An affine combination of two LMS adaptive filters - Transient mean-square analysis

Neil J. Bershad, José Carlos M. Bermudez, Jean-Yves Tourneret

An Affine Combination of Two LMS Adaptive Filters – Transient Mean-Square Analysis

Neil J. Bershad *Fellow, IEEE*, José Carlos M. Bermudez, *Senior Member, IEEE*, and Jean-Yves Tourneret, *Member, IEEE*

Abstract

This paper studies the statistical behavior of an affine combination of the outputs of two LMS adaptive filters that simultaneously adapt using the same white Gaussian inputs. The purpose of the combination is to obtain an LMS adaptive filter with fast convergence and small steady-state mean-square deviation (MSD). The linear combination studied is a generalization of the convex combination, in which the combination factor $\lambda(n)$ is restricted to the interval $(0, 1)$. The viewpoint is taken that each of the two filters produces dependent estimates of the unknown channel. Thus, there exists a sequence of optimal affine combining coefficients which minimizes the MSE. First, the optimal unrealizable affine combiner is studied and provides the best possible performance for this class. Then two new schemes are proposed for practical applications. The mean-square performances are analyzed and validated by Monte Carlo simulations. With proper design, the two practical schemes yield an overall MSD that is usually less than the MSD's of either filter.

Index Terms

Adaptive filters, analysis, affine combination, convex combination, LMS, stochastic algorithms.

EDICS Category: ASP-ANAL

I. Introduction

The design of many adaptive filters requires a trade-off between convergence speed and steady-state mean square error (MSE). A faster (slower) convergence speed yields a larger (smaller) steady-state mean-square deviation (MSD) and MSE. This property is usually independent of the type of adaptive algorithm, i.e. Least Mean-Square (LMS), Normalized Least Mean-Square (NLMS), Recursive Least Squares (RLS) or Affine Projection (AP). This design trade-off is usually controlled by some design parameter of the weight update, such as the step-size in LMS or AP, the step-size or the regularization parameter in NLMS or the forgetting factor in RLS. Variable step-size modifications of the basic adaptive algorithms offer a possible solution to this design problem [1]–[5].

Recently, a novel scheme has been proposed in [6] which uses a convex combination of two fixed step-size adaptive filters as shown in Fig. 1, where adaptive filter $\boldsymbol{W}_1(n)$ uses a larger step-size than adaptive filter $\boldsymbol{W}_2(n)$. The key to this scheme is the selection of the scalar mixing parameter $\lambda(n)$ for combining the two filter outputs. The mixing parameter is defined in [7] as a sigmoid function whose free parameter is adaptively optimized using a stochastic gradient search which minimizes the quadratic error of the overall filter. The steady-state performance

This work was partly supported by CNPq under grants No. 307024/2006-7 and 470792/2006-0.

Neil J. Bershad is with the Dept. of Electrical Engineering and Computer Science, University of California Irvine, Irvine, 1621 Santiago Drive, Newport Beach, CA 92660, U.S.A., e-mail:bershad@ece.uci.edu. José Carlos Bermudez is with the Dept. of Electrical Engineering, Federal University of Santa Catarina, Florianópolis, SC, Brazil, e-mail:j.bermudez@ieee.org. Jean-Yves Tourneret is with IRIT - ENSEEIHT - TéSA, 2 rue Camichel, BP 7122, 31071 Toulouse cedex 7, France, e-mail:Jean-Yves.Tourneret@enseeiht.fr

SP EDICS: 2-ADAPT, Corresponding author: Neil J. Bershad.

of this adaptive scheme has been recently studied in [8]. The convex combination performed as well as the best of its components in the MSE sense. These results indicate that a combination of adaptive filters can lead to fast convergence rates *and* good steady-state performance, an attribute that is usually obtained only in variable step-size algorithms. Thus, there is great interest in learning more about the properties of such adaptive structures.

This paper provides new results for the performance of the combined structure which supplement the work presently available in the literature. The achievable performance is studied for an affine combination of two LMS adaptive filters using the structure shown in Fig. 1 with stationary signals. Here, the combination parameter $\lambda(n)$ is not restricted to the range $(0, 1)$. Thus, Fig. 1 is interpreted from the viewpoint of a linear combiner. Each adaptive filter is estimating the unknown channel impulse response using the same input data. Thus, $\boldsymbol{W}_1(n)$ and $\boldsymbol{W}_2(n)$ are statistically dependent estimates of the unknown channel. There exists a single combining parameter sequence $\lambda(n)$ which minimizes the MSD. The parameter $\lambda(n)$ does not necessarily lie within $(0, 1)$ for all n. Thus, the output $y(n)$ in Fig. 1 is an affine (see footnote 1) combination of the individual outputs $y_1(n)$ and $y_2(n)$. The convex combination is a particular case.

The adaptive scheme is first studied from the viewpoint of an optimal affine combiner. The value of $\lambda(n)$ that minimizes the MSE for each n (conditioned on the filter parameters at iteration n) is determined as a function of the unknown system response. This leads to an optimal affine sequence $\lambda_o(n)$. The statistical properties of an optimal affine combiner are then studied. It is shown that $\lambda_o(n)$ can be outside of the interval $(0, 1)$ for several iterations. Most importantly, $\lambda_o(n)$ is usually negative in steady-state. It is of interest to compare the performance of the adaptive filter using a suboptimal but feasible adjustment algorithm for $\lambda(n)$ with that of the optimal affine combiner. Although the latter is unrealizable, its performance provides an upper bound on the performance of any realizable affine combiner. Suppose a suboptimal (but realizable) algorithm leads to a performance close to that of the optimal affine combiner. Then, there is sufficient motivation for a more detailed study of the algorithm with respect to analysis and implementation issues.

Finally, two realizable schemes for updating $\lambda(n)$ are proposed. The first scheme is based on a stochastic gradient approximation to $\lambda_o(n)$. The second scheme is based on the relative values of averaged estimates of the individual error powers. Both schemes are briefly studied, and their performances are compared to that of the optimal affine combiner. Numerical results support the theoretical findings and show that the analysis closely predicts the probabilistic behavior of the algorithms as observed in Monte Carlo simulations, especially in the neighborhood of the intersection of the MSD's of the individual filters when the hand-off from one filter to the other filter occurs.

II. The Optimal Affine Combiner

A. The Affine Combiner

The system under investigation is shown in Fig. 1. Each filter uses the LMS adaptation rule but with different step sizes μ_i, $i = 1, 2$:

$$\boldsymbol{W}_i(n+1) = \boldsymbol{W}_i(n) + \mu_i e_i(n)\boldsymbol{U}(n), \quad i = 1, 2 \tag{1}$$

where

$$e_i(n) = d(n) - \boldsymbol{W}_i^T(n)\boldsymbol{U}(n), \quad i = 1, 2 \tag{2}$$

$$d(n) = e_o(n) + \boldsymbol{W}_o^T\boldsymbol{U}(n), \tag{3}$$

where $\boldsymbol{W}_i(n)$, $i = 1, 2$ are the N-dimensional adaptive coefficient vectors, $e_o(n)$ is assumed zero-mean, i.i.d. and statistically independent of any other signal in the system, and the input process $u(n)$ is assumed wide-sense stationary. $\boldsymbol{U}(n) = [u(n), \ldots, u(n-N+1)]^T$ is the input vector. It will be assumed, without loss, that $\mu_1 \geq \mu_2$, so that $\boldsymbol{W}_1(n)$ will, in general, converge faster than $\boldsymbol{W}_2(n)$. Also, $\boldsymbol{W}_2(n)$ will converge to the lowest individual steady-state weight misadjustment. The weight vectors $\boldsymbol{W}_1(n)$ and $\boldsymbol{W}_2(n)$ are coupled both deterministically and statistically through $\boldsymbol{U}(n)$ and $e_o(n)$. The stochastic analysis of each individual adaptive filter behavior in (1) is well-known [9]–[11].

The outputs of the two filters are combined as in Fig. 1,

$$y(n) = \lambda(n)y_1(n) + [1 - \lambda(n)]y_2(n), \tag{4}$$

where $y_i(n) = \boldsymbol{W}_i^T\boldsymbol{U}(n)$, $i = 1, 2$, $\lambda(n)$ can be any real number[1] and the overall system error is given by

$$e(n) = d(n) - y(n). \tag{5}$$

The adaptive filter output combination (4) is an affine combination, as $y(n)$ can assume any value on the real line. This setup generalizes the combination of adaptive filter outputs, and can be used to study the properties of the optimal combination.

[1]This case corresponds to an affine (as opposed to convex) combination. The output in (4) can have any real value on the line containing $y_1(n)$ and $y_2(n)$. $y(n)$ is restricted to the points on the line between $y_1(n)$ and $y_2(n)$ in the convex combination case. Two reviewers have noted that the optimal combiner is of the form $y(n) = a(n)y_1(n) + b(n)y_2(n)$ where $a(n) + b(n) \neq 1$ in general. This paper studies the affine combiner.

B. The Optimal Mixing Parameter

Eq. (4) can be written as

$$\begin{aligned} y(n) &= \lambda(n)\boldsymbol{W}_1^T(n)\boldsymbol{U}(n) + [1-\lambda(n)]\boldsymbol{W}_2^T(n)\boldsymbol{U}(n) \\ &= \big\{\lambda(n)\big[\boldsymbol{W}_1(n) - \boldsymbol{W}_2(n)\big] + \boldsymbol{W}_2(n)\big\}^T\boldsymbol{U}(n) \\ &= \big\{\lambda(n)\boldsymbol{W}_{12}(n) + \boldsymbol{W}_2(n)\big\}^T\boldsymbol{U}(n), \end{aligned} \tag{6}$$

where $\boldsymbol{W}_{12}(n) = \boldsymbol{W}_1(n) - \boldsymbol{W}_2(n)$.

Eq. (6) shows that $y(n)$ can be interpreted as a combination of $\boldsymbol{W}_2(n)$ and a weighted version of the difference filter $\boldsymbol{W}_{12}(n)$. It also shows that the combined adaptive filter has an equivalent weight vector given by

$$\boldsymbol{W}_{eq}(n) = \lambda(n)\boldsymbol{W}_{12}(n) + \boldsymbol{W}_2(n). \tag{7}$$

Subtracting (1) for $i = 2$ from (1) for $i = 1$ yields a recursion for $\boldsymbol{W}_{12}(n)$:

$$\boldsymbol{W}_{12}(n+1) = \big[\boldsymbol{I} - \mu_1\boldsymbol{U}(n)\boldsymbol{U}^T(n)\big]\boldsymbol{W}_{12}(n) + (\mu_1 - \mu_2)e_2(n)\boldsymbol{U}(n). \tag{8}$$

Next, let us consider a rule for choosing $\lambda(n)$ that minimizes the conditional mean-square error (MSE) at time n $E[e^2(n)|\boldsymbol{W}_1(n), \boldsymbol{W}_{12}(n)]$. Writing $e(n)$ in (5) as

$$e(n) = e_o(n) + \big[\boldsymbol{W}_{o2}(n) - \lambda(n)\boldsymbol{W}_{12}(n)\big]^T\boldsymbol{U}(n) \tag{9}$$

where $\boldsymbol{W}_{o2}(n) = \boldsymbol{W}_o(n) - \boldsymbol{W}_2(n)$ yields

$$\frac{\partial E[e^2(n)|\boldsymbol{W}_2(n), \boldsymbol{W}_{12}(n)]}{\partial\lambda(n)} = -2E\left[e(n)\boldsymbol{W}_{12}^T(n)\boldsymbol{U}(n)|\boldsymbol{W}_2(n), \boldsymbol{W}_{12}(n)\right] = 0. \tag{10}$$

Using (9), taking the expectation over $\boldsymbol{U}(n)$ and defining the input conditional autocorrelation matrix $\boldsymbol{R}_u = E\left[\boldsymbol{U}(n)\boldsymbol{U}^T(n)|\boldsymbol{W}_2(n), \boldsymbol{W}_{12}(n)\right]$ results in

$$\big[\boldsymbol{W}_{o2}(n) - \lambda(n)\boldsymbol{W}_{12}(n)\big]^T\boldsymbol{R}_u\boldsymbol{W}_{12}(n) = 0. \tag{11}$$

Solving (11) for $\lambda(n) = \lambda_o(n)$ yields

$$\lambda_o(n) = \frac{\boldsymbol{W}_{o2}^T(n)\boldsymbol{R}_u\boldsymbol{W}_{12}(n)}{\boldsymbol{W}_{12}^T(n)\boldsymbol{R}_u\boldsymbol{W}_{12}(n)}, \tag{12}$$

which is the expression for the optimum[2] $\lambda(n)$, as a function of the unknown weight vector $\boldsymbol{W}_o$. The behavior of the LMS algorithm in the adaptive structure of Fig. 1 using $\lambda(n) = \lambda_o(n)$ given by (12) can be used as a benchmark for evaluation of different schemes devised for online adjustment of $\lambda(n)$.

III. Statistical Properties of the Optimal Affine Combiner

This section derives important properties of the optimal adaptive affine combiner. These properties can be used as design bounds or as benchmarks for evaluation of realizable adaptive filtering combining schemes. In the analysis that follows, $u(n)$ will be assumed for simplicity to be white, Gaussian, with zero mean and variance σ_u^2. Thus, $\boldsymbol{R}_u = E\left[\boldsymbol{U}(n)\boldsymbol{U}^T(n)|\boldsymbol{W}_2(n), \boldsymbol{W}_{12}(n)\right] = E\left[\boldsymbol{U}(n)\boldsymbol{U}^T(n)\right] = \sigma_u^2\boldsymbol{I}$ since the input at time n is assumed statistically independent of the weights at time n (independence theory). The step sizes μ_1 and μ_2 will be given by

$$\mu_1 = \frac{1}{(N+\gamma)\sigma_u^2} \tag{13}$$

$$\mu_2 = \delta\mu_1, \tag{14}$$

where $\gamma > 0$ and $0 < \delta < 1$. Parameter γ controls the departure of μ_1 from $1/(N\sigma_u^2)$, which is approximately the LMS step-size for maximum convergence speed [10]. Parameter δ controls the ratio μ_1/μ_2. Eqs. (13) and (14) allow μ_1 and μ_2 to lie in the algorithm's stability range and yield $\mu_1 > \mu_2$. Thus, filter 1 is the faster adapting filter. The white Gaussian input signal assumption is not necessarily representative of the signals used in every application. However, the theoretical approach and results obtained under this assumption are suggestive of fundamental analysis and design issues [9], [12].

A. Properties of $\lambda_o(n)$

This section derives approximate results for the initial and steady-state behaviors of $\lambda_o(n)$. The mathematical models provide important insights into the ideal behavior of the mixing parameter $\lambda(n)$. The structure of Fig. 1 is used in practical situations where μ_1 is large for fast initial convergence and μ_2 is small to reduce the steady-state misadjustment. After a brief initial transient (very difficult to model because of the fast changes occurring in the two statistics of the weight vectors during this period), filter 1 ($\boldsymbol{W}_1(n)$) dominates the adaptive behavior of the combined adaptive filter. Initially $\boldsymbol{W}_1(n)$ adapts more quickly towards the optimum weight vector than $\boldsymbol{W}_2(n)$. Thus, from (4), the optimum $\lambda_o(n)$ is close to unity during this initial phase. On the other hand, $\boldsymbol{W}_2(n)$ dominates the adaptive filter behavior near steady-state. Thus $\lambda_o(n)$ is near zero for large n. The model should also accurately estimate the number of iterations when filter 1 hands over the control of the adaptation to filter 2. Thus $\lambda_o(n)$ moves away from unity. The optimum behavior of $\lambda(n)$ at the start and in steady-state, and an accurate estimate of

[2] $\lambda_o(n)$ is optimum in the sense that it yields the minimum conditional MSE at each time instant.

when $\lambda(n)$ transfers the combined weight from filter 1 to filter 2 are useful pieces of information. This information should help in the design of effective practical algorithms for adjusting of mixing parameter.

1) Steady-State Behavior: Consider an approximation[3] of the steady-state mean of eq. (12) for the white input case

$$\lim_{n\to\infty} E[\lambda_o(n)] \simeq \lim_{n\to\infty} \frac{E\Big[\boldsymbol{W}_{o2}^T(n)\boldsymbol{W}_{12}(n)\Big]}{E\big[\boldsymbol{W}_{12}^T(n)\boldsymbol{W}_{12}(n)\big]}. \tag{15}$$

Since $\lim_{n\to\infty} E[\boldsymbol{W}_1(n)] = \lim_{n\to\infty} E[\boldsymbol{W}_2(n)] = \boldsymbol{W}_o$,

$$\lim_{n\to\infty} E\Big[\boldsymbol{W}_{o2}^T(n)\boldsymbol{W}_{12}(n)\Big] = -E[\boldsymbol{W}_2^T(n)\boldsymbol{W}_1(n)] + E[\boldsymbol{W}_2^T(n)\boldsymbol{W}_2(n)]. \tag{16}$$

Eq. (16) requires the evaluation of $E[\boldsymbol{W}_2^T(n)\boldsymbol{W}_1(n)]$ and $E[\boldsymbol{W}_2^T(n)\boldsymbol{W}_2(n)]$. Using (1) and (2) with the appropriate indexes and (3), neglecting the statistical correlations between the input and the two weight vectors [9], [11] and using the Gaussian moment factoring theorem [14], the following recursion can be written for $E[\boldsymbol{W}_2^T(n)\boldsymbol{W}_1(n)]$:

$$\begin{aligned} E[\boldsymbol{W}_2^T(n+1)\boldsymbol{W}_1(n+1)] = {} & [1-(\mu_1+\mu_2)\sigma_u^2 + (N+2)\mu_1\mu_2\sigma_u^4]E[\boldsymbol{W}_2^T(n)\boldsymbol{W}_1(n)] \\ & + \mu_2\sigma_u^2\big[1-(N+2)\mu_1\sigma_u^2\big]\boldsymbol{W}_o^T E[\boldsymbol{W}_1(n)] \\ & + \mu_1\sigma_u^2\big[1-(N+2)\mu_2\sigma_u^2\big]\boldsymbol{W}_o^T E[\boldsymbol{W}_2(n)] \\ & + \mu_1\mu_2\sigma_u^4\left[N\left(\frac{\sigma_o^2}{\sigma_u^2}\right) + (N+2)\boldsymbol{W}_o^T\boldsymbol{W}_o\right]. \end{aligned} \tag{17}$$

The steady-state solution of (17) is determined in Appendix I for $\boldsymbol{W}_1(0) = \boldsymbol{W}_2(0) = \mathbf{0}$, yielding:

$$\lim_{n\to\infty} E[\boldsymbol{W}_2^T(n)\boldsymbol{W}_1(n)] = \boldsymbol{W}_o^T\boldsymbol{W}_o + \frac{\mu_1\mu_2 N\sigma_o^2}{(\mu_1+\mu_2) - \mu_1\mu_2(N+2)\sigma_u^2} \tag{18}$$

and

$$\lim_{n\to\infty} E[\boldsymbol{W}_2^T(n)\boldsymbol{W}_2(n)] = \boldsymbol{W}_o^T\boldsymbol{W}_o + \frac{\mu_2 N\sigma_o^2}{2-\mu_2(N+2)\sigma_u^2}. \tag{19}$$

Thus, (16) can be written as

[3]There are two main justifications for the approximations such as in proceeding from Eq. (12) to Eq. (15): 1) Evaluating expectations of quotients of correlated random variables is usually a very difficult undertaking. Approximations are often made in order to make progress in the analysis. We have chosen the approximation that the expectation of the ratio is approximately the ratio of the expectations (see for instance [13] and reference [13] within), 2) The latter simulation results support this approximation.

$$\lim_{n\to\infty} E\big[\boldsymbol{W}_{o2}^T(n)\boldsymbol{W}_{12}(n)\big] = -\frac{\mu_1\mu_2 N\sigma_o^2}{(\mu_1+\mu_2)-\mu_1\mu_2(N+2)\sigma_u^2} + \frac{\mu_2 N\sigma_o^2}{2-\mu_2(N+2)\sigma_u^2}.^{4} \tag{20}$$

Using the same results, the denominator of (15) can be written as

$$\begin{aligned}\lim_{n\to\infty} E\big[\boldsymbol{W}_{12}^T(n)\boldsymbol{W}_{12}(n)\big] &= E\big[\boldsymbol{W}_1^T(n)\boldsymbol{W}_1(n)\big] - 2E\big[\boldsymbol{W}_2^T(n)\boldsymbol{W}_1(n)\big] + E\big[\boldsymbol{W}_2^T(n)\boldsymbol{W}_2(n)\big] \\ &= \frac{\mu_1 N\sigma_o^2}{2-\mu_1(N+2)\sigma_u^2} - 2\frac{\mu_1\mu_2 N\sigma_o^2}{(\mu_1+\mu_2)-\mu_1\mu_2(N+2)\sigma_u^2} + \frac{\mu_2 N\sigma_o^2}{2-\mu_2(N+2)\sigma_u^2}.\end{aligned} \tag{21}$$

Using (13) and (14), these expressions reduce to

$$\lim_{n\to\infty} E\Big[\boldsymbol{W}_{o2}^T(n)\boldsymbol{W}_{12}(n)\Big] = \frac{\delta N\sigma_o^2}{(N+2)\sigma_u^2}\left[\frac{1}{2-\delta+\frac{2(\gamma-2)}{N+2}} - \frac{1}{1+\frac{(1+\delta)(\gamma-2)}{N+2}}\right] \tag{22}$$

and

$$\lim_{n\to\infty} E\big[\boldsymbol{W}_{12}^T(n)\boldsymbol{W}_{12}(n)\big] = \frac{N\sigma_o^2}{(N+2)\sigma_u^2}\left[\frac{1}{1+2\frac{\gamma-2}{N+2}} - \frac{2\delta}{1+\frac{(1+\delta)(\gamma-2)}{N+2}} + \frac{\delta}{2-\delta-2\frac{\gamma-2}{N+2}}\right]. \tag{23}$$

For the practical case of $\gamma \ll N+4$, (22) and (23) yield[5]

$$\lim_{n\to\infty} E\Big[\boldsymbol{W}_{o2}^T(n)\boldsymbol{W}_{12}(n)\Big] = \frac{\delta(\delta-1)}{2-\delta}\frac{N\sigma_o^2}{(N+2)\sigma_u^2} \tag{24}$$

and

$$\lim_{n\to\infty} E\big[\boldsymbol{W}_{12}^T(n)\boldsymbol{W}_{12}(n)\big] = \frac{2(\delta-1)^2}{2-\delta}\frac{N\sigma_o^2}{(N+2)\sigma_u^2}. \tag{25}$$

Finally, using (24) and (25) in (15) yields the steady-state value of $E[\lambda_o(n)]$:

$$\lim_{n\to\infty} E[\lambda_o(n)] \simeq \frac{\delta}{2(\delta-1)}. \tag{26}$$

Expression (26) shows two interesting properties of $E[\lambda_o(n)]$ for typical (large) values of N. Its steady-state value is negative (since $\delta < 1$) and depends only on the ratio μ_2/μ_1.

2) Initial Behavior: This section studies the behavior of $E[\lambda_o(n)]$ after $\boldsymbol{W}_1(n)$ has converged but $\boldsymbol{W}_2(n)$ is still in transient mode. This operating region includes the period during which the steady-state behavior of $\boldsymbol{W}_1(n)$

[4]These simulation results imply that 1) disabling filter 1 by setting $\lambda(n) = 0$, and 2) transferring convergence control to filter 2 is not sufficient to minimize the steady-state MSE. This is because of the effect of the cross-correlation between $\boldsymbol{W}_1(n)$ and $\boldsymbol{W}_2(n)$ for some values of μ_1 and μ_2 (see eqs. (16) and (20)). Thus, it is necessary to increase the contribution of $y_1(n)$ in Eq. (4) to greater than unity and subtract a small part of $y_1(n)$. This is done by making $\lambda(n) < 0$ in steady-state.

[5]Note that $\mu_1 = 1/(N\sigma_u^2) \simeq \mu_{\max}/2$ for $\gamma = 0$, for $\mu_{\max}$ equal to the LMS stability limit.

is still a better estimate of $\boldsymbol{W}_o$ than $\boldsymbol{W}_2(n)$ and the period during which the convergence control is being handed from $\boldsymbol{W}_1(n)$ over to $\boldsymbol{W}_2(n)$.

In steady-state, $\boldsymbol{W}_1(n)$ can be written as

$$\boldsymbol{W}_1(n) = \boldsymbol{W}_o + \boldsymbol{\Psi}_1(n) \tag{27}$$

where $\boldsymbol{\Psi}_1(n)$ is a zero-mean stationary white vector [9]. It is also assumed that the fluctuations in $\boldsymbol{W}_1(n)$ at convergence are statistically independent of $\boldsymbol{W}_2(n)$ in this phase (i.e. the independence assumption is used in the transition phase to provide mathematical tractability).

Using (12), the above assumptions and the approximation in (15) for all n,

$$E[\lambda_o(n)] \simeq \frac{E\Big[\boldsymbol{W}_{o2}^T(n)\boldsymbol{W}_{12}(n)\Big]}{E\Big[\boldsymbol{W}_{12}^T(n)\boldsymbol{W}_{12}(n)\Big]}, \tag{28}$$

yields

$$E[\lambda_o(n)] \simeq \frac{\boldsymbol{W}_o^T\Big\{\boldsymbol{W}_o - 2E[\boldsymbol{W}_2(n)]\Big\} + E[\boldsymbol{W}_2^T(n)\boldsymbol{W}_2(n)]}{D(n)} \tag{29}$$

where

$$\begin{aligned} D(n) = \boldsymbol{W}_o^T\Big\{\boldsymbol{W}_o - 2E[\boldsymbol{W}_2(n)]\Big\} + E[\boldsymbol{W}_2^T(n)\boldsymbol{W}_2(n)] \\ + E[\boldsymbol{\Psi}_1^T(n)\boldsymbol{\Psi}_1(n)]. \end{aligned}$$

From [9],

$$E[\boldsymbol{\Psi}_1(n)\boldsymbol{\Psi}_1^T(n)] = \frac{\mu_1\sigma_o^2}{2-\mu_1(N+2)\sigma_u^2}\boldsymbol{I}, \tag{30}$$

and thus

$$E\big[\boldsymbol{\Psi}_1^T(n)\boldsymbol{\Psi}_1(n)\big] = \frac{\mu_1 N\sigma_o^2}{2-\mu_1(N+2)\sigma_u^2}. \tag{31}$$

Eq. (29) also requires the evaluation of $E[\boldsymbol{W}_2(n)]$ and $E[\boldsymbol{W}_2^T(n)\boldsymbol{W}_2(n)]$. For the typical adaptive filter initialization $\boldsymbol{W}_1(0) = \boldsymbol{W}_2(0) = \boldsymbol{0}$, these expressions can easily be obtained from Appendix I (Eq. (53) for $i = 2$ and (57) with μ_1 replaced with μ_2). After some algebraic manipulations, this procedure yields

$$E[\lambda_o(n)] \simeq \frac{1}{1+A(n)}, \tag{32}$$

where

$$A(n)=\frac{\dfrac{\mu_1 N\sigma_o^2}{2-\mu_1\sigma_u^2(N+2)}}{\left[\boldsymbol{W}_o^T\boldsymbol{W}_o-\dfrac{\mu_2 N\sigma_o^2}{2-\mu_2\sigma_u^2(N+2)}\right]\rho^n+\dfrac{\mu_2 N\sigma_o^2}{2-\mu_2\sigma_u^2(N+2)}} \tag{33}$$

with $\rho=1-2\mu_2\sigma_u^2+\mu_2^2\sigma_u^4(N+2)$.

B. Mean-Square Deviation

The optimum MSD[6] of the combined filter at time n is

$$\begin{aligned}
&\text{MSD}_c(n)\\
&=E\Big\{\big[\boldsymbol{W}_o-\boldsymbol{W}_{eq}(n)\big]^T\big[\boldsymbol{W}_o-\boldsymbol{W}_{eq}(n)\big]\Big\}\Big|_{\lambda(n)=\lambda_o(n)}\\
&=E\Big\{\big[\boldsymbol{W}_{o2}(n)-\lambda_o(n)\boldsymbol{W}_{12}(n)\big]^T\big[\boldsymbol{W}_{o2}(n)-\lambda_o(n)\boldsymbol{W}_{12}(n)\big]\Big\}\\
&=E\Big[\boldsymbol{W}_{o2}^T(n)\boldsymbol{W}_{o2}(n)\Big]-2E\Big[\lambda_o(n)\boldsymbol{W}_{o2}^T(n)\boldsymbol{W}_{12}(n)\Big]\\
&\quad+E\Big[\lambda_o^2(n)\boldsymbol{W}_{12}^T(n)\boldsymbol{W}_{12}(n)\Big].
\end{aligned} \tag{34}$$

Inserting (12) in (34) yields

$$\text{MSD}_c(n)=E\Big[\boldsymbol{W}_{o2}^T(n)\boldsymbol{W}_{o2}(n)\Big]-E\left\{\frac{\Big[\boldsymbol{W}_{o2}^T(n)\boldsymbol{W}_{12}(n)\Big]^2}{\boldsymbol{W}_{12}^T(n)\boldsymbol{W}_{12}(n)}\right\}. \tag{35}$$

The first term of (35) is $\text{MSD}_2(n)$, the MSD of the second adaptive filter. Since the MSD is a positive quantity, (35) indicates that $\text{MSD}_c(n)$ is always less than $\text{MSD}_2(n)$. Eq. (7) can also be written in terms of $\boldsymbol{W}_1(n)$ and $\boldsymbol{W}_{12}(n)$ as

$$\boldsymbol{W}_{eq}(n)=\boldsymbol{W}_1(n)-[1-\lambda(n)]\boldsymbol{W}_{12}(n). \tag{36}$$

Also, it can be easily verified that

$$1-\lambda_o(n)=-\frac{\boldsymbol{W}_{o1}^T(n)\boldsymbol{W}_{12}(n)}{\boldsymbol{W}_{12}^T(n)\boldsymbol{W}_{12}(n)}. \tag{37}$$

where $\boldsymbol{W}_{o1}(n)=\boldsymbol{W}_o-\boldsymbol{W}_1(n)$. Then, substituting (36) for $\boldsymbol{W}_{eq}(n)$ in the first line of (34) and using (37), it can be easily shown that

[6]The optimum MSD is defined here as the MSD obtained for $\lambda(n)=\lambda_o(n)$.

$$\text{MSD}_c(n) = E\Big[\boldsymbol{W}_{o1}^T(n)\boldsymbol{W}_{o1}(n)\Big] - E\left\{\frac{\Big[\boldsymbol{W}_{o1}^T(n)\boldsymbol{W}_{12}(n)\Big]^2}{\boldsymbol{W}_{12}^T(n)\boldsymbol{W}_{12}(n)}\right\}. \tag{38}$$

Eqs. (35) and (38) show the important result that the optimum linear combiner leads to a $\text{MSD}_c(n)$ that is smaller than both $\text{MSD}_1(n)$ and $\text{MSD}_2(n)$ for every n. Thus, the optimum linear combiner defined by $\lambda_o(n)$ in (12) always performs better than any of the two individual adaptive filters. An approximate evaluation of (35) will be performed in Section III-C. A similar approximation could be derived using (38).

C. Behavior of the MSD$_c$ in the Transition Region

This section studies the behavior of the $\text{MSD}_c(n)$ after $\boldsymbol{W}_1(n)$ has converged but $\boldsymbol{W}_2(n)$ has not yet converged. This operating region corresponds to the range of iterations during which the convergence control is being handed from $\boldsymbol{W}_1(n)$ to $\boldsymbol{W}_2(n)$. This region will be denoted here as the transition region. From (35) and using the same approximation used in (28),

$$\begin{aligned}\text{MSD}_c(n) &= \text{MSD}_2(n) - E\left\{\frac{\boldsymbol{W}_{o2}^T(n)\boldsymbol{W}_{12}(n)\boldsymbol{W}_{12}^T(n)\boldsymbol{W}_{o2}(n)}{\boldsymbol{W}_{12}^T(n)\boldsymbol{W}_{12}(n)}\right\} \\ &\simeq \text{MSD}_2(n) - C(n),\end{aligned} \tag{39}$$

where

$$C(n) = \frac{E\Big[\boldsymbol{W}_{o2}^T(n)\boldsymbol{W}_{12}(n)\boldsymbol{W}_{12}^T(n)\boldsymbol{W}_{o2}(n)\Big]}{E\Big[\boldsymbol{W}_{12}^T(n)\boldsymbol{W}_{12}(n)\Big]}. \tag{40}$$

Now, assuming $\boldsymbol{W}_1(n)$ in steady-state as in Section III-A.2 and using (27) in $\boldsymbol{W}_{12}(n) = \boldsymbol{W}_1(n) - \boldsymbol{W}_2(n)$, the second term of (39) can be written as

$$\begin{aligned}C(n) &= \frac{E\Big\{\boldsymbol{W}_{o2}^T(n)\big[\boldsymbol{W}_{o2}(n) + \boldsymbol{\Psi}_1(n)\big]\big[\boldsymbol{W}_{o2}(n) + \boldsymbol{\Psi}_1(n)\big]^T\boldsymbol{W}_{o2}(n)\Big\}}{E\Big\{\big[\boldsymbol{W}_{o2}(n) + \boldsymbol{\Psi}_1(n)\big]^T\big[\boldsymbol{W}_{o2}(n) + \boldsymbol{\Psi}_1(n)\big]\Big\}} \\ &\simeq \frac{E\Big\{\boldsymbol{W}_{o2}^T(n)E\big[\boldsymbol{\Psi}_1(n)\boldsymbol{\Psi}_1^T(n)\big]\boldsymbol{W}_{o2}(n)\Big\}}{B(n)} + \frac{E\Big\{\big[\boldsymbol{W}_{o2}^T(n)\boldsymbol{W}_{o2}(n)\big]^2\Big\}}{B(n)}\end{aligned} \tag{41}$$

with $B(n) = E\Big[\boldsymbol{W}_{o2}^T(n)\boldsymbol{W}_{o2}(n)\Big] + E\big[\boldsymbol{\Psi}_1^T(n)\boldsymbol{\Psi}_1(n)\big]$. Eq. (41) assumes that the fluctuations in $\boldsymbol{W}_1(n)$ at convergence are independent of $\boldsymbol{W}_2(n)$ (still in the transient mode). Now, using (31) in (41) yields

$$C(n) \simeq \frac{\dfrac{\mu_1\sigma_o^2}{2-\mu_1(N+2)\sigma_u^2}\text{MSD}_2(n) + E\Big\{\big[\boldsymbol{W}_{o2}^T(n)\boldsymbol{W}_{o2}(n)\big]^2\Big\}}{\text{MSD}_2(n) + \dfrac{\mu_1 N\sigma_o^2}{2-\mu_1(N+2)\sigma_u^2}}. \tag{42}$$

Assume, in the transition region, that the fluctuations in $\boldsymbol{W}_{o2}^T(n)\boldsymbol{W}_{o2}(n)$ are still small in comparison to its mean. Then,

$$E\left\{\left[\boldsymbol{W}_{o2}^T(n)\boldsymbol{W}_{o2}(n)\right]^2\right\} \simeq E^2\left[\boldsymbol{W}_{o2}^T(n)\boldsymbol{W}_{o2}(n)\right]. \tag{43}$$

Inserting (43) into (42) and then into (39) yields

$$\text{MSD}_c(n) = \frac{\text{MSD}_2(n)\dfrac{\mu_1\sigma_o^2(N-1)}{2-\mu_1(N+2)}}{\text{MSD}_2(n) + \dfrac{\mu_1\sigma_o^2 N}{2-\mu_1(N+2)}}, \tag{44}$$

where (44) assumes $\sigma_u^2 = 1$. Eq. (44) agrees with physical intuition. The right hand side of (44) is always less than either $\text{MSD}_1(n)$ or $\text{MSD}_2(n)$. When $\text{MSD}_2(n)$ becomes small in comparison to the steady-state of $\text{MSD}_1(n)$, then $\text{MSD}_c(n) \simeq \text{MSD}_2(n)$.

D. Simulation Results

This section presents some simulation results to verify the accuracy of the theoretical model developed for the behavior of the optimum adaptive filter affine combination. The unknown system response ($N = 32$) is shown in Fig. 2.

Figure 3 shows two representative simulation examples to verify the accuracy of the model for $E[\lambda_o(n)]$ given by (26) and (32). The horizontal lines show the steady-state mean behavior of $\lambda_o(n)$ as predicted by (26). Figures 3(a) and 3(b) display some properties of the model. There is excellent agreement between the steady-state value predicted by (26) and the simulation results. The negative steady-state value implies that a small portion of the estimate of $\boldsymbol{W}_o$ obtained using filter 1 should be subtracted for the estimate of $\boldsymbol{W}_o$ using filter 2. This occurs because the channel estimates from the two adaptive filters are correlated. An excellent match is observed for the dynamical model (32) at the beginning of operation, disregarding the initial transient behavior. The model accuracy decreases as $\boldsymbol{W}_1(n)$ becomes more correlated with $\boldsymbol{W}_2(n)$ as the algorithm approaches steady-state. Nevertheless, the theoretical model provides a good-to-moderately good prediction of $E[\lambda_o(n)]$ during the transfer. The MSD plots in figures 4(a) and 4(b) show that $\text{MSD}_c(n)$ is always less than either $\text{MSD}_1(n)$ or $\text{MSD}_2(n)$. This behavior is expected from an optimal combiner and verified in (35) and (38). These curves represent the best performance that could be obtained using two LMS adaptive filters as in Fig. 1.

Fig. 5 shows the evolution of $\text{MSD}_c(n)$ for two representative examples. The theoretical curves were obtained from the expression for $\text{MSD}_1(n)$ [9] for n before the convergence of $\boldsymbol{W}_1(n)$ and from (44) afterwards[7]. Apart

[7]The models were reversed when $\text{MSD}_1(n) \leq 1.01\mu_1\sigma_u^2 N/[2-(N+2)\mu_1]$. However, the value of this reversal threshold is not critical.

from a slightly slower convergence of the theoretical model for small n (error of the classical LMS model), excellent agreement can be observed. This figure confirms the accuracy of (44) to model the behavior of the combined filter in the transition region.

IV. Iterative Algorithms to Adjust $\lambda(n)$

The previous derivation of the optimal linear combiner was based upon prior knowledge of the unknown system response $\boldsymbol{W}_o$. Clearly, this is not the case in reality. However, the theoretical model and its derived properties can be used to upper bound the performance of practical algorithms for adjusting $\lambda(n)$ without such knowledge. Algorithms that yield close-to-optimal performance for typical unknown responses can be considered as good candidates for practical applications. Performance close to the optimal suggests that further analytical study of a new algorithm would be worth the effort. This is especially important for the adaptive combiner structure. A detailed performance analysis of specific adaptation schemes is a nontrivial task as demonstrated in [6], for example. This section studies two algorithms for the adjustment of $\lambda(n)$. The first algorithm is based upon a stochastic gradient search for the optimal $\lambda(n)$. The second is based on the ratio of the average error powers from each individual adaptive filter. The performances of these algorithms are then compared to the optimal performance. The performance of other algorithms applicable to the system in Fig. 1, such as the algorithm studied in [6]–[8], can also be compared with the optimum performance.

A. Stochastic Gradient Search for $\lambda(n)$

Consider a stochastic gradient search to estimate the optimum instantaneous value of $\lambda(n)$. From (10), the stochastic gradient of the conditional MSE is proportional to $e(n)\boldsymbol{W}_{12}^T(n)\boldsymbol{U}(n)$. Using (9) for $e(n)$, the stochastic gradient algorithm to estimate $\lambda_o(n)$ is

$$\lambda_1(n+1) = \lambda_1(n) + \mu_\lambda[d(n) - \widetilde{\boldsymbol{W}}_{12}^T(n)\boldsymbol{U}(n)]\boldsymbol{W}_{12}^T(n)\boldsymbol{U}(n). \tag{45}$$

with

$$\widetilde{\boldsymbol{W}}_{12}(n) = \lambda_1(n)\boldsymbol{W}_1(n) + [1-\lambda_1(n)]\boldsymbol{W}_2(n). \tag{46}$$

Eq. (45) is a linear first order stochastic time-varying recursion in the scalar parameter $\lambda_1(n)$. The stochastic behavior of this recursion has been analyzed elsewhere but is not presented here for space reasons. The accuracy of the theoretical analysis and the performance of the proposed algorithm for adjusting $\lambda_1(n)$ are evaluated here. Appropriate values of μ_λ were chosen so that the algorithm was able to track the adaptation of $\boldsymbol{W}_1(n)$ and $\boldsymbol{W}_2(n)$.

Some difficulties were encountered regarding the tradeoff between stability of the recursion (45) and the algorithm's tracking capabilities in the initial phase of adaptation. Sufficiently small values of μ_λ, ($\mu_\lambda < 1$), were found so that (45) was stable. However, these values were not large enough to track the adaptation of $\boldsymbol{W}_1(n)$ and $\boldsymbol{W}_2(n)$. Larger values of μ_λ ($\mu_\lambda > 1$) improved the tracking but led to instability during the early phase of the adaptation in (45). Considering the optimum desired behavior for $E[\lambda_o(n)]$, a satisfactory solution was obtained by constraining $\lambda_1(n)$ in (45) to be less than or equal to 1 for all n. Larger (smaller) signal-to-noise ratios (SNR $= \sigma_u^2/\sigma_o^2$) require larger (smaller) μ_λ, which in turn requires the application of the constraint to $\lambda_1(n)$ for longer (shorter) periods.

Fig. 6 displays the behaviors of Monte Carlo simulations (50 runs) and the theoretical model for $E[\lambda_1(n)]$ for two distinct sets of parameters. The large initial fluctuations in the simulations indicate the application of the stability limitation imposed on $\lambda_1(n)$ in (45). The theoretical model has also been limited so that $E[\lambda_1(n)] \leq 1$. Note the close agreement between the theory and simulations, especially at convergence. Note also the wide range of μ_λ needed for effective performance for the two cases. For $\sigma_o^2 = 10^{-6}$, for example, values of μ_λ around 5000 are required. This is caused by the nonlinear behavior of (45). The parameter $\lambda(n)$ must change very rapidly for large signal-to-noise ratios (such as in Fig. 6(b)) in order to track the rapid changes in $\boldsymbol{W}_1(n)$ and $\boldsymbol{W}_2(n)$.

Fig. 7 compares the behavior of $E[\lambda_1(n)]$ with the optimum behavior of $E[\lambda_o(n)]$. It can be easily verified that the performance of the stochastic gradient algorithm for $\lambda_1(n)$ is very close to that of the optimum combiner. Fig. 8 displays the simulation results for MSD$_c(n)$ obtained from Monte Carlo simulations (50 MC's) using (1)–(4) and (45), and the theoretical predictions. A good-to-excellent agreement between the theory and the simulations can be verified, especially convergence time and the steady-state behavior. Again, the fluctuations in the initial transient phase indicate the action of the stability control. Fig. 9 compares the behaviors of MSD$_c(n)$ using $\lambda_1(n)$ and the optimum $\lambda_o(n)$. Again, it is clear that the updating of $\lambda(n)$ using (45) leads to a performance that is very close to that of the optimum combiner, especially in convergence speed and steady-state MSD.

The stochastic gradient algorithm requires a good estimate of the noise power to reasonably select μ_λ and mildly constrain $\lambda_1(n)$ in recursion (45). The accuracy of this estimate could limit the usefulness of the stochastic gradient algorithm for some applications. The next section considers a different scheme for choosing $\lambda(n)$, based on the average error powers of the two filters. This scheme is insensitive to σ_o^2 and performs nearly as well as the stochastic gradient approach.

B. Error Power Based Scheme for Updating $\lambda(n)$

A function of time averaged error powers could be a good candidate for an estimator of the optimum $\lambda(n)$ for each n. The individual adaptive error powers are good indicators of the contribution of each adaptive output to the quality of the present estimation of $d(n)$. These errors are readily available and do not need an estimate of the additive noise power.

Consider a uniform sliding time average of the instantaneous errors

$$\hat{e}_1^2(n) = \frac{1}{K} \sum_{m=n-K+1}^{n} e_1^2(m) \tag{47}$$

$$\hat{e}_2^2(n) = \frac{1}{K} \sum_{m=n-K+1}^{n} e_2^2(m) \tag{48}$$

where K is the averaging window. Then, consider the instantaneous value of $\lambda(n)$ determined as

$$\lambda_2(n) = 1 - \kappa \text{ erf}\left(\frac{\hat{e}_1^2(n)}{\hat{e}_2^2(n)}\right) \tag{49}$$

where

$$\text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2/2}\, dt. \tag{50}$$

Eq. (49) allows $\lambda_2(n)$ to vary smoothly over $(1-\kappa, 1)$. The value of κ can be selected so that

$$\lim_{n\to\infty} E[\lambda_2(n)] \simeq \lim_{n\to\infty} E[\lambda_o(n)]. \tag{51}$$

Using (26) in (51) and assuming $\lim_{n\to\infty} \frac{e_1^2(n)}{e_2^2(n)} = \infty$ yields

$$\kappa = 1 - \frac{\delta}{2(\delta - 1)}. \tag{52}$$

Note that (49) and (52) do not require any *a priori* information about the noise power nor an additional memory parameter as in the stochastic gradient scheme. Fig. 10 shows two typical examples of the behavior of $\lambda_2(n)$, as compared to $E[\lambda_o(n)]$. It is clear from these figures that the use of (49) and (52) leads to a behavior for the weighting factor that is reasonably close to the optimum. Fig. 11 shows the behaviors of MSD$_c(n)$ using $\lambda(n) = \lambda_2(n)$ (red curves) and $\lambda(n) = \lambda_o(n)$ (black curves). These results clearly show that the proposed algorithm leads to a very good practical implementation of the linear combiner. Note that a stochastic analysis of the transient behavior is quite complicated for this algorithm because of the erf nonlinearity. Nevertheless, the theoretical analysis of the optimum case provided useful insights for the design and evaluation of the algorithm.

V. CONCLUSIONS

This paper has studied the statistical behavior of an affine combination of the outputs of two LMS adaptive filters that simultaneously adapt using the same inputs. The purpose of the affine combination is to obtain an LMS adaptive filter with fast convergence and small steady-state MSD. The affine combination studied is a generalization of the convex combination where the combination factor $\lambda(n)$ is restricted to the interval $(0, 1)$. Here the viewpoint was taken that the two filters each produce dependent estimates of the unknown channel. Thus, there exists a sequence

of optimal affine combining coefficients which minimizes the MSE. First, the optimal unrealizable affine combiner was studied and provided the best possible performance. Then, two new schemes were proposed for practical applications. The first scheme used an unconstrained linear stochastic scalar gradient algorithm for estimating the optimal affine combiner coefficient. This first new scheme performed nearly as well as the optimal unrealizable combiner, providing the same convergence time and steady-state behavior. A second new scheme was investigated that depended upon the time-averaged instantaneous squared error of each adaptive filter. This new scheme was designed using the design information from the optimal affine combiner. With proper design, its performance was also very close to that of the optimum affine combiner for many cases of interest. The theoretical approximations used in the analytical models were validated by Monte Carlo simulations which were in close agreement with the predictions of the analytical models.

APPENDIX I
SOLUTION TO EQ. (17)

Let $q(n) = E[\boldsymbol{W}_2^T(n)\boldsymbol{W}_1(n)]$. Then, using the closed form solutions of (1) for $\boldsymbol{W}_i(0) = \mathbf{0}$

$$E[\boldsymbol{W}_i(n)] = \big[1 - (1 - \mu_i\sigma_u^2)^n\big]\boldsymbol{W}_o, \qquad i = 1, 2, \tag{53}$$

it can be easily verified that (17) can be written in the form

$$q(n+1) = aq(n) + c_1 + c_2\alpha^n + c_3\beta^n, \qquad n \geq 0 \tag{54}$$

where

$$\begin{aligned}
a &= 1 - (\mu_1 + \mu_2)\sigma_u^2 + \mu_1\mu_2(N+2)\sigma_u^4 \\
\alpha &= 1 - \mu_2\sigma_u^2 \\
\beta &= -\mu_1\sigma_u^2 \\
c_1 &= \mu_1\sigma_u^2\big[1 - (N+2)\mu_2\sigma_u^2\big]\boldsymbol{W}_o^T\boldsymbol{W}_o \\
&\quad + \mu_2\sigma_u^2\big[1 - (N+2)\mu_1\sigma_u^2\big]\boldsymbol{W}_o^T\boldsymbol{W}_o \\
&\quad + \mu_1\mu_2\sigma_u^4\left[N\left(\frac{\sigma_o^2}{\sigma_u^2}\right) + (N+2)\boldsymbol{W}_o^T\boldsymbol{W}_o\right] \\
c_2 &= -\mu_1\sigma_u^2\big[1 - (N+2)\mu_2\sigma_u^2\big]\boldsymbol{W}_o^T\boldsymbol{W}_o \\
c_3 &= -\mu_2\sigma_u^2\big[1 - (N+2)\mu_1\sigma_u^2\big]\boldsymbol{W}_o^T\boldsymbol{W}_o.
\end{aligned}$$

Eq. (54) is a first order linear constant-coefficient difference equation with initial condition $q(0) = 0$. Taking the

z-transform of both sides and solving for $Q(z) = \mathcal{Z}[q(n)]$ yields

$$\begin{aligned} Q(z) = &\frac{c_1 z^{-1}}{(1-az^{-1})(1-z^{-1})} + \frac{c_2 z^{-1}}{(1-az^{-1})(1-\alpha z^{-1})} \\ &+ \frac{c_3 z^{-1}}{(1-az^{-1})(1-\beta z^{-1})}. \end{aligned} \tag{55}$$

Making a partial fraction expansion of each term on the right and combining like terms yields

$$\begin{aligned} Q(z) = &\left(\frac{c_1}{1-1/a} + \frac{ac_2}{a-\alpha} + \frac{ac_3}{a-\beta}\right)\frac{z^{-1}}{1-az^{-1}} \\ &+ \left(\frac{c_1}{1-a}\right)\frac{z^{-1}}{1-z^{-1}} + \left(\frac{c_2}{1-a\alpha^{-1}}\right)\frac{z^{-1}}{1-\alpha z^{-1}} \\ &+ \left(\frac{c_3}{1-a\beta^{-1}}\right)\frac{z^{-1}}{1-\beta z^{-1}}. \end{aligned} \tag{56}$$

Finally, inverse transforming (56), yields

$$\begin{aligned} q(n) = &\left(\frac{c_1}{1-1/a} + \frac{ac_2}{a-\alpha} + \frac{ac_3}{a-\beta}\right)a^{n-1} + \left(\frac{c_1}{1-a}\right) \\ &+ \left(\frac{c_2}{1-a\alpha^{-1}}\right)\alpha^{n-1} + \left(\frac{c_3}{1-a\beta^{-1}}\right)\beta^{n-1}, \;\; n > 0 \end{aligned} \tag{57}$$

which is the solution to (17).

For $|a| < 1$, $|\alpha| < 1$ and $|\beta| < 1$, the steady-state value of (57) reduces to $c_1/(1-a)$, or

$$\begin{aligned} &\lim_{n\to\infty} E[\boldsymbol{W}_2^T(n)\boldsymbol{W}_1(n)] \\ &\quad = \boldsymbol{W}_o^T\boldsymbol{W}_o + \frac{\mu_1\mu_2 N\sigma_o^2}{(\mu_1+\mu_2) - \mu_1\mu_2(N+2)\sigma_u^2}. \end{aligned} \tag{58}$$

Using the same calculation, only for $\boldsymbol{W}_1(n) = \boldsymbol{W}_2(n)$ and $\mu_1 = \mu_2$, leads to the steady-state expression for $E[\boldsymbol{W}_2^T(n)\boldsymbol{W}_2(n)]$:

$$\lim_{n\to\infty} E[\boldsymbol{W}_2^T(n)\boldsymbol{W}_2(n)] = \boldsymbol{W}_o^T\boldsymbol{W}_o + \frac{\mu_2 N\sigma_o^2}{2-\mu_2(N+2)\sigma_u^2}. \tag{59}$$

REFERENCES

[1] R. W. Harris, D. M. Chabries, and F. A. Bishop, "Variable step (VS) adaptive filter algorithm," *IEEE Transactions on Acoustic, Speech, and Signal Processing*, vol. 34, no. 2, pp. 309–316, April 1986.

[2] R. H. Kwong and E. W. Johnston, "A variable step size LMS algorithm," *IEEE Transactions on Signal Processing*, vol. 40, no. 7, pp. 1633–1642, July 1992.

[3] T. Aboulnasr and K. Mayyas, "A robust variable step-size LMS-type algorithm: Analysis and simulations," *IEEE Transactions on Signal Processing*, vol. 45, no. 3, pp. 631–639, March 1997.

[4] K. R. Müller, A. Ziehe, N. Murata, and S. I. Amari, *On-Line Learning in Neural Networks*. Cambridge, U.K.: Cambridge University Press, 1998, ch. On-line learning in switching and drifting environments with application to blind source separation, pp. 93–110.

[5] H. C. Shin and A. H. Sayed, "Variable step-size nlms and affine projection algorithms," *IEEE Signal Processing Letters*, vol. 11, no. 11, pp. 132–135, November 2004.

[6] M. Martínez-Ramón, J. Arenas-García, A. Navia-Vásquez, and A. R. Figueiras-Vidal, "An adaptive combination of adaptive filters for plant identification," in *Proc. 14th Int. Conf. Digital Signal Processing*, Santorini, Greece, 2002, pp. 1195–1198.

[7] J. Arenas-García, V. Gómes-Verdejo, M. Martínez-Ramón, and A. R. Figueiras-Vidal, "Separate-variable adaptive combination of LMS adaptive filters for plant identification," in *Proc. 13th IEEE Int. Workshop on Neural Networks and Signal Processing*. Toulouse, France: IEEE, 2003, pp. 239–248.

[8] J. Arenas-Garcia, A. R. Figueiras-Vidal, and A. H. Sayed, "Mean-square performance of a convex combination of two adaptive filters," *IEEE Transactions on Signal Processing*, vol. 54, no. 3, pp. 1078–1090, March 2006.

[9] S. Haykin, *Adaptive Filter Theory*, 4th ed. New Jersey: Prentice-Hall, 2002.

[10] N. J. Bershad and A. Bist, "Fast coupled adaptation for sparse impulse responses using a partial haar transform," *IEEE Transactions on Signal Processing*, vol. 53, no. 3, pp. 966–976, March 2005.

[11] A. H. Sayed, *Fundamentals of Adaptive Filtering*. New York: Wiley-Interscience, 2003.

[12] D. G. Manolakis, V. K. Inglr, and S. M. Kogon, *Statistical and Adaptive Signal Processing*. Boston: McGraw-Hill, 1999.

[13] S. J. M. Almeida, J. C. M. Bermudez, and N. J. Bershad, "A statistical analysis of the affine projection algorithm for unity step size and autoregressive inputs," *IEEE Transactions on Circuits and Systems I - Fundamental Theory and Applications*, vol. 52, no. 7, pp. 1394–1405, July 2005.

[14] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, 3rd ed. New York: McGraw-Hill, 1991.

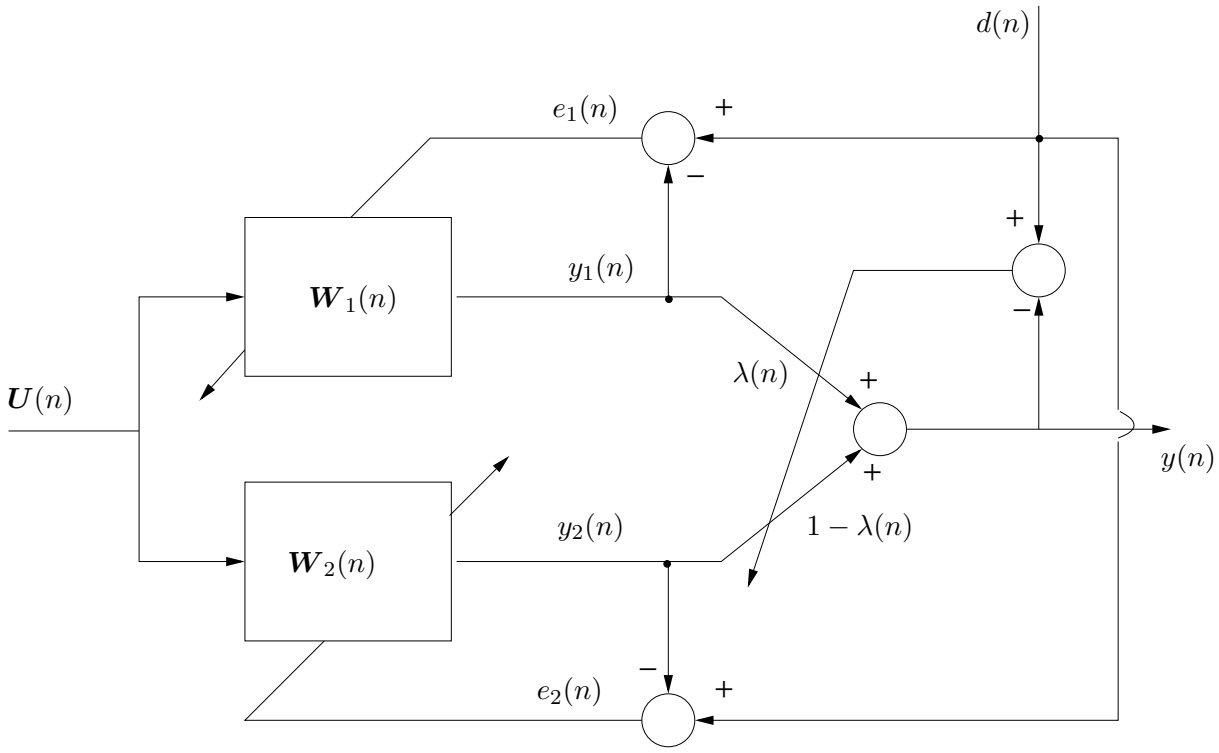


Fig. 1. Adaptive combining of two transversal adaptive filters.

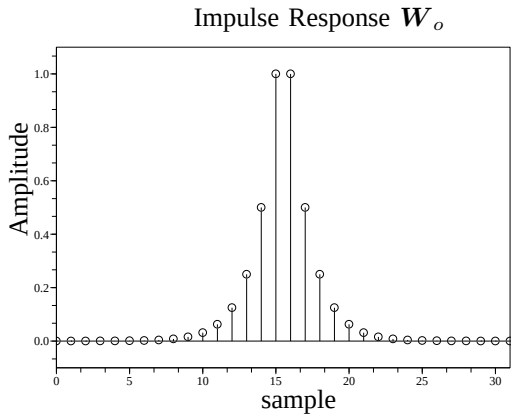
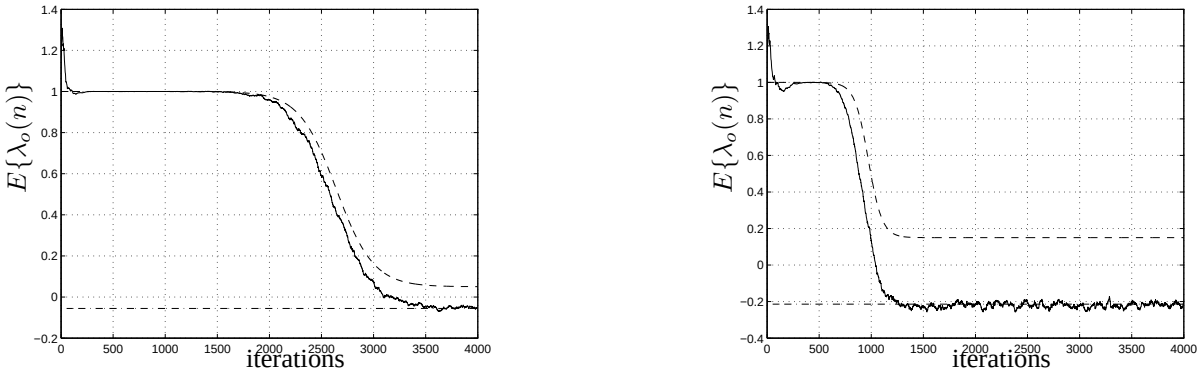
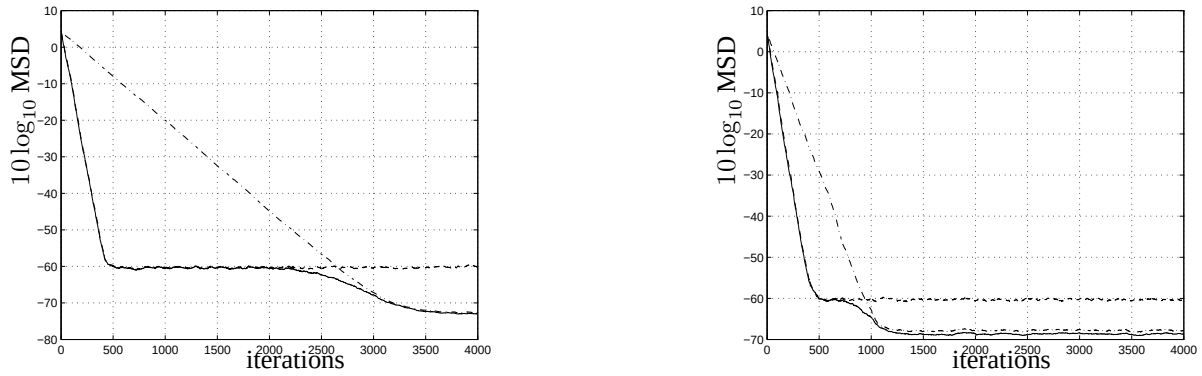


Fig. 2. Impulse response $\boldsymbol{W}_o$ of the unknown system.



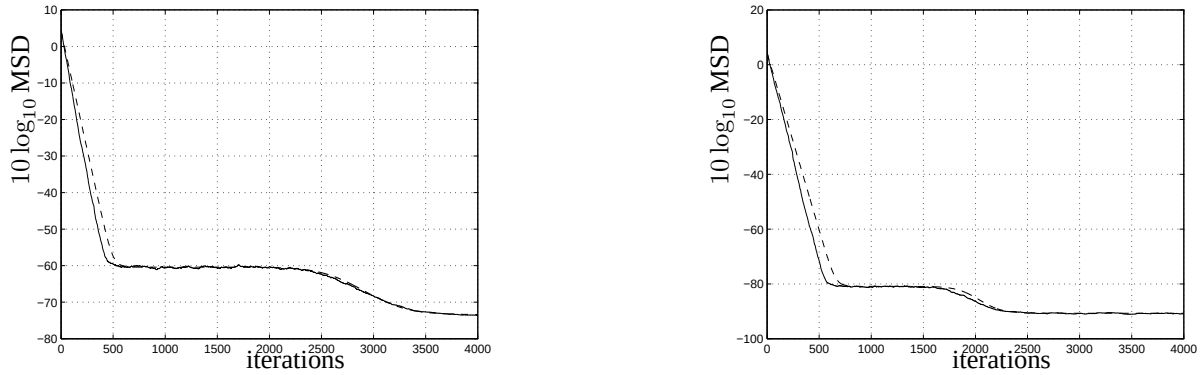
(a) $\delta = 0.1$. (b) $\delta = 0.3$.

Fig. 3. $E\{\lambda_o(n)\}$ for $\sigma_o^2(n) = 10^{-6}$ and $\gamma = 2$. Simulations averaging 50 Monte Carlo runs.



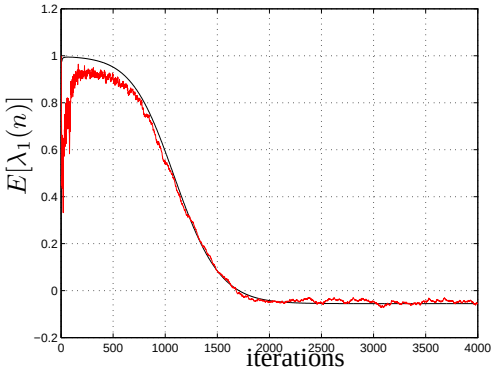
(a) $\delta = 0.1$. (b) $\delta = 0.3$.

Fig. 4. Mean-square deviations for $\sigma_o^2(n) = 10^{-6}$ and $\gamma = 2$. Simulations averaging 50 Monte Carlo runs. $(--)$ $\text{MSD}_1(n)$; $(-\cdot)$ $\text{MSD}_2(n)$; $(-)$ $\text{MSD}_c(n)$..

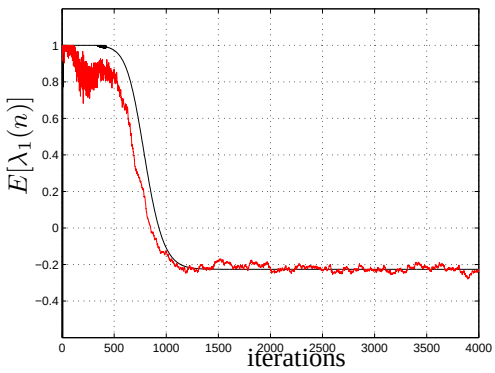


(a) $10\log\{\text{MSD}_c(n)\}$ for $\gamma = 2$, $\delta = 0.1$, $\sigma_o^2 = 10^{-6}$. (b) $10\log\{\text{MSD}_c(n)\}$ for $\gamma = 4$, $\delta = 0.2$, $\sigma_o^2 = 10^{-8}$.

Fig. 5. Mean-square deviations $10\log_{10}\text{MSD}_c(n)$. Theory derived from $\text{MSD}_1(n)$ before convergence of $\boldsymbol{W}_1(n)$ and from (44) after that. Simulations averaged over 50 Monte Carlo runs.

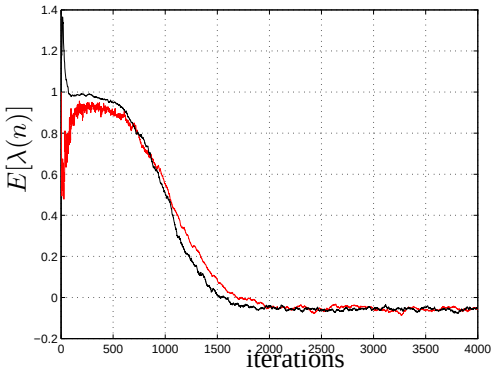


(a) $E\{\lambda_1(n)\}$ for $\gamma = 2$, $\delta = 0.1$, $\sigma_o^2 = 10^{-2}$, $\mu_\lambda = 1$.

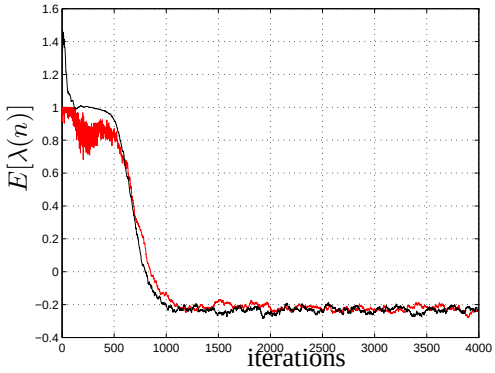


(b) $E[\{\lambda_1(n)\}$ for $\gamma = 4$, $\delta = 0.3$, $\sigma_o^2 = 10^{-4}$, $\mu_\lambda = 300$.

Fig. 6. $E[\lambda_1(n)]$ for different parameter values. Simulations (average of 50 Monte Carlo runs) in red (ragged curve). Theory (derived elsewhere) in black.

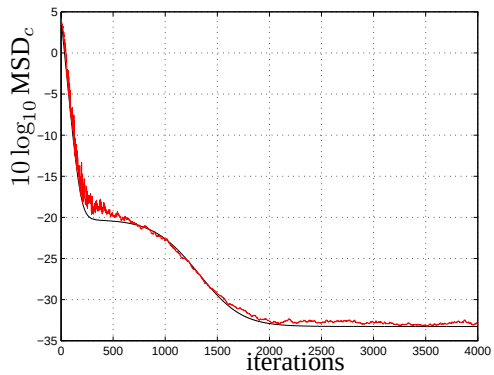


(a) $E\{\lambda_1(n)\}$ and $E\{\lambda_o(n)\}$ for $\gamma = 2$, $\delta = 0.1$, $\sigma_o^2 = 10^{-2}$ and $\mu_\lambda = 1$.

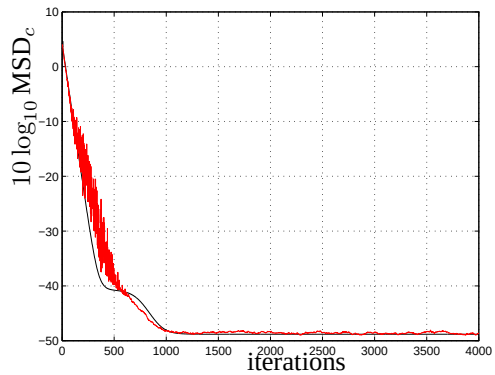


(b) $E[\{\lambda_1(n)\}$ and $E\{\lambda_o(n)\}$ for $\gamma = 4$, $\delta = 0.3$, $\sigma_o^2 = 10^{-4}$ and $\mu_\lambda = 300$.

Fig. 7. Behavior of $E[\lambda_1(n)]$ (in red, ragged curve) as compared to the optimum behavior of $E[\lambda_o(n)]$ (in black, smoother curve). All plots are simulation results (average of 50 Monte Carlo runs).

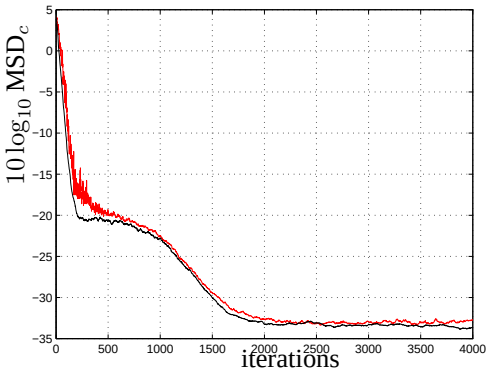


(a) $10\log_{10}\text{MSD}_c$ for $\gamma = 2$, $\delta = 0.1$, $\sigma_o^2 = 10^{-2}$, $\mu_\lambda = 1$.

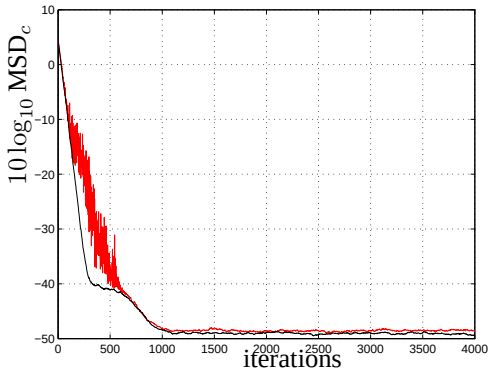


(b) $10\log_{10}\text{MSD}_c$, $\gamma = 4$, $\delta = 0.3$, $\sigma_o^2 = 10^{-4}$, $\mu_\lambda = 300$.

Fig. 8. $10\log_{10}\text{MSD}_c$. Simulations (average of 50 runs, in red, ragged curve) using $\lambda(n) = \lambda_1(n)$ from (45) for the same parameter values as in Fig. 6. Theory derived elsewhere in black (smooth curve).

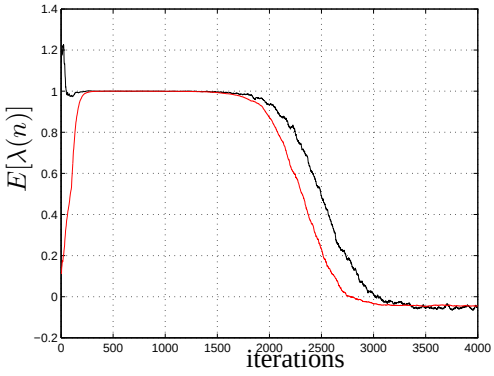


(a) $10\log_{10}\text{MSD}_c$ for $\gamma = 2$, $\delta = 0.1$, $\sigma_o^2 = 10^{-2}$, $\mu_\lambda = 1$.

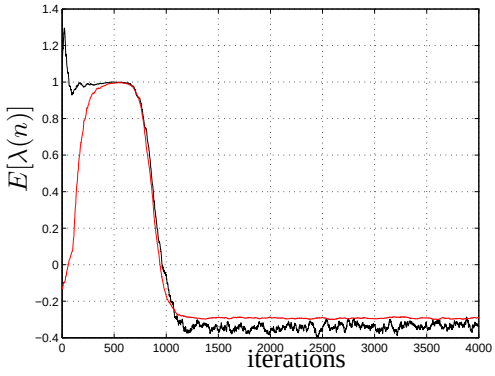


(b) $10\log_{10}\text{MSD}_c$, $\gamma = 4$, $\delta = 0.3$, $\sigma_o^2 = 10^{-4}$, $\mu_\lambda = 300$.

Fig. 9. Comparison between $10\log_{10}\text{MSD}_c$ using $\lambda(n) = \lambda_1(n)$ from (45) (ragged curve) and using $\lambda_o(n)$ for the same parameter values as in Fig. 6 (smooth curve). Monte Carlo simulations averaged over 50 runs.



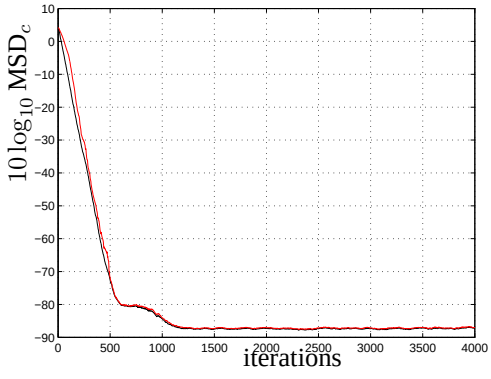
(a) $E\{\lambda_2(n)\}$ and $E\{\lambda_o(n)\}$ for $\gamma = 1$, $\delta = 0.1$, $\sigma_o^2 = 10^{-6}$, $\sigma_u^2 = 1$.



(b) $E\{\lambda_2(n)\}$ and $E\{\lambda_o(n)\}$ for $\gamma = 2$, $\delta = 0.4$, $\sigma_o^2 = 10^{-8}$, $\sigma_u^2 = 1$.

Fig. 10. Simulation results (average of 50 MC's) on $E\{\lambda_2(n)\}$ and $E\{\lambda_o(n)\}$ for $K = 100$. Curves in black (more ragged) for $\lambda(n) = \lambda_o(n)$. Curves in red (smoother) for $\lambda(n) = \lambda_2(n)$.

(a) $10\log_{10}\text{MSD}_c(n)$ for $\gamma = 1$, $\delta = 0.1$, $\sigma_o^2 = 10^{-6}$, $\sigma_u^2 = 1$.



(b) $10\log_{10}\text{MSD}_c(n)$ for $\gamma = 2$, $\delta = 0.4$, $\sigma_o^2 = 10^{-8}$, $\sigma_u^2 = 1$.

Fig. 11. Mean-square deviation $10\log_{10}\text{MSD}_c(n)$ using the error power based scheme. Curves in black (slightly below) for $\lambda(n) = \lambda_o(n)$. Curves in red (slightly above) for $\lambda(n) = \lambda_2(n)$.