telephony [2], prevention of bursting in adaptive echo cancelation and auto regressive moving average (ARMA) predictors [3], adaptive control [3], [4], and antenna arrays [5], among others. The algorithm was originally proposed to stabilize the weight-drift problem that occurs when the standard LMS algorithm is used in environments that do not satisfy a certain persistence of excitation (PE) condition. Unfortunately, however, the solution provided by leaky LMS comes at a price [6, p. 746]. There is both an increase in the computational/hardware cost when compared with the conventional LMS algorithm, and there is a degradation in performance due to the introduction of bias to the weight estimates. The drift and bias problems, and their implications, are discussed in the references cited above, as well as in [7]–[11] and in Section III.

The purpose of this paper is to address the two issues of drift and bias, by proposing a variant to the LMS algorithm that we refer to as the *circular-leaky* LMS *algorithm*. Under some conditions that are described in this paper, this algorithm

Manuscript received December 23, 1996; revised April 20, 1999. This work was supported in part by the National Science Foundation under Grants MIP-9796147 and CCR-9732376. The work of V. H. Nascimento was supported by a fellowship from Conselho Nacional de Pesquisa e Desenvolvimento-CNPq-Brazil, while on leave from Escola Politécnica da Universidade de São Paulo. The associate editor coordinating the review of this paper and approving it for publication was Dr. Sergios Theodoridis.

V. H. Nascimento was with the Department of Electrical Engineering, University of California, Los Angeles, CA 90095 USA. He is now with the Department of Electrical Engineering, Universidade de São Paulo, São Paulo, Brazil.

A. H. Sayed is with the Department of Electrical Engineering, University of California, Los Angeles, CA 90095 USA.

Publisher Item Identifier S 1053-587X(99)09210-7.

TABLE I
Comparison of the Various Adaptive Algorithms. The Matrix R_k in the Third Column is the Autocorrelation Matrix of the Input Sequence x_k

Algorithm	Drift Problem	Biased when $R_k > 0$	Complexity MA	M	A	IF
LMS	**YES**	NO	$2M$	1	0	0
Leaky LMS	NO	**YES**	$2M$	$\mathbf{M+1}$	0	0
Switching-σ	NO	NO	$\mathbf{3M}$	$\mathbf{M+2}$	**2**	**3**
Circular Leaky	**NO**	**NO**	$\mathbf{2M}$	**3**	**2**	**3**

solves the weight drift problem without introducing bias to the estimates and at essentially the same computational cost as conventional LMS. These facts are established by relying on results from averaging theory and Lyapunov stability theory. We use averaging theory to show that circular leaky does not lead to biased estimates and then employ a deterministic stability analysis to show that the algorithm avoids unbounded growth of the weight estimates. In fact, we establish stronger results by showing that these properties hold even in the presence of finite-precision effects in fixed-point implementations.

The results of the paper are further extended to a modified version of the so-called *switching-σ* algorithm, which is studied in the adaptive control literature [12], [13]. This algorithm also provides unbiased weight estimates, but it has a computational cost higher than that of leaky LMS. While the literature currently available for the switching-σ algorithm provides only deterministic analyses for infinite-precision arithmetic, our analysis will provide both stochastic and deterministic results for the finite-precision case as well. Table I summarizes the properties of the four different algorithms mentioned above. In the complexity column, we list approximate values for the number of multiplications (M), additions (A), multiply-and-accumulate (MA), and if-then (IF) commands necessary for each algorithm.

The paper is organized as follows. In the next section, we describe the algorithms studied here, giving the infinite-precision models and update-laws (while the fixed-point update equations are delayed to Appendix A). To further motivate our results, we also present a few examples showing both the drift problem of LMS and the bias introduced by leaky LMS in Section III. Results from averaging theory are briefly reviewed in Section IV-A and then used to study both circular leaky and switching-σ. Our main results are in Sections IV and V, which describe, respectively, the performance and stability properties of the algorithms. Simulations showing the advantages of circular-leaky are provided in Section VI.

II. Models and Algorithm Definitions

The adaptive problem we are concerned with is the following. Given noisy scalar measurements $\{y(k) \in \mathbb{R}\}_{k=0}^{\infty}$ that satisfy a linear model of the form

$$y(k) = \boldsymbol{x}_k^T \boldsymbol{w}_* + v(k) \tag{1}$$

we want to estimate the unknown constant vector $\boldsymbol{w}_* \in \mathbb{R}^M$. Here, the $\{\boldsymbol{x}_k \in \mathbb{R}^M\}$ are known regressor vectors, whereas the $\{v(k) \in \mathbb{R}\}$ is an unknown disturbance (noise) sequence. We define the *output error* for a given estimate $\boldsymbol{w}_k$ of $\boldsymbol{w}_*$ at time k as $e(k) = y(k) - \boldsymbol{x}_k^T \boldsymbol{w}_k$ and the *weight error vector* as the difference $\tilde{\boldsymbol{w}}_k = \boldsymbol{w}_* - \boldsymbol{w}_k$. We further introduce the input covariance matrix $R_k = \mathrm{E}\boldsymbol{x}_k \boldsymbol{x}_k^T$ in addition to the upper bounds

$$\sup_{k \geq 0} \|\boldsymbol{x}_k\|^2 = \beta, \quad \sup_{k \geq 0} |v(k)| = v_{\max} \tag{2}$$

where β and $v_{\max}$ are finite positive constants. The requirement of bounded $\{\boldsymbol{x}_k, v(k)\}$ is actually a standard one in the literature whenever finite-precision arithmetic effects are being studied, although it is often implicit in the assumptions. For example, the assumption that all variables are suitably scaled so that overflow never occurs in fact requires that all variables be bounded (see [14]–[16]).

There are several algorithms that can be used to compute estimates $\boldsymbol{w}_k$ for $\boldsymbol{w}_*$. In this paper, we focus on the following adaptive schemes of the LMS class.

LMS: In the standard LMS algorithm, the estimates $\boldsymbol{w}_k$ are computed via [17], [18]

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k + \mu \boldsymbol{x}_k e(k), \quad \text{with initial condition } \boldsymbol{w}_0. \tag{3}$$

Leaky LMS: In order to prevent unbounded growth of the weight estimates in LMS (see Section III), this algorithm incorporates a positive leakage factor $\mu\alpha_0$ to the adder in (3)

$$\boldsymbol{w}_{k+1}^l = (1 - \mu\alpha_0)\boldsymbol{w}_k^l + \mu \boldsymbol{x}_k e^l(k) \tag{4}$$

where we use the symbol $e^l(k)$ to indicate that the output error is computed using the leaky LMS estimate $\boldsymbol{w}_k^l$.

Circular-Leaky: We denote the weight vector estimate for this algorithm by $\boldsymbol{w}_k^c$, and let $\{w_{k,j}^c\}$ denote its individual entries (for $j = 1, \cdots, M$). There are three modifications with respect to leaky LMS in this new variant. First, leakage is applied to a *single* tap at each iteration. Second, leakage is applied *only* if the tap magnitude exceeds a prespecified level, say, C_1, and finally, the value of the leakage factor α_c is *dependent* on the magnitude of the tap (and therefore changes with time as well).

Before exhibiting the mathematical description of the algorithm, let us first explain its operation schematically for ease of presentation. Thus assume that $M = 3$, i.e., assume that we are dealing with weight vector estimates that are three taps long. The diagram below shows the proposed procedure for the first five iterations of the algorithm, where the arrows indicate the entries that are checked for *possible* application of leakage at each iteration:

$$\begin{array}{ccccc} \boldsymbol{w}_0^c & \boldsymbol{w}_1^c & \boldsymbol{w}_2^c & \boldsymbol{w}_3^c & \boldsymbol{w}_4^c \\ \rightarrow \times & \times & \times & \rightarrow \times & \times \\ \times & \rightarrow \times & \times & \times & \rightarrow \times \\ \times & \times & \rightarrow \times & \times & \times \end{array}$$

In words, we start by checking the top entry of $\boldsymbol{w}_0^c$ and verifying whether its magnitude exceeds or not the prespecified level C_1. If it does, then we apply leakage to it. If not, then no leakage is applied. The weight vector is then updated (as explained below) to obtain $\boldsymbol{w}_1^c$. We now repeat the procedure by checking the *second* entry of $\boldsymbol{w}_1^c$ followed by the *third* entry of $\boldsymbol{w}_2^c$. At the end of these three iterations, we return to checking the top entry of $\boldsymbol{w}_3^c$, the second entry of $\boldsymbol{w}_4^c$, and so on.

We thus see that this procedure employs a *nonlinear* and *time-variant* leakage term $\alpha_c(k, \cdot)$ instead of the constant factor α_0 in leaky LMS (4). More specifically, at an arbitrary time k, we check whether $|w_{k,\bar{k}}^c| > C_1$, where $\bar{k} = (k \bmod M)$. If the condition is true, we compute an intermediate estimate $\bar{\boldsymbol{w}}_k^c$ that is identical to $\boldsymbol{w}_k^c$ except for a leakage term that is applied to its $\bar{k}$th entry, as shown in

$$\bar{\boldsymbol{w}}_k^c = \begin{cases} \begin{bmatrix} w_{k,0}^c \\ \vdots \\ (1 - \mu\alpha_c(k, w_{k,\bar{k}}^c))w_{k,\bar{k}}^c \\ \vdots \\ w_{k,M-1}^c \end{bmatrix}, & \text{if } |w_{k,\bar{k}}^c| > C_1 \\ \boldsymbol{w}_k^c, & \text{otherwise.} \end{cases}$$

Note that at most one entry of $\boldsymbol{w}_k^c$ is modified in the computation of $\bar{\boldsymbol{w}}_k^c$ [the value of the leakage term $\alpha_c(k, w_{k,\bar{k}}^c)$ is defined later]. Once the intermediate estimate $\bar{\boldsymbol{w}}_k^c$ has been computed, we proceed with an LMS-type update, namely

$$\boldsymbol{w}_{k+1}^c = \bar{\boldsymbol{w}}_k^c + \mu \boldsymbol{x}_k e^c(k) \tag{5}$$

where $e^c(k) = y(k) - \boldsymbol{x}_k^T \boldsymbol{w}_k^c$. We can describe the algorithm more compactly as follows. Let $\boldsymbol{e}_{\bar{k}}$ denote the $\bar{k}$th basis vector (i.e., $e_{\bar{k},\bar{k}} = 1$, $e_{\bar{k},j} = 0$ for $j \neq \bar{k}$). Then, the new algorithm takes the form

$$\boldsymbol{w}_{k+1}^c = (I - \mu\alpha_c(k, w_{k,\bar{k}}^c)\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T)\boldsymbol{w}_k^c + \mu \boldsymbol{x}_k e^c(k). \tag{6}$$

The function $\alpha_c(k, \cdot)$ is defined as follows. Let α_0, C_1, and $C_2 > C_1$ be given positive constants,[1] and define $D = (C_2 - C_1)/2$. Then[2]

$$\alpha_c(k, w_{k,\bar{k}}^c) = \begin{cases} \alpha_0, & \text{if } |w_{k,\bar{k}}^c| \geq C_2 \\ \alpha_0 - \dfrac{\alpha_0}{2}\left(\dfrac{C_2 - |w_{k,\bar{k}}^c|}{D}\right)^2 & \text{if } C_1 + D \leq |w_{k,\bar{k}}^c| < C_2 \\ \dfrac{\alpha_0}{2}\left(\dfrac{|w_{k,\bar{k}}^c| - C_1}{D}\right)^2 & \text{if } C_1 < |w_{k,\bar{k}}^c| < C_1 + D \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

Fig. 1 shows a plot of the leakage function $\alpha_c(k, \cdot)$ for the choices $C_1 = 0.5$, $C_2 = 0.7$, and $\alpha_0 = 0.1$. Later in the paper [see, e.g., (18)], we show how $\{\mu, \alpha_0, C_1\}$ should be chosen.

[1] Later in Section VII, we show how these constants could be chosen.

[2] It is possible to simplify this definition and use a discontinuous $\alpha_c(k, w_{k,\bar{k}}^c)$—see Section VII as well as [19].

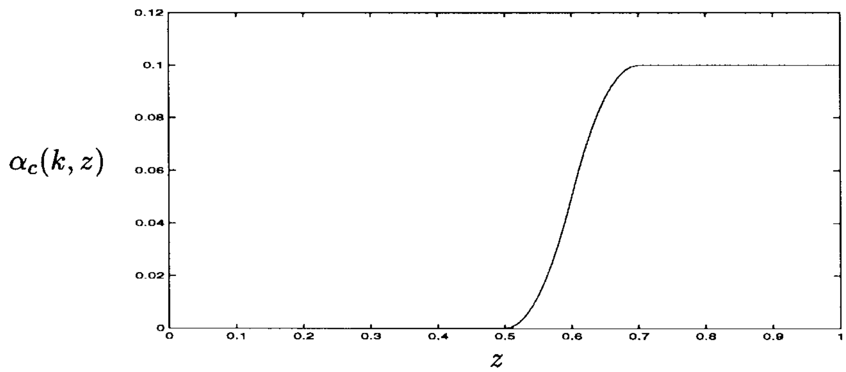


Fig. 1. Leakage function $\alpha_c(k,\cdot)$ with $C_1 = 0.5$, $C_2 = 0.7$, and $\alpha_0 = 0.1$.

TABLE II
DIFFERENCES IN THE LEAKAGE TERMS AMONG THE ALGORITHMS

Algorithm	Leakage applied if	Leakage term	Applied to
Leaky LMS	always applied	$\mu\alpha_0 \boldsymbol{w}_k^l$	all taps
Circular-leaky	$\lvert w^c_{k,\bar{k}}\rvert > C_1$	$\mu\alpha_c(w^c_{k,\bar{k}})\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T\boldsymbol{w}_k^c$	a single tap
Modified switching-σ	$\lVert\boldsymbol{w}_k^s\rVert > S_1$	$\mu\alpha_s(\boldsymbol{w}_k^s)\boldsymbol{w}_k^s$	all taps

In words, (7) shows that starting from $k = 0$, we examine the magnitude of the *top* entry of $\boldsymbol{w}_0^c$ and check in which interval it lies

$$(0, C_1], \qquad (C_1, C_1 + D), \qquad [C_1 + D, C_2), \qquad [C_2, \infty).$$

The interval tells us the value of the leakage α_c that we should apply to this tap entry. In this way, we create $\bar{\boldsymbol{w}}_0^c$ and then $\boldsymbol{w}_1^c$ via (5). Next, we examine the magnitude of the *second* entry of $\boldsymbol{w}_1^c$, determine in which interval it lies, and find the appropriate α_c. We then generate $\bar{\boldsymbol{w}}_1^c$ and $\boldsymbol{w}_2^c$ via (5). Next, we examine the magnitude of the *third* entry of $\boldsymbol{w}_2^c$ and determine α_c, $\bar{\boldsymbol{w}}_2^c$, and $\boldsymbol{w}_3^c$. We continue in this fashion by examining in each iteration k a single entry of $\boldsymbol{w}_k^c$ and by moving circularly from one entry in a weight vector to the following entry in the next weight vector as the iterations progress.

Note that the constant C_1 must satisfy $C_1 > \|\boldsymbol{w}\|_\infty$ in order to guarantee that the leakage term $\alpha_c(\cdot)$ is zero when the estimate $\boldsymbol{w}_k$ is close to $\boldsymbol{w}_*$. Hence, in the sequel, we shall assume that a bound $W_\infty \geq \|\boldsymbol{w}_*\|_\infty$ is available (see Section VII).[3]

The time dependency of α_c comes from the fact that a different entry of $\boldsymbol{w}_k^c$ is checked at each time instant. To simplify the notation, we will not explicitly indicate this time dependency in the remainder of the paper and will thus write $\alpha_c(w^c_{k,\bar{k}})$ instead of $\alpha_c(k, w^c_{k,\bar{k}})$.

Modified Switching-σ Algorithm: In this algorithm, the leakage factor is applied to all taps whenever $\|\boldsymbol{w}_k^s\|$ is too large[4]

$$\boldsymbol{w}_{k+1}^s = (1 - \mu\alpha_s(\boldsymbol{w}_k^s))\boldsymbol{w}_k^s + \mu\boldsymbol{x}_k e^s(k) \tag{8}$$

where the function $\alpha_s(\boldsymbol{w}_k^s)$ is defined as follows. Let α_0, S_1, and $S_2 > S_1$ be positive constants, and define $E = (S_2 - S_1)/2$. Then

$$\alpha_s(\boldsymbol{w}_k^s) = \begin{cases} \alpha_0, & \text{if } \|\boldsymbol{w}_k^s\| \geq S_2 \\ \alpha_0 - \dfrac{\alpha_0}{2}\left(\dfrac{S_2 - \|\boldsymbol{w}_k^s\|}{E}\right)^2 & \text{if } S_1 + E \leq \|\boldsymbol{w}_k^s\| < S_2 \\ \dfrac{\alpha_0}{2}\left(\dfrac{\|\boldsymbol{w}_k^s\| - S_1}{E}\right)^2 & \text{if } S_1 < \|\boldsymbol{w}_k^s\| < S_1 + E \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

As was the case for circular leaky, the constant S_1 is chosen so that leakage is off when $\boldsymbol{w}_k$ is close to $\boldsymbol{w}_*$ (i.e., we assume that a bound $W_2 \geq \|\boldsymbol{w}_*\|$ is available and choose $S_1 > W_2$). A variant of this algorithm is well known in the adaptive control literature [12], where the leakage function $\alpha_s(\boldsymbol{w}_k^s)$ is not smooth, as above, but has a discontinuous transition between 0 and α_0.

Table II summarizes the differences among the leakage-based algorithms.

III. THE WEIGHT DRIFT AND BIAS PROBLEMS

The fact that the LMS algorithm (3) can produce unbounded weight estimates in some situations is described in several works including, for example, [1], [4], [6], and [8]. The work [8] studies the drift problem in a deterministic infinite-precision setting, whereas finite precision effects are considered in [9] and [20]. The work [4] provides an analysis in the adaptive control context.

The use of the leaky LMS algorithm to avoid the drift problem of LMS was apparently proposed as early as 1973 [1], [2], [5]. Leaky LMS, however, introduces a bias problem that was also described and studied in these references, as well as in [10] and others.

Given that many works on the drift and bias problems of LMS and leaky LMS exist in the literature, we shall provide

[3] The notation $\|\cdot\|_\infty$ denotes the largest absolute entry of its argument.

[4] The notation $\|\cdot\|$ denotes the Euclidean norm of its vector argument or the maximum singular value of its matrix argument.

here only a brief description of these problems (in addition to a few examples) in order to better motivate the discussion in later sections and in order to highlight the problems that we address in this paper. We consider both cases of infinite precision and finite-precision arithmetic for reasons explained below.

A. The Drift Problem in Infinite Precision Arithmetic

We illustrate the drift problem of LMS as follows. Consider the following contrived (deterministic) example. Let the regressors be scalar $(M = 1)$ and given by $\boldsymbol{x}(k) = (1/\sqrt{k+1})$. In addition, assume that the step-size μ is 1, that the noise sequence is $v(k) = 10^{-4}$, and that the "true" weight vector is $\boldsymbol{w}_* = 0$. It then follows from the model (1) and from the LMS recursion (3) that

$$\boldsymbol{w}_{k+1} = \left(1 - \frac{1}{k+1}\right)\boldsymbol{w}_k + 10^{-4}\,\frac{1}{\sqrt{k+1}}.$$

Solving this time-variant linear equation, we find that for a zero initial condition $\boldsymbol{w}_0$ and for $k \geq 1$

$$\boldsymbol{w}_k = \frac{10^{-4}}{k}\sum_{i=1}^{k}\sqrt{i} \geq \frac{2\times 10^{-4}}{3}\sqrt{k}$$

which implies that $\boldsymbol{w}_k \rightarrow \infty$ as $k \rightarrow \infty$.

This example shows that the weight estimates computed by the LMS algorithm can grow slowly to very large values, even when the noise is small. Even with zero noise, unbounded growth of the estimates can happen due to finite-precision arithmetic errors (see [9], [20], and the example below). Such unbounded growth of the LMS estimates can happen if two conditions are satisfied: 1) The noise or the finite-precision arithmetic errors have nonzero mean, and 2) the covariance matrix of the input sequence $\{\boldsymbol{x}_k\}$ is not uniformly positive definite (i.e., there is no $\rho > 0$ such that $R_k > \rho I$ for all k). As shown in [1] and [20], these situations do arise in practice. For example, applications such as adaptive equalization with fractionally spaced equalizers do not have inputs with uniformly positive-definite covariance matrices.

B. The Drift Problem in Fixed-Point Arithmetic

The example in this section shows how finite-precision errors can also cause drift. For this purpose, we assume that fixed-point arithmetic is used and employ the symbol fx$[a]$ to denote the fixed-point representation of a real number a. We denote by ϵ the *machine precision* or the largest absolute difference between a real number a and its fixed-point representation, namely, $|\text{fx}[a] - a| \leq \epsilon$. For simplicity, we assume that all variables are stored with B bits plus sign and that rounding is used (this implies that $\epsilon = 2^{-B-1}$).

Finite-precision errors can result in nonzero mean variables in a number of ways. Consider, for example, a random variable a with distribution

$$a = \begin{cases} 0.5 + 2^{-7}, & \text{with probability } 0.5 - 2^{-7} \\ -0.5 + 2^{-7}, & \text{with probability } 0.5 + 2^{-7}. \end{cases}$$

The expected value of a is E$a = 0$. Assume, however, that a is quantized to fixed-point, with six bits plus sign (so that $\epsilon = 2^{-7}$). If rounding is used, the quantized variable will have the distribution[5]

$$\text{fx}[a] = \begin{cases} 0.5 + 2^{-6}, & \text{with probability } 0.5 - 2^{-7} \\ -0.5, & \text{with probability } 0.5 + 2^{-7}. \end{cases}$$

The mean of fx$[a]$ is -2^{-13}. Another situation where finite-precision errors introduce nonzero mean variables is discussed in [20]. Basically, this reference shows that the rounding error of a product $\delta = \text{fx}[a\cdot b] - ab$ may not have zero mean in some situations.

Thus, a zero-mean variable may become nonzero-mean after quantization or after a fixed-point multiplication. This small mean might cause a slow drift of the LMS estimates, causing the algorithm to overflow. We illustrate this effect by simulating an $M = 2$ LMS filter whose input regressors satisfy (the values shown below are chosen such that the weight drift effect is amplified)

$$\boldsymbol{x}_k = \begin{cases} [0.5 \quad -0.5]^T, & \text{with probability } 0.5 \\ -[0.5 \quad -0.5]^T, & \text{with probability } 0.5. \end{cases}$$

The noise is uniformly distributed with variance $\sigma_v^2 = 1/3 \times 10^{-3}$, the step-size is $\mu = 0.15$, and the true weight vector is $\boldsymbol{w}_* = [\sqrt{0.2} \quad -\sqrt{0.2}]^T$. The weight estimates of the LMS recursion in finite precision are denoted by $\boldsymbol{z}_k$, and they are computed via (the rounding function is implemented as described in [20])

$$\boldsymbol{z}_{k+1} = \boldsymbol{z}_k + \text{fx}[\mu\boldsymbol{x}_k\text{fx}[e(k)]].$$

Fig. 2 plots the values of $\|\boldsymbol{z}_k\|_\infty$. We see that overflow occurs at approximately $k = 250$.

C. Solution of Drift Problem by Leakage

The leakage term in (4) prevents unbounded growth of the weight vectors from occurring. In the first of our examples, using leakage, we obtain the recursion for the error vector

$$\tilde{\boldsymbol{w}}_{k+1}^l = \left(1 - \mu\alpha_0 - \frac{\mu}{k+1}\right)\tilde{\boldsymbol{w}}_k^l + \mu\alpha_0\boldsymbol{w}_* - \mu\,\frac{10^{-4}}{\sqrt{k+1}}.$$

This recursion can be shown to result in a bounded sequence $\{\tilde{\boldsymbol{w}}_k^l\}$ if $0 < \mu < 2/(\alpha_0 + 1)$. More generally, the following result can be established for leaky LMS (see [8]).

Lemma 1 (BIBS Stability of Leaky LMS): Consider the leaky LMS algorithm (4) in infinite-precision arithmetic. If $\mu < 2/(\alpha_0 + \beta)$, then $\|\boldsymbol{w}_k^l\|$ remains bounded if the noise sequence $\{v(k)\}$ is bounded.

In other words, under the condition $\mu < 2/(\alpha_0 + \beta)$, the leaky LMS algorithm is bounded-input bounded-state (BIBS)-stable, where we treat the weight estimates as the state and the noise sequence $v(k)$ as the input. This result can be extended to finite-precision arithmetic, as follows from the arguments we provide in Section V. We state the conclusion here.

Lemma 2 (Fixed-Point Stability of Leaky LMS): The leaky LMS algorithm implemented in fixed-point arithmetic guarantees that the sequence $\{\boldsymbol{w}_k^l\}$ is bounded if $\{v(k)\}$ is bounded, and $|1 - \mu\alpha_0 - \mu\beta| \leq |1 - \mu\alpha_0| < 1$.

[5] The result depends on exactly how the rounding function is implemented. For example, $-0.5 + 2^{-7}$ might be rounded to $-0.5 + 2^{-6}$ in some machines.

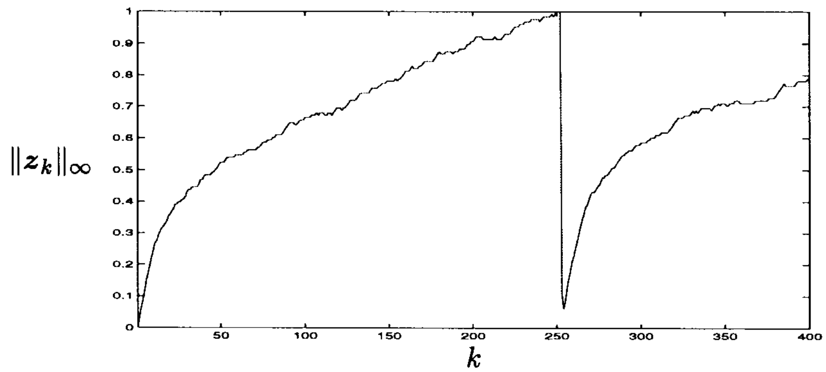


Fig. 2. Effect of small nonzero mean finite-precision error with the LMS algorithm. The plot shows $\|z_k\|_\infty$ for the $M = 2$ example described in the text.

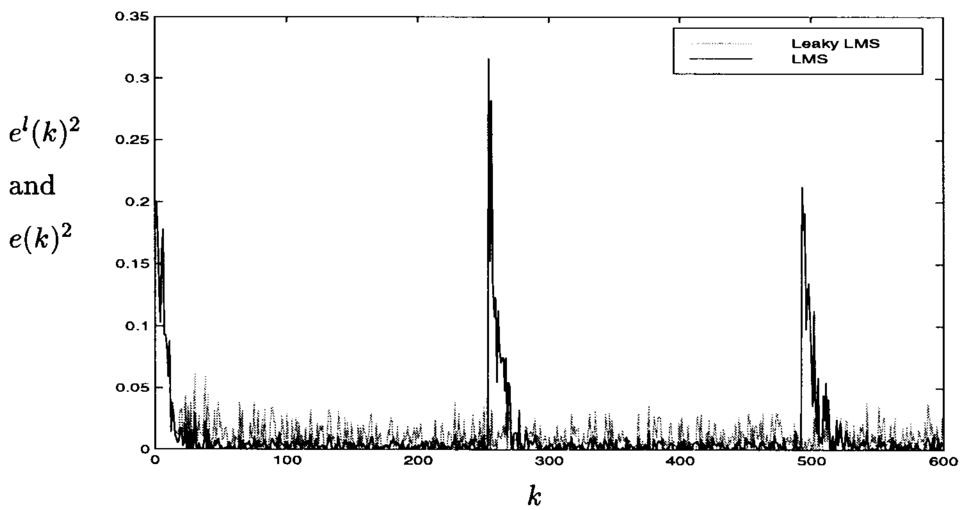


Fig. 3. Comparison of the squared errors $e(k)^2$ (LMS) and $e^l(k)^2$ (leaky LMS) for the $M = 2$ example of Fig. 2. The darker curve with the spikes corresponds to LMS. No averaging was performed.

Proof: This result follows from the proof of Theorem 4 further ahead. ◇

D. The Bias Problem of Leaky LMS

Although the leaky LMS algorithm (4) solves the weight-drift problem, it leads to biased estimates, which can be seen as follows. The error equation for leaky LMS is given by

$$\tilde{\boldsymbol{w}}^l_{k+1} = ((1-\mu\alpha_0)I - \mu\boldsymbol{x}_k\boldsymbol{x}_k^T)\tilde{\boldsymbol{w}}^l_k + \mu\alpha_0\boldsymbol{w}_* - \mu\boldsymbol{x}_k v(k).$$

Now, assume that $\{\boldsymbol{x}_k\}$ and $\{v(k)\}$ are stationary, independent, and identically distributed (iid) sequences. Assume also that these sequences are independent of each other and have zero mean. Computing the expectation of $\tilde{\boldsymbol{w}}^l_{k+1}$, we obtain

$$\mathrm{E}\tilde{\boldsymbol{w}}^l_{k+1} = ((1-\mu\alpha_0)I - \mu R)\mathrm{E}\tilde{\boldsymbol{w}}^l_k + \mu\alpha_0\boldsymbol{w}_*$$

where $R = E\boldsymbol{x}_k\boldsymbol{x}_k^T$. Therefore, if all eigenvalues of the coefficient matrix $[(1-\mu\alpha_0)I - \mu R]$ are strictly less than one in magnitude, we obtain in steady-state

$$\lim_{k\to\infty} \mathrm{E}\tilde{\boldsymbol{w}}^l_k = \alpha_0(\alpha_0 I + R)^{-1}\boldsymbol{w}_*. \tag{10}$$

That is, the average weight error $\mathrm{E}\tilde{\boldsymbol{w}}^l_k$ computed by the leaky LMS algorithm will not converge to zero, even in ideal conditions (positive-definite R, zero noise, and no quantization errors).

The conventional solution to the bias problem in (4) has been to use a very small α_0. However, this choice has its drawbacks. A value of α_0 too small might not be capable of countering the effects of finite-precision arithmetic. In addition, even a small α_0 might create a significant bias, as shown in the simulation in Fig. 3. The lighter curve is the plot of the squared error $e^l(k)^2$ (not its average) computed by the leaky LMS algorithm for the same environment as in Fig. 2 (same $\boldsymbol{w}_*$ and noise and input statistics). The step size is again $\mu = 0.15$, and the leakage parameter is $\mu\alpha_0 = 2^{-6}$. Note that this is the second smallest value that could be chosen for $\mu\alpha_0$, corresponding to twice the value of the least-significant bit (1 LSB $= 2^{-7}$ in this example).

The darker curve is a plot of the squared error computed by the LMS algorithm $e(k)^2$. Almost all the time, $e(k)^2$ is smaller than $e^l(k)^2$, but there are spikes when overflow occurs. [This kind of sudden worsening of the performance is what turns the filter unusable for some applications.] Comparing the results for LMS and leaky LMS, we note that although the latter avoids overflow, the level of the error is significantly increased. More examples are provided in Section VI.

E. Objectives of the Analyzes in the Sequel

The above examples and discussion motivate us to pursue in this paper other ways to solve the bias and drift problems without compromising the performance of the adaptive algorithm. We do so by introducing a leakage-based algorithm, called circular-leaky (6) and by studying the performance of the modified switching-σ algorithm (8). The purpose of the discussion in the sequel is twofold.

1) We want to establish that the modified switching-σ and the proposed circular-leaky algorithms solve the drift problem even under the more demanding environment of a finite-precision implementation. In particular, we determine conditions on the leakage parameters so that rounding effects will not contribute to drift.
2) We want to establish that both algorithms also compute asymptotically unbiased estimates when the regressor covariance matrix is positive-definite $(R > 0)$.

We employ two tools in our analysis. The first tool is a stochastic averaging analysis, which is used in Section IV to establish point 2) above. The second tool is based on a deterministic Lyapunov stability analysis, which is used in Section V to show that both algorithms avoid unbounded growth of the weight error vector.

IV. Stochastic Performance Analysis

In this section, we show that the estimates provided by circular-leaky and switching-σ algorithms are unbiased. In fact, we establish a stronger conclusion, namely, that this property holds even when using fixed-point arithmetic with rounding. These results are established by relying on averaging methods, which we first review.

A. Averaging Analysis

Averaging methods provide a powerful means to study the performance and stability of adaptive algorithms under the assumption of sufficiently small step-sizes. There are many excellent expositions on the subject (see, for example, [21], [24], [25], and the references therein). For this reason, we restrict our discussion here only to the steady-state results that are needed in our derivation, following, for the most part, [21].

Consider an adaptive update of the general form

$$\tilde{\boldsymbol{w}}_{k+1} = \tilde{\boldsymbol{w}}_k + \mu f(k, \tilde{\boldsymbol{w}}_k), \text{with some initial condition } \tilde{\boldsymbol{w}}_0 \tag{11}$$

where $\tilde{\boldsymbol{w}}_k$ is the error vector we want to minimize. The function f is stochastic, i.e., for every k and $\tilde{\boldsymbol{w}}_k$, $f(k, \tilde{\boldsymbol{w}}_k)$ is a random vector. We could be more explicit in the notation and write $f(\boldsymbol{\xi}_k, \tilde{\boldsymbol{w}}_k)$, where $\{\boldsymbol{\xi}_k\}$ is a stochastic sequence. For example, in the LMS case, we have $f(k, \tilde{\boldsymbol{w}}_k) = -\boldsymbol{x}_k \boldsymbol{x}_k^T \tilde{\boldsymbol{w}}_k - \boldsymbol{x}_k v(k)$, and $\boldsymbol{\xi}_k$ would be formed from $\boldsymbol{x}_k$ and $v(k)$. Now, define the averaged function f_{av} as

$$f_{av}(k, \tilde{\boldsymbol{w}}) = \mathrm{E} f(k, \tilde{\boldsymbol{w}})$$

where $\tilde{\boldsymbol{w}}$ is considered *constant* for the computation of the expected value. For example, if $\{\boldsymbol{x}_k\}$ is a stationary sequence, the averaged function for LMS is $f_{av}(k, \tilde{\boldsymbol{w}}) = -R\tilde{\boldsymbol{w}}$. In addition, define the *averaged system*

$$\tilde{\boldsymbol{w}}_{k+1}^{av} = \tilde{\boldsymbol{w}}_k^{av} + \mu f_{av}(k, \tilde{\boldsymbol{w}}_k^{av}), \quad \tilde{\boldsymbol{w}}_0^{av} = \tilde{\boldsymbol{w}}_0. \tag{12}$$

The fully averaged system does not allow us to predict the steady-state performance of the adaptive algorithm. For this purpose, it is necessary to study the *partially averaged system*

$$\tilde{\boldsymbol{w}}_{k+1}^{pav} = [I + \mu \nabla_{\tilde{\boldsymbol{w}}} f_{av}(\mathbf{0})] \tilde{\boldsymbol{w}}_k^{pav} + \mu (f(k, \mathbf{0}) - f_{av}(k, \mathbf{0})) \tag{13}$$

where $\nabla_{\tilde{\boldsymbol{w}}} f_{av}(\mathbf{0})$ denotes the value of the gradient of f_{av} (with respect to $\tilde{\boldsymbol{w}}$) at the origin.

The following result, which is proven in [21, ch. 9], shows that if the step-size μ is sufficiently small, the original estimates $\tilde{\boldsymbol{w}}_k$ will remain close to the partially averaged estimates $\tilde{\boldsymbol{w}}_k^{pav}$ and that the steady-state covariance of $\tilde{\boldsymbol{w}}_k$ will be close to that of $\tilde{\boldsymbol{w}}_k^{pav}$. The theorem assumes that the $\{\boldsymbol{\xi}_k\}$ satisfy a *uniform mixing* property. Essentially, this condition says that the correlation of $\boldsymbol{\xi}_i$ and $\boldsymbol{\xi}_j$ dies out as $|i - j|$ increases (see [22]).

Theorem 1 (Averaging Result): Consider the error equation (11) and its averaged forms (12) and (13), where the sequence $\{\boldsymbol{\xi}_k\}$ is uniform-mixing (see [21, p. 357]). Assume the following.

i) The origin $\mathbf{0}$ is an exponentially stable equilibrium point of the averaged system (12) with decay rate $O(\mu)$.
ii) The gradient $\nabla_{\tilde{\boldsymbol{w}}} f_{av}(k, \tilde{\boldsymbol{w}})$ exists and is continuous at the origin.
iii) There exists a constant c such that, for any vectors $\boldsymbol{a}$ and $\boldsymbol{b}$, the following Lipschitz condition holds

$$\|\nabla_{\tilde{\boldsymbol{w}}} f(k, \boldsymbol{a}) - \nabla_{\tilde{\boldsymbol{w}}} f(k, \boldsymbol{b})\| \le c \|\boldsymbol{a} - \boldsymbol{b}\|.$$

Under these three conditions, $\tilde{\boldsymbol{w}}_k$ obtained from (11) satisfies

$$\lim_{\mu \to 0} \sup_{k \ge 0} \mathrm{P}\{\|\tilde{\boldsymbol{w}}_k - \tilde{\boldsymbol{w}}_k^{pav}\| > \epsilon\} = 0 \tag{14}$$

for every $\epsilon > 0$, and

$$\lim_{\mu \to 0} \lim_{k \to \infty} \left(\frac{1}{\mu} \mathrm{E} \tilde{\boldsymbol{w}}_k \tilde{\boldsymbol{w}}_k^T \right) = \lim_{\mu \to 0} \lim_{k \to \infty} \left(\frac{1}{\mu} \mathrm{E} \tilde{\boldsymbol{w}}_k^{pav} \tilde{\boldsymbol{w}}_k^{pav,T} \right). \tag{15}$$

$\diamondsuit$

Using the LMS algorithm as an example again, we have

$$\nabla_{\tilde{\boldsymbol{w}}} f_{av}(\mathbf{0}) = -R, \quad f(k, \mathbf{0}) - f_{av}(k, \mathbf{0}) = -\boldsymbol{x}_k v(k).$$

B. Circular-Leaky Algorithm

We now consider the circular-leaky algorithm (6) and show that contrary to the standard leaky form (4), circular leaky does not lead to biased estimates. We establish this result in the more demanding context of a fixed-point implementation.

Since we are interested in accounting for finite precision effects, we need to distinguish between the infinite-precision and the finite precision versions of the update laws. For this reason, we shall denote the weight error vector in finite precision by $\tilde{\boldsymbol{z}}_k^c$ (and reserve $\tilde{\boldsymbol{w}}_k^c$ for the infinite-precision case). Using (A.5) from Appendix A, we can show that $\tilde{\boldsymbol{z}}_k^c$ satisfies the recursion

$$\begin{aligned} \tilde{\boldsymbol{z}}_{k+1}^c = &(I - \mu \alpha_c(z_{k,\bar{k}}) \boldsymbol{e}_{\bar{k}} \boldsymbol{e}_{\bar{k}}^T - \mu \boldsymbol{x}_k \boldsymbol{x}_k^T) \tilde{\boldsymbol{z}}_k^c \\ &+ \mu \alpha_c(z_{k,\bar{k}}) \boldsymbol{e}_{\bar{k}} \boldsymbol{e}_{\bar{k}}^T \boldsymbol{w}_* - \mu \boldsymbol{x}_k v(k) - \boldsymbol{\delta}_k^c \end{aligned} \tag{16}$$

where the variable $\boldsymbol{\delta}_k^c$ accounts for all finite-precision errors and satisfies certain bounds given by (A.2)–(A.4), and $z_{k,\bar{k}} = w_{*,\bar{k}} - \tilde{z}_{k,\bar{k}}$ with $\bar{k} = (k \bmod M)$. We shall assume for simplicity that $\boldsymbol{\delta}_k^c$ is independent of all other quantities.[6]

To use Theorem 1, we need to prove that the fixed-point circular-leaky error equation (16) and its averaged counterparts satisfy i)–iii) given in the statement of the theorem. Averaging the error equation (16) over the input $\boldsymbol{x}_k$, the noise $v(k)$, and the finite-precision errors $\boldsymbol{\delta}_k^c$, we obtain the recursion[7]

$$\tilde{\boldsymbol{z}}_{k+1}^{av} = (I - \mu\alpha_c(w_{*,\bar{k}} - \tilde{z}_{k,\bar{k}}^{av})\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T - \mu R)\tilde{\boldsymbol{z}}_k^{av} + \mu\alpha_c(w_{*,\bar{k}} - \tilde{z}_{k,\bar{k}}^{av})\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T\boldsymbol{w}_*. \quad (17)$$

It is shown in Appendix B that this recursion satisfies i)–iii) for values of μ and α_0 that satisfy

$$\mu\left(1 + \frac{1}{\eta_c}\right)\alpha_0 < 2 - \mu\lambda_{\max}(R) \quad (18)$$

where $\eta_c > 0$ is a constant that satisfies $C_1 \geq (1+\eta_c)\|\boldsymbol{w}_*\|_\infty$. The partially averaged system is further given by

$$\tilde{\boldsymbol{z}}_{k+1}^{pav} = (I - \mu R)\tilde{\boldsymbol{z}}_k^{pav} - \mu\boldsymbol{x}_k v(k) - \boldsymbol{\delta}_k^c.$$

This is, in fact, the same partially averaged recursion that would result for the LMS algorithm in fixed-point arithmetic. Therefore, in steady-state, the circular-leaky algorithm will behave like the LMS algorithm. In particular, circular-leaky computes asymptotically unbiased estimates since the estimates computed by LMS have this property. The value of the steady-state mean-square error, $\lim_{k\to\infty} \mathrm{E}e^c(k)^2$ can then be obtained from the literature (e.g., [15] and [28]) and is stated below, with the necessary conditions.

Theorem 2 (Steady-State Performance of Circular-Leaky): Assume that $\{\boldsymbol{x}_k\}$, $\{v(k)\}$, and $\{\boldsymbol{\delta}_k^c\}$ are stationary, have zero mean, and satisfy $\mathrm{E}\boldsymbol{x}_k\boldsymbol{x}_k^T = R > 0$. Assume further that $\{v(k)\}$ is iid and independent of $\{\boldsymbol{x}_k\}$ and that this last sequence is uniform mixing. Then, if the step-size μ is small enough and (18) holds, the circular-leaky estimates $\boldsymbol{z}_k^c$ are asymptotically unbiased, and in the steady-state, we have

$$\lim_{k\to\infty} \mathrm{E}e^c(k)^2 \approx \sigma_v^2 + \mu(\sigma_v^2 + \sigma_d^2)\cdot\frac{\mathrm{Tr}(R)}{2} + \frac{\mathrm{Tr}(\sigma_d^2(I+R))}{2} \quad (19)$$

where $\sigma_d^2 = 2^{-2B}/12$ for a fixed-point implementation with B bits plus sign.

Proof: The complete argument requires some effort and is given in Appendix B. ◇

This theorem shows that circular-leaky has essentially the same good performance as LMS if $R > 0$ and (18) is satisfied. Therefore, the parameters α_0, μ, and η_c must be chosen so that (18) holds. We provide design examples in Sections VI and VII.

C. The Modified Switching-σ Algorithm

A similar result can be obtained for the modified switching-σ algorithm, but the conditions are less restrictive than for circular-leaky. (The finite-precision error equation for switching-σ is given in Appendix A.)

Theorem 3 (Steady-State Performance of Switching-σ): Assume that $\{\boldsymbol{x}_k\}$, $\{v(k)\}$, and $\{\boldsymbol{\delta}_k^c\}$ are stationary, have zero mean, and satisfy $\mathrm{E}\boldsymbol{x}_k\boldsymbol{x}_k^T = R > 0$. Assume further that $\{v(k)\}$ is iid and independent of $\{\boldsymbol{x}_k\}$ and that this last sequence is uniform mixing. Then, if the step-size μ is small enough and $S_1 > \|\boldsymbol{w}_*\|$, the switching-$\sigma$ estimates $\boldsymbol{z}_k^s$ are asymptotically unbiased, and in the steady state, we have

$$\lim_{k\to\infty} \mathrm{E}e^s(k)^2 \approx \sigma_v^2 + \mu(\sigma_v^2 + \sigma_d^2)\cdot\frac{\mathrm{Tr}(R)}{2} + \frac{\mathrm{Tr}(\sigma_d^2(I+R))}{2}. \quad (20)$$

Proof: As we did for Theorem 2 in Appendix B, we need to check conditions i)–iii) from Theorem 1. Conditions ii) and iii) can be checked as before, but a stronger result can be obtained if we modify the argument for checking condition i). Indeed, instead of working with the averaged error $\tilde{\boldsymbol{z}}_k^{av}$ we now work with the averaged version of $\boldsymbol{z}_k^s$, namely

$$\boldsymbol{z}_{k+1}^{av} = ((1 - \alpha_s(\boldsymbol{z}_k^{av}))I - \mu R)\boldsymbol{z}_k^{av} + \mu R\boldsymbol{w}_*. \quad (21)$$

Condition i) is equivalent to proving that $\boldsymbol{w}_*$ is an exponentially stable equilibrium point for (21).

We show in Appendix C that there exists a K such that $\|\boldsymbol{z}_k^{av}\| < S_1$ for all $k \geq K$. Therefore, for large k, the leakage term remains equal to zero $(\alpha_s(\boldsymbol{z}_k^{av}) = 0)$, and the averaged recursion (21) becomes

$$\boldsymbol{z}_{k+1}^{av} = (I - \mu R)\boldsymbol{z}_k^{av} + \mu R\boldsymbol{w}_*$$

from which we conclude that $\boldsymbol{z}_k^{av} \to \boldsymbol{w}_*$ exponentially fast if μ satisfies $0 < \mu\lambda_{\max}(R) < 2$. Having verified that condition i) is satisfied, we can then apply Theorem 1 to obtain (20), just as we did for circular-leaky in Theorem 2. ◇

V. Deterministic Performance Analysis: Stability

Having shown that the circular-leaky and switching-σ algorithms do not introduce bias, we now prove that they also avoid drift for any bounded input and noise sequences (provided that the step-size is small enough). The following result is proved in Appendix D.

[6]We should note that the results obtained with this "linear" model for the quantization error are valid if the so-called stopping phenomenon does not occur (i.e., when the step-size is large enough; see [15] and [26]). Reference [14] considers an alternative nonlinear model for finite-precision errors, albeit under the more restrictive assumption of iid Gaussian input variables with $R = \sigma_x^2 I$—see the comments immediately before the concluding remarks of [27].

[7]To simplify the notation, in this section we will drop the superscript c from the averaged variables.

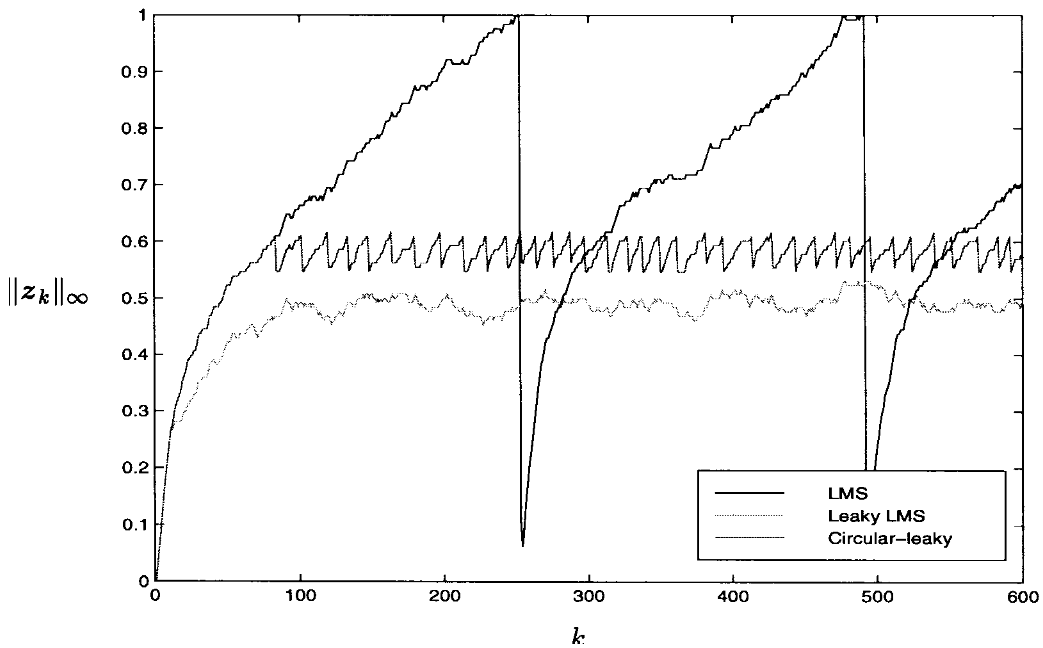


Fig. 4. Application of the LMS, leaky LMS, and circular-leaky algorithms to the example of Fig. 2.

Theorem 4 (Stability of Switching-σ: If μ satisfies

$$|1-\mu\alpha_0-\mu\beta| \leq |1-\mu\alpha_0| < 1 \tag{22}$$

then the fixed-point switching-σ algorithm is bounded-input bounded-state stable [with the $v(k)$ as input and z_k^s as the state]. ◇

The stability analysis of the circular-leaky algorithm is similar in spirit to, although more involved than, that for switching-σ. However, the fact that leakage is applied (or not) to only one tap at each time instant in a prespecified circular order requires a closer study to verify stability. This is because it can happen that $\|z_k^c\|_\infty$ is large, but the tap that is being checked by $\alpha_c(\cdot)$ at time k, i.e., $z_{k,\bar{k}}^c$, is small, so that no leakage is applied. We then need to verify that such possibilities do not cause instability. To account for this scenario, we need to look at the variation of the norm of z_k^c after M time-steps, i.e., we compare $\|z_{k+M}^c\|$ with $\|z_k^c\|$. The reassuring conclusion is that circular leaky is also stable—see Appendix E.

Theorem 5 (Stability of Circular-Leaky): If μ satisfies

$$|1-\mu\alpha_0-\mu\beta| < 1 \quad \text{and}$$
$$\sqrt{1-\frac{\mu\alpha_0(2-\mu\alpha_0)}{M}}+\mu M\beta < 1 \tag{23}$$

then the fixed-point circular-leaky algorithm is bounded-input bounded-state stable.

VI. Simulation Results

We now present several simulation results. We first apply the circular-leaky algorithm to the example of drift shown in Fig. 2. In that example, we had $\mu = 0.15$, $\|w_*\|_\infty = 0.44$, and $\mu\lambda_{\max}(R) = 0.075$. As in Section III-A, we implemented the algorithms in fixed point with 7 bits plus sign.

To choose the parameters for the circular-leaky algorithm, we need bounds on $\|w_*\|_\infty$ and on $\lambda_{\max}(R)$. Assume that the bound $\|w_*\|_\infty \leq 0.55$ is given. Choosing $\mu\alpha_0 = 0.1$, (18) requires that $\eta_c > 0.055$, and thus, we need $C_1 > 0.58$. We chose $C_1 = 0.60$ and $C_2 = 0.61$. The results are shown in Fig. 4, where we plotted $\|z_k\|_\infty$ for circular-leaky, LMS, and for leaky LMS with $\mu\alpha_0 = 0.0156$ (note that for fixed-point numbers with 7 bits plus sign, this value of $\mu\alpha_0$ is only the second smallest representable number). Since, in this example, the input distribution does not satisfy $R > 0$, the LMS algorithm overflows, as we saw in Section III-A. Circular-leaky (middle curve) prevents the overflow, keeping the estimates at a safe level. The squared error curves $e^l(k)^2$ and $e^c(k)^2$ are presented in Fig. 5 (without averaging), where we see that the error level is clearly smaller for circular-leaky (dark curve) than for leaky LMS (light curve).

In Fig. 6, we plot the ensemble-averaged learning curves (i.e., $\mathrm{E}e(k)^2$) computed by the same algorithms, when $R = \mathrm{diag}(0.25, 0.25)$. Note that the performance of leaky LMS (light curve) is considerably worse, even though we have used the second smallest value for $\mu\alpha_0$.

We now present two examples to highlight the robustness of circular-leaky. In the first one, we used $M = 10$ and, again, $\mu = 0.15$, $\mu\alpha_0 = 0.1$, $C_1 = 0.60$, $C_2 = 0.61$, and $\|w_*\|_\infty = 0.44$. The input sequence has covariance matrix with nine zero eigenvalues and one eigenvalue equal to 2.5. We also artificially added 1 LSB $(= 1/128)$ to every entry of z_k at every time step, in order to make the task of circular-leaky and leaky LMS more challenging. In Fig. 7, we plotted $\|z_k\|_\infty$ for LMS and circular-leaky. The discontinuities in the LMS plot correspond to points where overflow occurs; circular-leaky avoids overflow even in this demanding environment.

The last example has $M = 100$, and was implemented with 11 bits plus sign. The input and true weight were

$$w_* = [0.06 \quad -0.06 \quad 0.06 \cdots -0.06]^T \quad \text{and}$$
$$x_k = [\pm 0.5 \quad \mp 0.5 \quad \pm 0.5 \quad \cdots \mp 0.5]^T.$$

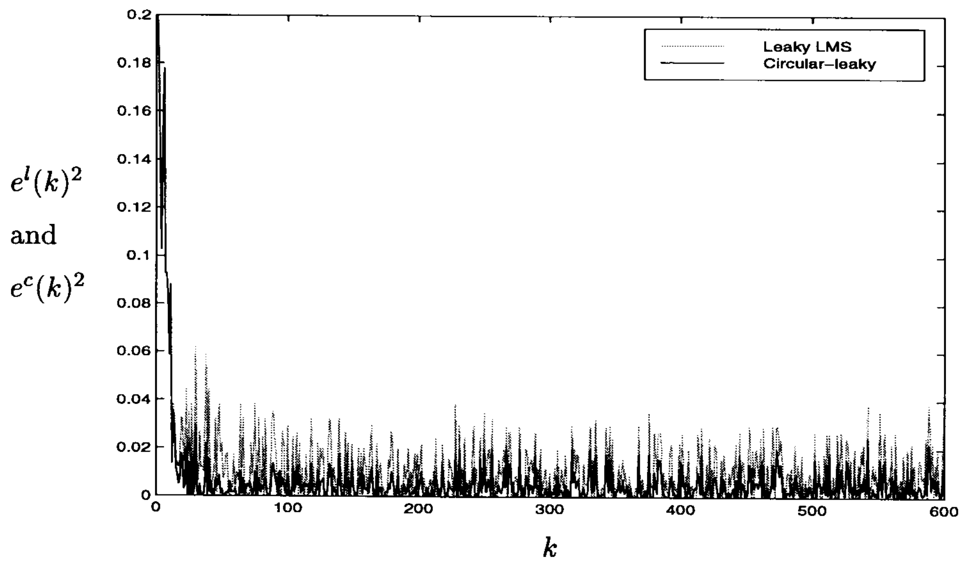


Fig. 5. Squared error curves for leaky LMS and circular-leaky in the same example as in Fig. 2.

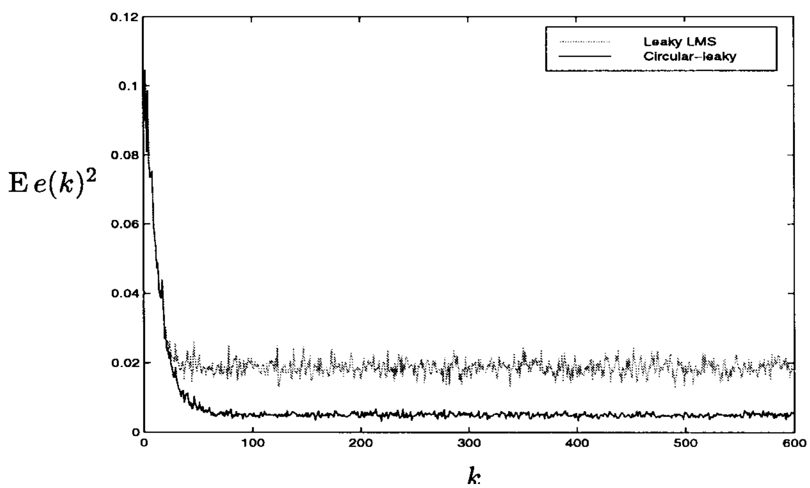


Fig. 6. Learning curves ($e(k)^2$ averaged over 100 runs) for leaky LMS and circular-leaky, with $R = \mathrm{diag}(0.25, 0.25)$.

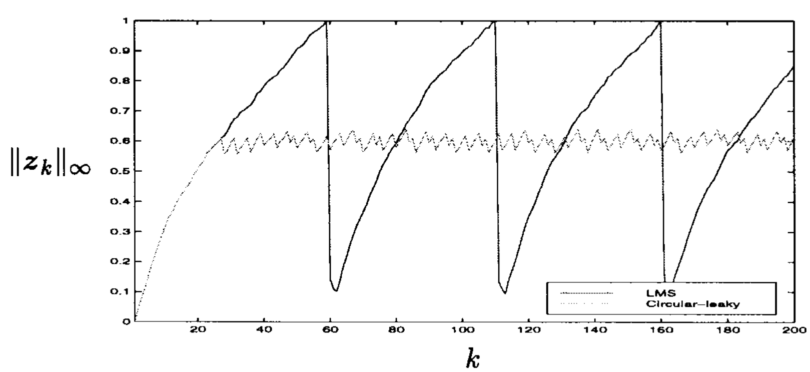


Fig. 7. $\|z_k\|_\infty$ for LMS and circular-leaky, with $M = 10$, $\mu\alpha_0 = 0.1$ and $C_1 = 0.60$, $C_2 = 0.61$.

The input correlation matrix had 99 zero eigenvalues and one eigenvalue equal to 25. The other parameters were $\mu = 0.01$, $\mu\alpha_0 = 0.1$, $\sigma_v^2 = 1/3 \times 10^{-3}$, $C_1 = 0.21$, and $C_2 = 0.22$. The plots of $\|\boldsymbol{z}_k\|_\infty$ (LMS) and $\|\boldsymbol{z}_k^c\|_\infty$ (circular-leaky) are shown in Fig. 8.

VII. Filter Design

In order to choose the design parameters for the circular-leaky algorithm (6), a bound $W_\infty \geq \|\boldsymbol{w}_*\|_\infty$ is necessary. This bound could be obtained from approximations for the statistics

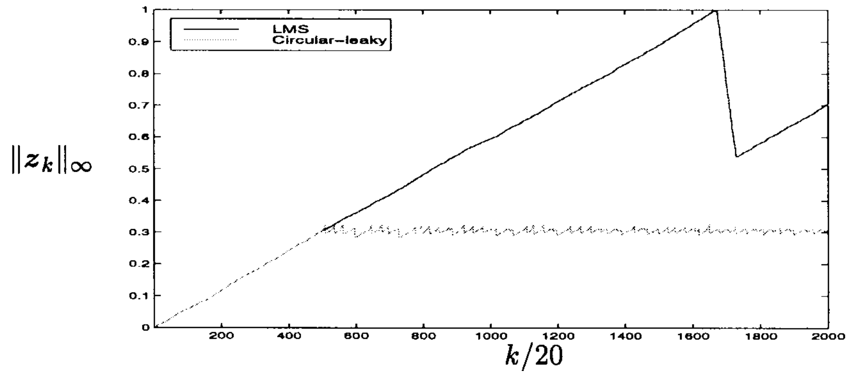


Fig. 8. $\|z_k\|_\infty$ for LMS and circular-leaky, with $M = 100$, $\mu\alpha_0 = 0.1$. Only one out of every 20 samples is plotted.

of the signals involved. For example, if we know that the true covariance $R = \mathrm{E}\boldsymbol{x}_k\boldsymbol{x}_k^T$ and cross-correlation $\boldsymbol{p} = \mathrm{E}y(k)\boldsymbol{x}_k$ are inside balls, say

$$R = \tilde{R} + \delta R, \quad \boldsymbol{p} = \tilde{\boldsymbol{p}} + \boldsymbol{\delta p}$$

with $\|\delta R\|_\infty \le \eta_R$, $\|\boldsymbol{\delta p}\|_\infty \le \eta_p$, a bound for $\|\boldsymbol{w}_*\|_\infty$ could be computed from

$$\|\boldsymbol{w}_*\|_\infty = \|R^{-1}\boldsymbol{p}\|_\infty = \|(\tilde{R} + \delta R)^{-1}(\tilde{\boldsymbol{p}} + \boldsymbol{\delta p})\|_\infty.$$

The matrix inversion lemma can be applied to obtain

$$\|\boldsymbol{w}_*\|_\infty \le \|\tilde{R}^{-1}\tilde{\boldsymbol{p}} - \tilde{R}^{-1}(I + \delta R)^{-1}\delta R\tilde{R}^{-1}\tilde{\boldsymbol{p}} + \tilde{R}^{-1}\boldsymbol{\delta p}\| + O(\eta_R\eta_p).$$

Assuming that $\eta_R \ll 1$ so that $(I + \delta R)^{-1} \approx I$, we obtain

$$\|\boldsymbol{w}_*\|_\infty \le \|\tilde{R}^{-1}\tilde{\boldsymbol{p}}\|_\infty + \|\tilde{R}^{-2}\tilde{\boldsymbol{p}}\|_\infty\eta_R + \|\tilde{R}^{-1}\|_\infty\eta_p.$$

Assuming that a bound for $\|\boldsymbol{w}_*\|_\infty$ is available, the design is made in the following way. Begin by choosing adequate values for μ and α_0 satisfying $0 < \mu\alpha_0 < 1$, and use (18) to find the smallest possible η_c. The parameter C_1 is then chosen from $C_1 \ge (1 + \eta_c)W_\infty$. If the resulting C_1 is too large, we can reduce α_0 or μ or both to allow for a smaller η_c in (18) and repeat the above procedure.

This procedure guarantees that circular-leaky is unbiased if the step-size used is small enough (unfortunately, as always with the use of averaging results, we cannot tell how small μ must be).

Although it is not a necessary condition, (18) is not excessively conservative. We also proved that circular-leaky is stable if (23) is satisfied. This condition, however, is conservative; the filter may be stable even if the condition is not satisfied.

VIII. Concluding Remarks

We proposed a leakage-based LMS algorithm, called circular-leaky, that avoids the drift problem of LMS without the drawbacks of leaky LMS, namely, the introduction of bias and the higher computational cost. In addition, circular-leaky is cheaper to implement than leaky LMS (for large filter lengths, the computational cost of circular-leaky is only slightly larger than that of standard LMS). The arguments in this paper relied on results from averaging theory and (Lyapunov) stability theory. They essentially established that for small enough step sizes, the two problems of bias and drift are solved by circular-leaky and by switching-σ.

A point that deserves further investigation is the choice of the leakage function $\alpha_c(\cdot)$. Our choice of a differentiable $\alpha_c(\cdot)$ was motivated by the fact that the averaging results of Theorem 1 are not applicable to discontinuous $f(k, \tilde{\boldsymbol{w}}_k)$. The stability results of Section V, however, are still valid if instead of (7) we choose a hard-limiting α_c, say, one of the form

$$\alpha_c(a) = \begin{cases} \alpha_0, & \text{if } |a| > C_1 \\ 0, & \text{if } |a| \le C_1. \end{cases}$$

This is, of course, a simpler function to implement than (7). In the related work [19], we used a deterministic argument (rather than one based on averaging theory) and provided simulation results, showing that if this alternative leakage function is used, circular-leaky still computes unbiased estimates.

Appendix A
Finite Precision Update Laws

We assume, as explained in Section III-B, that all algorithms are implemented using fixed-point arithmetic, where all variables are stored with B bits plus sign, and that rounding is used (with $\epsilon = 2^{-B-1}$). We also assume that $\boldsymbol{x}_k$ and $y(k)$ represent already-quantized variables [i.e., there are exact fixed-point representations for $y(k)$ and $\boldsymbol{x}_k$].

In fixed-point arithmetic, additions are performed without error if the variables are scaled so that overflow does not occur. On the other hand, there is an error when performing a multiplication, say, $\mathrm{fx}[ab] = ab + \delta$, where $|\delta| \le \epsilon$. We usually assume that δ is a random variable with uniform distribution and zero mean [so that its variance is $\sigma_d^2 = (2^{-2B}/12)$] and that δ is independent of both a and b. It is also common to assume that errors in two different operations are independent. Note that none of these assumptions is exactly true—in particular, there are systems in which $\mathrm{E}\delta$, although small, is nonzero (see the discussion in Section III-B).

To differentiate between the infinite and finite-precision versions of the various algorithms, the weight estimates computed by the fixed-point algorithms are denoted by $\boldsymbol{z}_k$ (for LMS), $\boldsymbol{z}_k^c$ (for circular-leaky), and $\boldsymbol{z}_k^s$ (for switching-σ). Similarly, the weight error vectors are $\tilde{\boldsymbol{z}}_k$, $\tilde{\boldsymbol{z}}_k^c$, and $\tilde{\boldsymbol{z}}s_k$.

Circular-Leaky: In fixed-point, the update law of circular-leaky is given by

$$\boldsymbol{z}_{k+1}^c = \text{fx}[(I-\mu\alpha_c(z_{k,\bar{k}})\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T)\boldsymbol{z}_k^c] + \text{fx}[\mu\boldsymbol{x}_k\text{fx}[e^c(k)]] \quad \text{(A.1)}$$

where $e^c(k) = \boldsymbol{z}_k^T\boldsymbol{x}_k$. We now expand the terms fx[·] in (A.1), starting with $e_Q^c(k) \triangleq \text{fx}[e^c(k)]$. Following [29], we obtain

$$e_Q^c(k) = \text{fx}[\boldsymbol{z}_k^T\boldsymbol{x}_k] = \boldsymbol{z}_k^T\boldsymbol{x}_k + \eta(k)$$

where the error $\eta(k)$ satisfies[8] $|\eta(k)| \le M\epsilon$ and $\text{E}\eta(k)^2 \triangleq \sigma_\eta^2 = M\sigma_d^2$. Similarly, define the error $\boldsymbol{\xi}_k$ by $\boldsymbol{\xi}_k \triangleq \text{fx}[\mu\boldsymbol{x}_k e_Q^c(k)] - \mu\boldsymbol{x}_k e_Q^c(k)$. Expanding the term fx[·], we obtain

$$\text{fx}[\mu\boldsymbol{x}_k e_Q^c(k)] = \text{fx}[\text{fx}[\mu e_Q^c(k)]\boldsymbol{x}_k] = (\mu e_Q^c(k) + \xi_k')\boldsymbol{x}_k + \boldsymbol{\xi}_k''$$

where $|\xi_k'| \le \epsilon$, $\text{E}\xi_k'^2 = \sigma_d^2$, $\|\boldsymbol{\xi}_k''\| \le \sqrt{M}\epsilon$, and $\text{E}\boldsymbol{\xi}_k''\boldsymbol{\xi}_k''^T = \sigma_d^2 I$. We conclude that

$$\|\boldsymbol{\xi}_k\| \le \sqrt{M}\epsilon + \epsilon\|\boldsymbol{x}_k\|, \quad \text{E}\boldsymbol{\xi}_k\boldsymbol{\xi}_k^T \triangleq \Sigma_\xi = \sigma_d^2(I+R).$$

The last term we need to evaluate is $\text{fx}[(I-\mu\alpha_c\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T)\boldsymbol{z}_k^c]$. If $\alpha_c(z_{k,\bar{k}}^c) = 0$, $\boldsymbol{z}_k^c$ is not modified, and there is no error. On the other hand, if $|z_{k,\bar{k}}^c| \ge C_1$, we have

$$\text{fx}[(I-\mu\alpha_c\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T)\boldsymbol{z}_k^c] = (I-\mu\rho\alpha_0\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T)\boldsymbol{z}_k^c + \zeta(k)\boldsymbol{e}_{\bar{k}}$$

where $|\zeta(k)| \le \epsilon$, $\text{E}\zeta(k)^2 = \sigma_d^2$, and

$$\rho = \begin{cases} 1, & \text{if } |z_{k,\bar{k}}| \ge C_2 \\ \dfrac{1}{\mu\alpha_0}\,\text{fx}[\mu\alpha_c(z_{k,\bar{k}})], & \text{if } C_1 < |z_{k,\bar{k}}| < C_2. \end{cases}$$

In general, it can be shown that ρ satisfies $0 \le \rho < 1 + 3\epsilon$. However, it turns out that the error incurred in computing ρ does not affect our analysis in an important way; therefore, in the following, we will assume that this error is zero.

The combination of all finite-precision errors is denoted by $\boldsymbol{\delta}_k^c$, i.e.,

$$\boldsymbol{\delta}_k^c \triangleq \boldsymbol{\xi}_k + \zeta(k)\boldsymbol{e}_{\bar{k}} + \mu\eta(k)\boldsymbol{x}_k.$$

From our assumptions, it follows that $\boldsymbol{\delta}_k^c$ satisfies

$$\|\boldsymbol{\delta}_k^c\| \le (\sqrt{M} + (1+\mu\sqrt{M})\|\boldsymbol{x}_k\| + 1(\alpha_c \ne 0))\epsilon \quad \text{(A.2)}$$
$$\text{E}\boldsymbol{\delta}_k^c\boldsymbol{\delta}_k^{cT} = \Sigma_\xi + \mu\sigma_d^2 R + 1(\alpha_c \ne 0)\sigma_d^2\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T, \quad \text{(A.3)}$$
$$|\delta_{k,j}^c| \le (2 + (1+\mu)\|\boldsymbol{x}_k\|_\infty)\epsilon \quad \text{(A.4)}$$

where $1(\alpha_c \ne 0) = 1$ if $\alpha_c \ne 0$ and zero otherwise. In addition, the last equation provides bounds for the individual elements of $\boldsymbol{\delta}_k^c$.

With these definitions, we can write the finite-precision update law for circular leaky as

$$\boldsymbol{z}_{k+1}^c = \boldsymbol{z}_k + \mu\boldsymbol{x}_k e^c(k) - \mu\alpha_c\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T\boldsymbol{z}_k + \boldsymbol{\delta}_k^c. \quad \text{(A.5)}$$

[8] If the multiplications are computed in double precision and only the final result is rounded to B bits, then $|\eta(k)| \le \epsilon$ and $\sigma_\eta^2 = \sigma_d^2$.

Modified Switching-σ: The update law for the switching-σ algorithm is obtained by following a similar procedure. The result is

$$\boldsymbol{z}_{k+1}^s = ((1-\mu\alpha_s(\boldsymbol{z}_k^s))I - \mu\boldsymbol{x}_k\boldsymbol{x}_k^T)\boldsymbol{z}_k^s + \mu\boldsymbol{x}_k\boldsymbol{x}_k^T\boldsymbol{w}_* + \mu\boldsymbol{x}_k v(k) + \boldsymbol{\delta}_k^s. \quad \text{(A.6)}$$

The only difference is in the term $\boldsymbol{\delta}_k^s$, which satisfies

$$\|\boldsymbol{\delta}_k^s\| \le (\sqrt{M} + (1+\mu\sqrt{M})\|\boldsymbol{x}_k\| + 1(\alpha_s \ne 0)\sqrt{M})\epsilon$$
$$\text{E}\boldsymbol{\delta}_k^s\boldsymbol{\delta}_k^{sT} = \Sigma_\xi + \mu\sigma_d^2 R + 1(\alpha_s \ne 0)\sigma_d^2 I$$
$$|\delta_{k,j}^s| \le (1 + (1+\mu)\|\boldsymbol{x}_k\|_\infty + 1(\alpha_s \ne 0))\epsilon.$$

Appendix B
Averaged System for Circular-Leaky Algorithm

As mentioned in Section IV-B, in order to apply Theorem 1, we need to show that the fixed-point circular-leaky error equation (16), and its averaged counterparts, satisfy the conditions i)–iii) given in the statement of the theorem. Dropping the superscript c from the averaged variables for ease of notation, the averaged error equation is [cf. (17)]

$$\tilde{\boldsymbol{z}}_{k+1}^{av} = (I - \mu\alpha_c(w_{*,\bar{k}} - \tilde{z}_{k,\bar{k}}^{av})\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T - \mu R)\tilde{\boldsymbol{z}}_k^{av} + \mu\alpha_c(w_{*,\bar{k}} - \tilde{z}_{k,\bar{k}}^{av})\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T\boldsymbol{w}_*. \quad \text{(B.1)}$$

Conditions ii) and iii) follow from the above recursion and from the definition of $\alpha_c(\cdot)$. In fact

$$f_{av}(\tilde{\boldsymbol{z}}_k^{av}) = -R\tilde{\boldsymbol{z}}_k^{av} - \alpha_c(w_{*,\bar{k}}\tilde{z}_{k,\bar{k}}^{av})\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T\tilde{\boldsymbol{z}}_k^{av} + \alpha_c(w_{*,\bar{k}} - \tilde{z}_{k,\bar{k}}^{av})\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T\boldsymbol{w}_*.$$

From the definition of $\alpha_c(\cdot)$, we obtain

$$\frac{\partial\alpha_c(a)}{\partial a} = \begin{cases} \dfrac{\alpha_0}{2}\left(\dfrac{a-C_1}{D}\right), & \text{if } C_1 < a < C_1 + D \\ -\dfrac{\alpha_0}{2}\left(\dfrac{C_2-a}{D}\right), & \text{if } C_1 + D \le a < C_2 \\ 0, & \text{otherwise} \end{cases}$$

and thus, the gradient of f_{av} is

$$\nabla_{\tilde{\boldsymbol{z}}} f_{av} = -R - \alpha_c(\boldsymbol{w}_{*,\bar{k}} - \tilde{z}_{k,\bar{k}}^c)\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T - \frac{\partial\alpha_c}{\partial\tilde{z}_{k,\bar{k}}^c}\,\tilde{\boldsymbol{z}}_k^c\boldsymbol{e}_{\bar{k}}^T.$$

From this relation, we conclude that condition ii) is satisfied. We now compute $\nabla_{\tilde{\boldsymbol{z}}} f$

$$\nabla_{\tilde{\boldsymbol{z}}} f = -\boldsymbol{x}_k\boldsymbol{x}_k^T - \alpha_c(\boldsymbol{w}_{*,\bar{k}} - \tilde{z}_{k,\bar{k}}^c)\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T - \frac{\partial\alpha_c}{\partial\tilde{z}_{k,\bar{k}}^c}\,\tilde{\boldsymbol{z}}_k^c\boldsymbol{e}_{\bar{k}}^T.$$

Condition iii) follows from this relation and the fact that both $\alpha_c(\cdot)$ and its derivative are continuous and bounded functions.

We still need to check condition i) before we can use Theorem 1, i.e., we need to prove that the origin $\tilde{\boldsymbol{z}}_k^{av} = \mathbf{0}$ is an exponentially stable equilibrium point of (17). Note first that $\mathbf{0}$ is an equilibrium point of (17) since, by definition, $\alpha_c(w_{*,\bar{k}}, 0) = 0$. To prove that it is exponentially stable, we shall proceed by showing that $\|\tilde{\boldsymbol{z}}_{k+1}^{av}\| \le \gamma\|\tilde{\boldsymbol{z}}_k^{av}\|$ for all $k \ge 0$ and for some $\gamma < 1$.

Before we evaluate the norm of $\tilde{\boldsymbol{z}}_{k+1}^{av}$, we need to relate the term $\mu\alpha_c(w_{*,\bar{k}} - \tilde{z}_{k,\bar{k}}^{av})\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T\boldsymbol{w}_*$ in (B.1) to $\tilde{\boldsymbol{z}}_k$. We do so by relating $\boldsymbol{w}_*$ to $\tilde{\boldsymbol{z}}_k$ as follows.

Let the constant $\eta_c > 0$ be such that C_1 satisfies

$$C_1 = (1+\eta_c)\|\boldsymbol{w}_*\|_\infty. \tag{B.2}$$

Recall that the leakage term is nonzero if and only if $|w_{*,\bar{k}} - \tilde{z}^{av}_{k,\bar{k}}| > C_1$. Therefore, if $w_{*,\bar{k}} \cdot \tilde{z}^{av}_{k,\bar{k}} \geq 0$, $\alpha_c(w_{*,\bar{k}} - \tilde{z}^{av}_{k,\bar{k}}) \neq 0$ implies that

$$C_1 = (1+\eta_c)\|\boldsymbol{w}_*\|_\infty < |w_{*,\bar{k}} - \tilde{z}^{av}_{k,\bar{k}}| = |\tilde{z}^{av}_{k,\bar{k}}| - |w_{*,\bar{k}}|$$

and thus, $\alpha_c \neq 0$ implies that $|\tilde{z}^{av}_{k,\bar{k}}| > (1+\eta_c)\|\boldsymbol{w}_*\|_\infty$ (if $\tilde{z}^{av}_{k,\bar{k}}$ and $w_{*,\bar{k}}$ have the same sign). Repeating the argument for $w_{*,\bar{k}} \cdot \tilde{z}^{av}_{k,\bar{k}} < 0$, we conclude that

$$\alpha_c(w_{*,\bar{k}} - \tilde{z}^{av}_{k,\bar{k}}) \neq 0 \Rightarrow \begin{cases} |w_{*,\bar{k}}| < \dfrac{1}{1+\eta_c}\,|\tilde{z}^{av}_{k,\bar{k}}|, & \text{if } w_{*,\bar{k}} \cdot \tilde{z}^{av}_{k,\bar{k}} \geq 0 \\ |w_{*,\bar{k}}| < \dfrac{1}{\eta_c}\,|\bar{z}^{c}_{k,\bar{k}}|, & \text{if } w_{*,\bar{k}} \cdot \tilde{z}^{av}_{k,\bar{k}} < 0. \end{cases} \tag{B.3}$$

This allows us to express $w_{*,\bar{k}}$ as $\epsilon_k \tilde{z}^{av}_{k,\bar{k}}$ for some ϵ_k in the interval $-(1/1+\eta_c) < \epsilon_k < (1/\eta_c)$. Using this result in (17), we obtain

$$\tilde{\boldsymbol{z}}^{av}_{k+1} = (I - \mu(1+\epsilon_k)\alpha_c(w_{*,\bar{k}} - \tilde{z}^{av}_{k,\bar{k}})\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T - \mu R)\tilde{\boldsymbol{z}}^{av}_k.$$

Introduce the coefficient matrix $A(\epsilon_k) = I - \mu(1+\epsilon_k)\alpha_c(\cdot)\boldsymbol{e}_{\bar{k}}\boldsymbol{e}_{\bar{k}}^T - \mu R$. We now show that $A(\epsilon_k)$ is uniformly contractive, i.e., $\|A(\epsilon_k)\| \leq \gamma < 1$ for all k (note that since A is symmetric, $\|A\| = |\lambda_{\max}(A)|$, and we can thus show instead that the eigenvalues of $A(\epsilon_k)$ are uniformly upper bounded by 1).

Therefore, let $\boldsymbol{y}$ be a vector with unit norm, and compute

$$\boldsymbol{y}^T A(\epsilon_k)\boldsymbol{y} = \boldsymbol{y}^T(I-\mu R)\boldsymbol{y} - \mu(1+\epsilon_k)\alpha_c(\cdot)(\boldsymbol{y}^T\boldsymbol{e}_{\bar{k}})^2.$$

If $\alpha_c = 0$, $A(\epsilon_k) = I - \mu R$, and $|\lambda(A)| < 1$ if and only if

$$\mu\lambda_{\min}(R) > 0 \quad \text{and} \quad \mu\lambda_{\max}(R) < 2$$

which are the usual conditions for the mean-square stability of LMS. On the other hand, if $\alpha_c = \rho\alpha_0$, for some $0 \leq \rho \leq 1$, we have

$$1 - \mu\lambda_{\max}(R) - \mu\rho\left(1 + \frac{1}{\eta_c}\right)\alpha_0 < \boldsymbol{y}^T A(\epsilon_k)\boldsymbol{y} < 1 - \mu\lambda_{\min}(R) - \mu\rho\left(1 - \frac{1}{1+\eta_c}\right)\alpha_0.$$

Therefore, $|\lambda[A(\epsilon_k)]| < 1$ if

$$\mu\lambda_{\min}(R) + \mu\left(1 - \frac{1}{1+\eta_c}\right)\alpha_0 > 0$$

and

$$\mu\left(1 + \frac{1}{\eta_c}\right)\alpha_0 < 2 - \mu\lambda_{\max}(R). \tag{B.4}$$

The first of these conditions is always satisfied since, by assumption, $\eta_c > 0$ and $\lambda_{\min}(R) > 0$. The second condition provides an upper bound on α_0 as a function of our choices for μ and η_c (or C_1). In this case, we obtain $\|A(\epsilon_k)\| \stackrel{\Delta}{=} \gamma < 1$. It then follows that $\|\bar{z}^c_{k+1}\| < \|A(\epsilon_k)\|\,\|\bar{z}^c_k\| \leq \gamma\|\bar{z}^c_k\|$, and thus, $\tilde{z}^{av}_k = \boldsymbol{0}$ is an exponentially stable equilibrium point of (17).

With this result, we can apply Theorem 1 and conclude that the steady state of circular-leaky can be obtained from the partially averaged recursion

$$\tilde{\boldsymbol{z}}^{pav}_{k+1} = (I-\mu R)\tilde{\boldsymbol{z}}^{pav}_k - \mu\boldsymbol{x}_k v(k) - \boldsymbol{\delta}^c_k.$$

For the MSE, we use the fact from Theorem 1 that

$$\sup_{k\geq 0} \mathrm{E}\|\tilde{\boldsymbol{z}}^c_k - \tilde{\boldsymbol{z}}^{pav}_k\|^2 \to 0 \tag{B.5}$$

as $\mu \to 0$. The computation of the MSE will be performed in several steps. First, note that $e(k) = \boldsymbol{x}_k^T\tilde{\boldsymbol{z}}^c_k + v(k)$, and from the independence of $\{v(k)\}$ and $\{\boldsymbol{x}_k\}$, we obtain

$$\mathrm{E}e(k)^2 = \mathrm{E}(\boldsymbol{x}_k\tilde{\boldsymbol{z}}^c_k)^2 + \sigma_v^2. \tag{B.6}$$

To compute $\mathrm{E}(\boldsymbol{x}_k\tilde{\boldsymbol{z}}^c_k)^2$, we show that (in steady state) this expectation is equal to $\mathrm{E}(\boldsymbol{x}_k\tilde{\boldsymbol{z}}_k)^2$, where $\tilde{\boldsymbol{z}}_k$ is the weight error obtained from the LMS algorithm with the same input and noise sequences.

This is shown as follows. Notice that the above recursion for $\tilde{\boldsymbol{z}}^{pav}_k$, which is obtained for circular-leaky, is the same partially averaged recursion that would be obtained for LMS. Therefore, Theorem 1 also implies that

$$\sup_{k\geq 0} \mathrm{E}\|\tilde{\boldsymbol{z}}_k - \tilde{\boldsymbol{z}}^{pav}_k\|^2 \to 0$$

as $\mu \to 0$, where now, $\tilde{\boldsymbol{z}}_k$ is the weight error computed by LMS. From this relation and (B.5), we conclude that

$$\sup_{k\geq 0} \mathrm{E}\|\tilde{\boldsymbol{z}}_k - \tilde{\boldsymbol{z}}^c_k\|^2 \to 0$$

as $\mu \to 0$.

Next, it can be verified that

$$\begin{aligned}(\boldsymbol{x}_k^T\tilde{\boldsymbol{z}}^c_k)^2 - (\boldsymbol{x}_k^T\tilde{\boldsymbol{z}}^{pav}_k)^2 &\leq \|\tilde{\boldsymbol{z}}^c_k\tilde{\boldsymbol{z}}^{cT}_k - \tilde{\boldsymbol{z}}^{pav}_k\tilde{\boldsymbol{z}}^{pav\,T}_k\|B_x, \\ &\leq (\|\tilde{\boldsymbol{z}}^c_k - \tilde{\boldsymbol{z}}^{pav}_k\|^2 + 2\|\tilde{\boldsymbol{z}}^{pav}_k\|\,\|\tilde{\boldsymbol{z}}^c_k - \tilde{\boldsymbol{z}}^{pav}_k\|)B_x.\end{aligned}$$

Now, since $\boldsymbol{x}_k$ and $v(k)$ are bounded, $\|\tilde{\boldsymbol{z}}^{pav}_k\|$ is also a bounded sequence for small μ. This fact and the general inequality for random variables $(\mathrm{E}a)^2 \leq \mathrm{E}a^2$ imply that

$$\lim_{k\to\infty}\sup_{k\geq 0} \mathrm{E}[(\boldsymbol{x}_k^T\tilde{\boldsymbol{z}}^c_k)^2 - (\boldsymbol{x}_k^T\tilde{\boldsymbol{z}}^{pav}_k)^2] \leq 0$$

as $\mu \to 0$. Similar arguments show that

$$\lim_{k\to\infty}\sup_{k\geq 0} -\mathrm{E}[(\boldsymbol{x}_k^T\tilde{\boldsymbol{z}}^c_k)^2 - (\boldsymbol{x}_k^T\tilde{\boldsymbol{z}}^{pav}_k)^2] \leq 0.$$

We conclude that, for small μ and in steady-state, $\mathrm{E}(\boldsymbol{x}_k^T\tilde{\boldsymbol{z}}^c_k)^2 \approx \mathrm{E}(\boldsymbol{x}_k^T\tilde{\boldsymbol{z}}^{pav}_k)^2$. We can repeat this argument, replacing $\tilde{\boldsymbol{z}}^c_k$ with $\tilde{\boldsymbol{z}}_k$ to conclude that, in the steady state

$$\mathrm{E}(\boldsymbol{x}_k^T\tilde{\boldsymbol{z}}^c_k)^2 \approx \mathrm{E}(\boldsymbol{x}_k^T\tilde{\boldsymbol{z}}^{pav}_k)^2 \approx \mathrm{E}(\boldsymbol{x}_k^T\tilde{\boldsymbol{z}}_k)^2. \tag{B.7}$$

The quantity on the right-hand side is the value obtained from the LMS algorithm. We can use the following argument to obtain the MSE formula in (19) [23].

Let the covariance of $\tilde{z}_k$ be $\bar{Z}_k \triangleq \mathrm{E}\tilde{z}_k\tilde{z}_k^T$. We know from the above arguments that this covariance reaches a steady state so that we can write

$$\mathrm{Tr}(\bar{Z}_\infty) = \lim_{k\to\infty} \{\mathrm{Tr}(\mathrm{E}[(I-\mu \boldsymbol{x}_k\boldsymbol{x}_k^T)\tilde{\boldsymbol{z}}_k\tilde{\boldsymbol{z}}_k^T(I-\mu \boldsymbol{x}_k\boldsymbol{x}_k^T)]) + \mu^2\sigma_v^2\mathrm{E}\|\boldsymbol{x}_k\|^2 + \mathrm{Tr}(\Sigma_\xi + \mu^2\sigma_d^2 MR)\}$$

where we already ignored the cross-terms. Expanding the first term in the right-hand side, we obtain

$$\lim_{k\to\infty} \mathrm{E}(\boldsymbol{x}_k^T\tilde{\boldsymbol{z}}_k)^2 = \frac{1}{2}\left\{\mu(\sigma_v^2 + M\sigma_d^2)\mathrm{Tr}(R) + \frac{1}{2\mu}\mathrm{Tr}(\Sigma_\xi)\right\}.$$

Combining this result with (B.6) and (B.7), we obtain the MSE formula (19) in the statement of the theorem.

APPENDIX C
AVERAGED SYSTEM FOR SWITCHING-σ

We need to show that in the steady state, the averaged variable $\boldsymbol{z}_k^{s,av}$ has norm less than S_1 so that the leakage term remains equal to zero. To do so, we compare the averaged recursion (21) with the averaged LMS recursion

$$\boldsymbol{z}_{k+1}^{av} = (I-\mu R)\boldsymbol{z}_k^{av} + \mu R\boldsymbol{w}_*.$$

Note that if $|\lambda(I-\mu R)| < 1$, this recursion satisfies

$$\lim_{k\to\infty} \boldsymbol{z}_k^{av} = \boldsymbol{w}_*. \tag{C.1}$$

On the other hand, expanding $\boldsymbol{z}_{k+1}^{av}$, we obtain

$$\boldsymbol{z}_{k+1}^{av} = \prod_{i=0}^{k}[I-\mu R]\boldsymbol{z}_0^{av} + \sum_{i=0}^{k}\prod_{j=i+1}^{k}[I-\mu R]\mu R\boldsymbol{w}_*.$$

Since the first term in the above relation tends to zero, (C.1) implies that

$$\lim_{k\to\infty}\left(\sum_{i=0}^{k}\prod_{j=i+1}^{k}[I-\mu R]\right)\mu R = I$$

and therefore, the relation below holds for any vector $\boldsymbol{y}$ with unit norm

$$\lim_{k\to\infty} \boldsymbol{y}^T\left(\sum_{i=0}^{k}\prod_{j=i+1}^{k}[I-\mu R]\right)\mu R\boldsymbol{y} = 1. \tag{C.2}$$

We will now rewrite this expression in a more adequate form. Let $R^{1/2}$ be a symmetric square-root factor of R, i.e., $(R^{1/2})^2 = R$, $(R^{1/2})^T = R^{1/2}$. We can rewrite each term in the above sum as

$$\left(\prod_{j=i+1}^{k}[I-\mu R]\right)\mu R = \mu R^{1/2}\left(\prod_{j=i+1}^{k}[I-\mu R]\right)R^{1/2} \tag{C.3}$$

From (C.3) and (C.2), we obtain

$$\lim_{k\to\infty} \mu\sum_{i=0}^{k} \boldsymbol{y}^T R^{1/2}\left(\prod_{j=i+1}^{k}[I-\mu R]\right)R^{1/2}\boldsymbol{y} \triangleq \mu\sum_{i=0}^{k}\boldsymbol{y}^T B_i\boldsymbol{y} = 1 \tag{C.4}$$

where we have defined the matrices B_i. Assuming that $0 < \mu R < I$, all the B_i are positive-definite, and thus, all the terms in the above sum are positive. We shall use this result soon.

Returning to the switching-σ algorithm, we have

$$\boldsymbol{z}_{k+1}^{s,av} = \prod_{i=0}^{k}[(1-\mu\alpha_s)I-\mu R]\boldsymbol{z}_0^{s,av} + \sum_{i=0}^{k}\prod_{j=i+1}^{k}[(1-\mu\alpha_s)I-\mu R]\mu R\boldsymbol{w}_*.$$

If $-1 < 1-\mu\alpha_0-\lambda_i(R) < 1$, the term in $\boldsymbol{z}_0^{s,av}$ will tend to zero, and therefore

$$\lim_{k\to\infty}\boldsymbol{z}_{k+1}^{s,av} = \lim_{k\to\infty}\sum_{i=0}^{k}\prod_{j=i+1}^{k}[(1-\alpha_s)I-\mu R]\mu R\boldsymbol{w}_*. \tag{C.5}$$

Taking norms on both sides, we obtain

$$\|\tilde{\boldsymbol{z}}_\infty^{av}\| \le \left\|\sum_{i=0}^{k}\prod_{j=i+1}^{k}[(1-\alpha_s)I-\mu R]\mu R\right\|\,\|\boldsymbol{w}_*\|.$$

We will now use (C.4) to show that the above matrix norm is not greater than one, from which we can conclude that $\|\tilde{\boldsymbol{z}}_k^{av}\| \le \|\boldsymbol{w}_*\| < S_1$ for large enough k.

Similarly to what we did in (C.3), we can write

$$\prod_{j=i+1}^{k}[(1-\mu\alpha_s)I-\mu R]\mu R = \mu R^{1/2}\left(\prod_{j=i+1}^{k}[(1-\mu\alpha_s)I-\mu R]\right)R^{1/2}.$$

Assume that $1-\mu\lambda_{\max}(R) > 0$, and choose α_0 such that $|1-\mu\alpha_0-\mu\lambda_i(R)| < 1-\mu\lambda_i(R) < 1$, i.e.,

$$0 < \alpha_0 < 2-2\mu\lambda_{\max}(R). \tag{C.6}$$

If this condition is satisfied, then for all k

$$-B_i \le \mu R^{1/2}\left(\prod_{j=i+1}^{k}[(1-\mu\alpha_s)I-\mu R]\right)R^{1/2} \le B_i.$$

Let $\boldsymbol{y}$ be a unit-norm vector. From the above relations, we conclude that

$$\boldsymbol{y}^T B_i\boldsymbol{y} \ge \left|\mu\boldsymbol{y}^T R^{1/2}\left(\prod_{j=i+1}^{k}[(1-\mu\alpha_s)I-\mu R]\right)R^{1/2}\boldsymbol{y}\right|.$$

Therefore, we have the bound

$$\left\|\sum_{i=0}^{\infty}\prod_{j=i+1}^{\infty}[(1-\mu\alpha_s)I-\mu R]\mu R\right\| \leq \max_{\|\boldsymbol{y}\|=1}\sum_{i=0}^{\infty}\boldsymbol{y}^T B_i \boldsymbol{y} = 1.$$

This relation and (C.5) imply that there is a K such that for all $k \geq K$

$$\lim_{k\to\infty}\|\boldsymbol{z}_k^{s,av}\| \leq \|\boldsymbol{w}_*\| < S_1$$

completing our proof.

APPENDIX D
PROOF OF THEOREM 4

From Appendix A, the fixed-point recursion for the switching-σ algorithm is

$$\boldsymbol{z}_{k+1}^s = ((1-\mu\alpha_s(\cdot))I-\mu\boldsymbol{x}_k\boldsymbol{x}_k^T)\boldsymbol{z}_k^s + \mu\boldsymbol{x}_k\boldsymbol{x}_k^T\boldsymbol{w}_* + \mu\boldsymbol{x}_k v(k) + \boldsymbol{\delta}_k^s. \tag{D.1}$$

Our goal is to show that the sequence $\{\|\boldsymbol{z}_k^s\|\}_{k=0}^{\infty}$ is bounded. The first task in the proof is to find a ball $\mathcal{B}$ (centered at the origin) outside of which the norm $\|\boldsymbol{z}_k^s\|$ is strictly decreasing, i.e.,

$$\|\boldsymbol{z}_{k+1}^s\| - \|\boldsymbol{z}_k^s\| < 0, \qquad \text{if } \boldsymbol{z}_k^s \notin \mathcal{B}.$$

We do not need to find the smallest ball satisfying the above property to prove that $\{\|\boldsymbol{z}_k^s\|\}_{k=0}^{\infty}$ is bounded; we only need to find one such ball. With this in mind, our argument is simplified if we restrict ourselves to balls $\mathcal{B}_r$ with radii $r > S_2$ so that $\alpha_s(\boldsymbol{z}) = \alpha_0$ for any $\boldsymbol{z} \notin \mathcal{B}_r$.

Assume then that $\boldsymbol{z}_k^s$ satisfies $\|\boldsymbol{z}_k^s\| > S_2$ at some instant k. Taking norms of both sides of (D.1) and using (22) to upper bound $\|[(1-\mu\alpha_0)I-\mu\boldsymbol{x}_k\boldsymbol{x}_k^T]\|$, we obtain the inequality

$$\|\boldsymbol{z}_{k+1}^s\| \leq |1-\mu\alpha_0|\|\boldsymbol{z}_k^s\| + \mu\beta\|\boldsymbol{w}_*\| + \mu\sqrt{\beta}v_{\max} + ((2+\mu\alpha_0)+\mu\sqrt{\beta})\sqrt{M}\epsilon \tag{D.2}$$

where we also used (2) to bound $v(k)$ and $\boldsymbol{x}_k$ and the bound for $\|\boldsymbol{\delta}_k^s\|$ from Appendix A. Subtracting $\|\boldsymbol{z}_k^s\|$ from both sides of (D.2), we obtain

$$\|\boldsymbol{z}_{k+1}^s\| - \|\boldsymbol{z}_k^s\| \leq -\mu\alpha_0\|\boldsymbol{z}_k^s\| + \mu\beta\|\boldsymbol{w}_*\| + \sqrt{\beta}v_{\max} + ((2+\mu\alpha_0)+\mu\sqrt{\beta})\sqrt{M}\epsilon.$$

From this inequality, and from our assumption that $\|\boldsymbol{z}_k^s\| > S_2$, it follows that $\|\boldsymbol{z}_{k+1}^s\| < \|\boldsymbol{z}_k^s\|$ if we have (D.3), shown at the bottom of the page. We therefore can choose $\mathcal{B} = \{\boldsymbol{z}: \|\boldsymbol{z}\| \leq \Omega\}$. To complete our argument, note the following.

1) We may have $\|\boldsymbol{z}_{n+1}^s\| \geq \|\boldsymbol{z}_n^s\|$ only if $\boldsymbol{z}_n^s \in \mathcal{B}$. However, $\|\boldsymbol{z}_{n+1}^s\|$ cannot be arbitrarily large. In fact, using the switching-σ recursion (D.1), we can evaluate the worst-case $\|\boldsymbol{z}_{n+1}^s\|$ (the bound below is not tight)

$$\sup_{\boldsymbol{z}_n^s\in\mathcal{B}}\|\boldsymbol{z}_{n+1}^s\| \leq \Omega + \mu\beta\|\boldsymbol{w}_*\| + \mu\sqrt{\beta}v_{\max} + ((2+\mu\alpha_0)+\mu\sqrt{\beta})\sqrt{M}\epsilon. \tag{D.4}$$

2) If $\boldsymbol{z}_n^s$ is not inside $\mathcal{B}$ at a particular time instant n (i.e., $\|\boldsymbol{z}_n^s\| > \Omega$), then $\|\boldsymbol{z}_{n+1}^s\| < |\boldsymbol{z}_n^s\|$. Repeating this argument, we conclude that either $\|\boldsymbol{z}_k^s\| < \|\boldsymbol{z}_n^s\|$ for all $k > n$ or there exists a time (say, N) such that $\boldsymbol{z}_{n+N}^s \in \mathcal{B}$.

The result of the theorem follows from these two observations.

APPENDIX E
PROOF OF THEOREM 5

The variable $\boldsymbol{z}_{k+M}^c$ can be shown to satisfy

$$\boldsymbol{z}_{k+M}^c = \prod_{l=k}^{k+M-1}(I-\mu\alpha_c\boldsymbol{e}_{\bar{l}}\boldsymbol{e}_{\bar{l}}^T-\mu\boldsymbol{x}_l\boldsymbol{x}_l^T)\boldsymbol{z}_k^c + \sum_{i=k}^{k+M-1}\left[\prod_{l=i}^{k+M-1}(I-\mu\alpha_c\boldsymbol{e}_{\bar{l}}\boldsymbol{e}_{\bar{l}}^T-\mu\boldsymbol{x}_l\boldsymbol{x}_l^T) \cdot (\mu\boldsymbol{x}_i\boldsymbol{x}_i^T\boldsymbol{w}_* - \mu\boldsymbol{x}_i v(i) - \boldsymbol{\delta}_i^c)\right]. \tag{E.1}$$

Given that α_0 and μ satisfy $|1-\mu\alpha_0-\mu\beta| < 1$, we find that all matrices in the expression for $\boldsymbol{z}_{k+M}^c$ above are contractive (i.e., have 2-induced norms less than or equal to 1). It follows that the second term in (E.1) is bounded from above by

$$M[\mu\beta\|\boldsymbol{w}_*\| + \sqrt{\mu\beta}v_{\max} + \|\boldsymbol{\delta}_i^c\|]. \tag{E.2}$$

We also need to bound the norm of the first term of (E.1). Define

$$A_i = I - \mu\alpha_c\boldsymbol{e}_{\bar{i}}\boldsymbol{e}_{\bar{i}}^T, \quad B_i = \mu\boldsymbol{x}_i\boldsymbol{x}_i^T$$

and note that $\|B_i\| \leq \mu\beta$. In addition, if $|1-\mu\alpha_0-\mu\beta| < 1$, we have

$$\|A_i - B_i\| \leq 1. \tag{E.3}$$

With these definitions, the product we want to bound is

$$\prod_{i=k}^{k+M-1}(A_i - B_i) = \prod_{i=k}^{k+M-1}A_i - \sum_{i=k}^{k+M-1}\left[\prod_{j=i+1}^{k+M-1}(A_j-B_j)\right]B_i\left[\prod_{l=k}^{i-1}(A_l-B_l)\right].$$

$$|\boldsymbol{z}_k^s\| > \Omega \triangleq \max\left\{S_2, \frac{\mu\beta\|\boldsymbol{w}_*\| + \mu\sqrt{\beta}v_{\max} + ((2+\mu\alpha_0)+\mu\sqrt{\beta})\sqrt{M}\epsilon}{\mu\alpha_0}\right\}. \tag{D.3}$$

Consider the second term. From (E.3), its norm is bounded by

$$\left\| \sum_{i=k}^{k+M-1} \left[\prod_{j=i+1}^{k+M-1} (A_j - B_j) \right] B_i \left[\prod_{l=k}^{i-1} (A_l - B_l) \right] \right\| \le \sum_{i=k}^{k+M-1} \|B_i\| \le M\mu\beta.$$

To approximate the first term, note that $e_{\bar{i}}^T e_{\bar{j}} = 0$ if $0 < |i - j| < M$; thus

$$\prod_{i=k}^{k+M-1} A_i = I - \sum_{i=k}^{k+M-1} e_{\bar{i}} e_{\bar{i}}^T.$$

Now, let $z_{k,m}^c$ denote the entry of z_k^c that has the largest absolute value ($|z_{k,m}^c| = \|z_k^c\|_\infty$). Let $k + l$ be such that $\overline{k+l} = m$, and assume for now that $\alpha_c(z_{k+l,m}^c) = \alpha_0$, so that

$$\begin{aligned} \left\| \left[I - \sum_{i=k}^{k+M-1} \alpha_c(z_{i,\bar{i}}^c) e_{\bar{i}} e_{\bar{i}}^T \right] z_k^c \right\| &= \left\| z_k^c - \sum_{i=k}^{k+M-1} \alpha_c(z_{i,\bar{i}}^c) e_{\bar{i}} z_{k,\bar{i}}^c \right\| \\ &\le \left[\sum_{\substack{i=1 \\ i \ne m}}^{M} z_{k,i}^{c2} + (1-\alpha_0)^2 z_{k,m}^{c2} \right]^{1/2} \\ &\le \sqrt{1 - \frac{\alpha_0(2-\alpha_0)}{M}} \, \|z_k^c\|. \end{aligned} \quad \text{(E.4)}$$

Putting all these results together, we obtain

$$\left\| \prod_{i=k}^{k+M-1} (A_i - B_i) z_k^c \right\| \le \left[\sqrt{1 - \frac{\mu\alpha_0(2-\mu\alpha_0)}{M}} + \mu M \beta \right] \|z_k^c\|. \quad \text{(E.5)}$$

Assume that $\mu\alpha_0 > \mu\beta$ such that

$$\sqrt{1 - \frac{\mu\alpha_0(2-\mu\alpha_0)}{M}} + \mu M\beta < 1.$$

We still need to show that if $\|z_k^c\|$ is large enough, then $|z_{k+l,m}^c| > C_2$, and $\alpha_c(z_{k+l,m}^c) = \alpha_0$. The expression for $z_{k+l,m}^c = e_m^T z_{k+l}^c$ is

$$\begin{aligned} z_{k+l,m}^c = e_m^T & \prod_{l=k}^{k+l-1} (I - \mu\alpha_c e_{\bar{l}} e_{\bar{l}}^T - \mu x_l x_l^T) z_k^c \\ & + e_m^T \sum_{i=k}^{k+l-1} \left[\prod_{j=i}^{k+l-1} (I - \mu\alpha_c e_{\bar{j}} e_{\bar{j}}^T - \mu x_j x_j^T) \right. \\ & \left. \cdot (\mu x_i x_i^T w_* - \mu x_i v(i) - \delta_i^c) \right]. \end{aligned}$$

Note that $e_m^T e_{\bar{i}} = 0$ for $k \le i \le k+l-1$, and therefore, using again the decomposition for $\Pi(A_i - B_i)$, we conclude that

$$\begin{aligned} |z_{k+l,m}^c| \ge |z_{k,m}^c| - \mu M\beta \|z_k^c\| - M(\mu\beta\|w_*\| \\ + \sqrt{\mu\beta} v_{\max} + \|\delta^c\|) > C_2 \end{aligned}$$

if

$$\begin{aligned} \|z_k^c\|_\infty = |z_{k,m}^c| > C_2 + \mu M\beta \|z_k^c\| \\ + M(\mu\beta\|w_*\| + \sqrt{\mu\beta} v_{\max} \\ + (\sqrt{M} + (1 + \mu\sqrt{M})\|x_k\| + 1)\epsilon). \end{aligned}$$

Since $\|z_k^c\| \le \sqrt{M}\|z_k^c\|_\infty$, it follows from the above inequality that the leakage term will be equal to α_0 at time $k + l$ (where $\overline{k+l} = m$) if

$$\|z_k^c\| \ge \frac{\sqrt{M}(C_2 + M(\mu\beta\|w_*\| + \sqrt{\mu\beta} v_{\max} + (\sqrt{M} + (1+\mu\sqrt{M})\|x_k\| + 1)\epsilon)}{1 - \mu M^{3/2}\beta} \triangleq \Omega_1^c.$$

If the above condition holds, using (E.5) and (E.2), we conclude that the norm $\|z_{k+M}^c\|$ will be smaller than $\|z_k^c\|$ if

$$\|z_k^c\| \ge \max \left\{ \Omega_1^c, M \frac{\mu\beta\|w_*\| + \sqrt{\mu\beta} v_{\max} + (\sqrt{M} + (1+\mu\sqrt{M})\|x_k\| + 1)\epsilon}{1 - \sqrt{1 - \frac{\mu\alpha_0(2-\mu\alpha_0)}{M}} + \mu M\beta} \right\} \triangleq \Omega_2^c.$$

Therefore, $\|z_{k+M}^c\|$ will be strictly smaller than $\|z_k^c\|$ if $\|z_k^c\| > \Omega_2^c$. From this point, we can use an argument similar to that of Theorem 4 to show that the sequence $\{\|z_k^c\|\}_{k=0}^\infty$ is bounded. $\diamondsuit$

REFERENCES

[1] R. D. Gitlin, H. C. Meadors, Jr., and S. B. Weinstein, "The tap-leakage algorithm: An algorithm for the stable operation of a digitally implemented, fractionally spaced adaptive equalizer," *Bell Syst. Tech. J.*, vol. 61, no. 8, pp. 1817–1839, Oct. 1982.

[2] D. L. Cohn and J. L. Melson, "The residual encoder: An improved ADPCM system for speech digitization," *IEEE Trans. Commun.*, vol. COMM-23, pp. 935–941, Sept. 1975.

[3] G. J. Rey, R. R. Bitmead, and C. R. Johnson, Jr., "The dynamics of bursting in simple adaptive feedback systems with leakage," *IEEE Trans. Circuits Syst.*, vol. 38, pp. 475–488, May 1991.

[4] P. A. Ioannou and P. V. Kokotović, "Instability analysis and improvement of robustness of adaptive control," *Automatica*, vol. 20, no. 5, pp. 583–594, 1984.

[5] C. L. Zahm, "Aplication of adaptive arrays to suppress strong jammers in the presence of weak signals," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-9, pp. 260–271, Mar. 1973.

[6] S. Haykin, *Adaptive Filter Theory*, 3rd ed. Englewood Cliffs, NJ: Prentice-Hall, 1996.

[7] G. Ungerboeck, "Fractional tap-spacing equalizer and consequences for clock recovery in data modems," *IEEE Trans. Commun.*, vol. COMM-24, pp. 856–864, Aug. 1974.

[8] W. A. Sethares, D. A. Lawrence, C. R. Johnson, Jr., and R. R. Bitmead, "Parameter drift in LMS adaptive filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, pp. 868–878, Aug. 1986.

[9] J. M. Cioffi, "Limited-precision effects in adaptive filtering," *IEEE Trans. Circuits Syst.*, vol. CAS-34, pp. 821–833, July 1987.

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

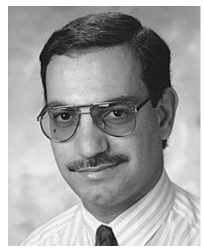
[11] K. D. Benson and W. A. Sethares, "Magnitude response peak detection and control using balanced model reduction and leakage to a target," *IEEE Trans. Signal Processing*, vol. 45, pp. 2442–2453, 1997.

[12] P. Ioannou and K. Tsakalis, "Robust discrete-time adaptive control," in K. S. Narendra, Ed., *Adaptive and Learning Systems: Theory and Applications*. New York: Plenum, 1986, pp. 73–87.

[13] P. A. Ioannou and J. Sun, *Robust Adaptive Control*. Englewood Cliffs, NJ: Prentice-Hall, 1996.
[14] J. C. M. Bermudez and N. J. Bershad, "A nonlinear analytical model for the quantized LMS algorithm—The arbitrary step size case," *IEEE Trans. Signal Processing*, vol. 44, pp. 1175–1183, May 1996.
[15] C. Caraiscos and B. Liu, "A roundoff error analysis of the LMS adaptive algorithm," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, pp. 34–41, Feb. 1984.
[16] S. T. Alexander, "Transient weight misadjustment properties for the finite precision LMS algorithm," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-35, pp. 1250–1258, Sept. 1987.
[17] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1985.
[18] O. Macchi, *Adaptive Processing—The Least Mean Squares Approach with Applications in Transmission*. West Sussex, U.K.: Wiley, 1995.
[19] V. H. Nascimento and A. H. Sayed, "An unbiased and cost-effective leaky-LMS filter," in *Proc. 30th Asilomar Conf. Signals, Syst. Comput.*, Oct. 1996, pp. 1078–1082.
[20] J. M. Cioffi and J. J. Werner, "The tap-drifting problem in digitally implemented data-driven echo cancellers," *Bell Syst. Tech. J.*, vol. 64, no. 1, pp. 115–138, Jan. 1985.
[21] V. Solo and X. Kong, *Adaptive Signal Processing Algorithms*. Englewood Cliffs, NJ: Prentice-Hall, 1995.
[22] V. Solo, "The stability of LMS," *IEEE Trans. Signal Processing*, vol. 45, pp. 3017–3026, Dec. 1997.
[23] ——, "The limiting behavior of LMS," *IEEE Trans. Signal Processing*, vol. 37, pp. 1909–1922, Dec. 1989.
[24] A. Benveniste, M. Métivier, and P. Priouret, *Adaptive Algorithms and Stochastic Approximations*. New York: Springer-Verlag, 1987.
[25] H. J. Kushner, *Approximation and Weak Convergence Methods for Random Processes, with Applications to Stochastic System Theory*. Cambridge, MA: MIT Press, 1984.
[26] R. D. Gitlin, J. E. Mazo, and M. G. Taylor, "On the design of gradient algorithms for digitally implemented adaptive filters," *IEEE Trans. Circuit Theory*, vol. CT-20, pp. 125–136, Mar. 1973.
[27] N. J. Bershad and J. C. M. Bermudez, "New insights on the transient and steady-state behavior of the quantized LMS algorithm," *IEEE Trans. Signal Processing*, vol. 44, pp. 2623–2625, Oct. 1996.
[28] E. Eweda, N. R. Yousef, and S. H. El-Ramly, "Reducing the effect of finite wordlength on the performance of an LMS adaptive filter," in *Proc. Int. Conf. Commun.*, Atlanta, GA, 1998.
[29] J. H. Wilkinson, *Rounding Errors in Algebraic Processes*. London, U.K.: Her Majesty's Stationery Office, 1963.

Vítor H. Nascimento was born in São Paulo, Brazil. He received the B.S. and M.S. degrees (with distinction) in electrical engineering from Escola Politécnica, University of São Paulo, in 1989 and 1992, respectively, and the Ph.D. degree in electrical engineering from the University of California, Los Angeles, in 1999.

He is currently an Assistant Professor with the Department of Electrical Engineering, University of São Paulo, where he also taught between 1990 and 1994. His research interests include digital signal processing, estimation, adaptive filtering, and control.



Ali H. Sayed (M'92–SM'99) received the Ph.D. degree in electrical engineering in 1992 from Stanford University, Stanford, CA.

He is currently an Associate Professor of Electrical Engineering at the University of California at Los Angeles. He has over 120 journal and conference publications, is co-author of *Indefinite Quadratic Estimation and Control* (Philadelphia, PA: SIAM, 1999) and *Linear Estimation* (Englewood Cliffs, NJ: Prentice Hall, in press), and a co-editor of the volume *Fast Reliable Algorithms for Matrices with Structure* (Philadelphia, PA: SIAM, 1999). His research interests span several areas including adaptive and statistical signal processing, linear and nonlinear filtering and estimation, linear and nonlinear system and control theories, interplays between signal processing and control methodologies, and reliable and efficient algorithms for large-scale structured computations.

Dr. Sayed is a member of the editorial board of the *SIAM Journal on Matrix Analysis and its Applications*, has served as co-editor of special issues of the journal *Linear Algebra and its Applications*, and is past associate editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING. He has contributed several articles to engineering and mathematical encyclopedias and handbooks and has served on the program committees of several international meetings. He received the 1996 IEEE Donald G. Fink Award.