Correspondences

Transient Analysis of Adaptive Affine Combinations

Suleyman S. Kozat, Alper T. Erdogan, Andrew C. Singer, and Ali H. Sayed

***Abstract*—In this correspondence, we provide a transient analysis of an affinely constrained mixture method that adaptively combines the outputs of adaptive filters running in parallel on the same task. The affinely constrained mixture is adapted using a stochastic gradient update to minimize the square of the prediction error. Although we specifically carry out the transient analysis for a combination of two equal length adaptive filters trying to learn a linear model working on real valued data, we also provide the final equations and the necessary extensions in order to generalize the transient analysis to mixtures combining more than two filters; using Newton based updates to train the mixture weights; working on complex valued data; or unconstrained mixtures. The derivations are generic such that the constituent filters can be trained using unbiased updates including the least-mean squares or recursive least squares updates. This correspondence concludes with numerical examples and final remarks.**

***Index Terms*—Adaptive filtering, least-mean squares, mixture methods, transient analysis.**

I. Introduction

In this correspondence, we study the transient behavior of linear mixture methods that combine outputs of adaptive filters running in parallel on the same task. Such adaptive mixture methods may be used in order to improve the steady-state and/or convergence performance over the constituent algorithms in the mixture [1]–[4]. This framework has two stages [1]–[4]. The first stage has adaptive filters that run in parallel to model a desired signal assumed to be generated by a linear model [5]. In the second stage, the outputs of the filters are linearly combined to yield the output of the system. Although the outputs of the first stage filters are linearly combined in the second stage, these combination weights can be adapted in a highly nonlinear manner [1]–[3]. One can use unconstrained [4], affinely constrained [2] or convexly constrained [1], [3], [6] mixture weights in the second stage to construct the final output. Under such constraints, one can adapt the combination weights using a variety of means including stochastic gradient updates [1] or Newton (or quasi-Newton) based updates [7]. We refer the reader to [1]–[4] for a general discussion of linear mixtures, possible configurations and an extended set of references.

We first concentrate on a mixture of two adaptive filters having equal lengths working on real valued data; the second stage mixture is constrained to be affine; and the mixture weights are trained using a particular stochastic gradient update, i.e., the least mean squares (LMS) algorithm. Our transient analysis is generic with respect to how the constituent filters are trained as long as they are of the same length and perform unbiased estimation. This transient analysis can be extended to mixtures having more than two constituent filters working on complex valued data; using different adaptation methods for each constituent filter; using Newton (or quasi-Newton) based updates to train the second stage weights; or to unconstrained mixtures. We provide these extensions and show how our derivations should be modified for these cases with the corresponding results as remarks in the correspondence. Note that although the transient analysis studied here may be extended for these configurations, one needs to provide the corresponding time-varying auto- and cross-correlation functions between the *a priori* errors [5] of the constituent filters. Such time-varying cross-correlation functions between the same length filters using the LMS, the least-mean fourth (LMF), or the recursive least squares (RLS) updates, combinations of these updates, or blind methods such as the constant modulus algorithm can be readily derived following the approaches in [1], [3], and [5] in certain scenarios.

A transient analysis of the adaptive affine combination studied here was first carried out in [8] and [9] (and then detailed in [10]) specifically for a combination of two adaptive filters. However, our analysis and the resulting conclusions differ from [8] and [10] in a number of important ways. As in [8] and [10], we first observe that the affine combination can be represented as an unconstrained stochastic update on a single parameter by using particular input and desired signal characterization. However, in order to generalize this interpretation to mixtures having more than two adaptive filters, unlike [8] and [10], we define the *a priori* error with respect to the second stage combination, i.e., we consider the second stage as another adaptive linear filter working on nonstationary outputs of the first stage filters. This definition then allows us to derive an energy conservation relation as in [5] to describe the time evolution of the affine combination weight error. As shown in Section III, such an energy conservation relation can be generalized to mixtures using more than two filters, affine or unconstrained adaptive methods or quasi-Newton algorithms, yielding the transient analysis for these configurations as well. In Section III, we provide the main differences between our analysis and that presented in [8].

After we introduce the basic system setup in Section II, we continue with the transient analysis of the affine combination of two adaptive filters having the same length. The affine combination parameter is trained using the LMS update. We also provide the corresponding conditions for convergence in the MSE sense. In Sections II and III, we provide the corresponding modifications to extend the derivations to mixtures having more than two constituent filters, using unconstrained weights or the case where the second stage combination weights are trained using Newton based updates. We present numerical examples in Section IV to test the validity of the derivations. The correspondence concludes with some remarks about the transient analysis.

Manuscript received November 03, 2010; revised March 27, 2011 and June 16, 2011; accepted June 16, 2011. Date of publication July 18, 2011; date of current version November 16, 2011. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Kainam Wong. This work is supported in part by TUBITAK Career Award, Contract No. 104E073 and No. 108E195, and by TUBA Outstanding Young Researcher Award program, and in part by NSF Grants ECS-0601266 and CCF-0942936.

S. S. Kozat and A. T. Erdogan are with the Electrical and Electronic Engineering Department, Koc University, Istanbul 06660, Turkey (e-mail: skozat@ku.edu.tr; alperdogan@ku.edu.tr).

A. C. Singer is with the Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801 USA (e-mail: acsinger@uiuc.edu).

A. H. Sayed is with the Department of Electrical Engineering, University of California at Los Angeles, Los Angeles, CA 90095 USA (e-mail: sayed@ee.ucla.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSP.2011.2162325

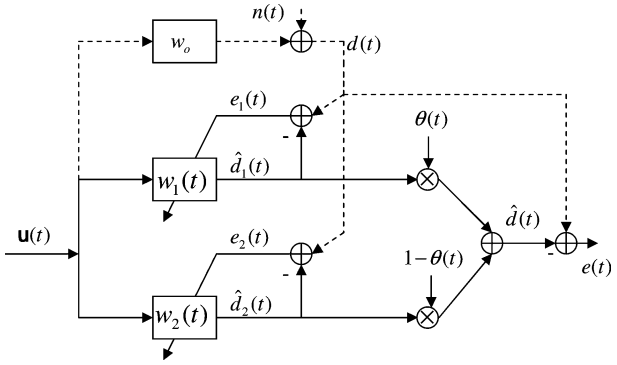

Fig. 1. An affine mixture of two adaptive filters working in parallel to model a desired signal.

II. System Description

The framework[1] we consider has two stages as shown in Fig. 1. The first stage has two constituent adaptive filters working in parallel to model a desired signal $d(t)$. The desired signal is assumed to be generated by a stationary discrete time linear model $d(t) = \boldsymbol{w}_o^T \boldsymbol{u}(t) + n(t)$, where $\boldsymbol{u}(t) \in \mathbb{R}^M$ is a zero mean stationary vector process with $\boldsymbol{Q} \triangleq E[\boldsymbol{u}(t)\boldsymbol{u}^T(t)]$, $n(t)$ is an i.i.d. noise process independent of $\boldsymbol{u}(t)$ with $\sigma_n^2 \triangleq E[n^2(t)]$ and $\boldsymbol{w}_o \in \mathbb{R}^M$ is the unknown system vector. Note that we use a stationary linear model $\boldsymbol{w}_o$ instead of the widely used random walk model [5] since our goal is to analyze the transient behavior of the combination algorithms. Each constituent filter updates a weight vector $\boldsymbol{w}_i(t) \in \mathbb{R}^M$ and produces estimates, $\hat{d}_i(t) = \boldsymbol{w}_i^T(t)\boldsymbol{u}(t)$, $i = 1, 2$, respectively. For each filter we also define estimation, *a priori* and *a posteriori* errors as

$$\begin{aligned} e_i(t) &= d(t) - \hat{d}_i(t) \\ e_{a,i}(t) &= [\boldsymbol{w}_o - \boldsymbol{w}_i(t)]^T \boldsymbol{u}(t) \\ e_{p,i}(t) &= [\boldsymbol{w}_o - \boldsymbol{w}_i(t+1)]^T \boldsymbol{u}(t) \end{aligned}$$

assuming $\boldsymbol{w}_1(t)$ and $\boldsymbol{w}_2(t)$ produce unbiased estimators of $\boldsymbol{w}_o$. Hence, for each filter we have $\hat{d}_i(t) = \boldsymbol{w}_o^T \boldsymbol{u}(t) - e_{a,i}(t)$, and $e_i(t) = e_{a,i}(t) + n(t)$. We also have $J_{ii}(t) \triangleq E[e_i^2(t)]$, $J_{\mathrm{ex},ii}(t) \triangleq E[e_{a,i}^2(t)]$ and their limiting values (if they exist) $J_{ii} \triangleq \lim_{t \to \infty} J_{ii}(t)$, $J_{\mathrm{ex},ii} \triangleq \lim_{t \to \infty} J_{\mathrm{ex},ii}(t)$, respectively. We further have $J_{\mathrm{ex},12}(t) \triangleq E[e_{a,1}(t)e_{a,2}(t)]$ and (if it exists) $J_{\mathrm{ex},12} \triangleq \lim_{t \to \infty} J_{\mathrm{ex},12}(t)$. The limiting values $J_{\mathrm{ex},11}$, $J_{\mathrm{ex},22}$ and $J_{\mathrm{ex},12}$ are derived in [1], [3] for a wide range of adaptation methods.

The second stage of the framework is the mixture stage. Here, the outputs of the two constituent adaptive filters are combined to produce the final output. Imposing an affine constraint on the combination weights, the final output is generated as $\hat{d}(t) = \theta(t)\hat{d}_1(t) + (1-\theta(t))\hat{d}_2(t) = \hat{d}_2(t) + \theta(t)(\hat{d}_1(t) - \hat{d}_2(t))$. The combination weight $\theta(t)$ is updated using a stochastic gradient update to minimize the square of the final estimation error, $e(t) = d(t) - \hat{d}(t)$, yielding

$$\theta(t+1) = \theta(t) + \mu e(t)(\hat{d}_1(t) - \hat{d}_2(t)). \tag{1}$$

[1]All vectors are column vectors represented by boldface lowercase letters, $(\cdot)^T$ is the transpose operation, $(\cdot)^H$ is the conjugate transpose operation. Matrices are represented with boldface capital letters. For $\boldsymbol{w}$, $\|\boldsymbol{w}\|_{\boldsymbol{\Sigma}}^2 = \boldsymbol{w}^H \boldsymbol{\Sigma} \boldsymbol{w}$ is the weighted l_2-norm for a positive semidefinite matrix $\boldsymbol{\Sigma}$. For a vector $\boldsymbol{w}$, $w^{(i)}$ is the ith entry. For a matrix $\boldsymbol{R}$, $\mathrm{tr}(\boldsymbol{R})$ is the trace and $R^{(i,j)}$ is the (i,j)th entry. The vector (or the matrix) $\mathbf{1}$ represents a vector (or a matrix) of all ones where the size is understood from the context.

We point out that this can be seen as a stochastic gradient update with a single parameter $\theta(t)$, with the desired signal as $d(t) - \hat{d}_2(t) = e_{a,2}(t) + n(t)$ and the input data as $\hat{d}_1(t) - \hat{d}_2(t) = e_{a,2}(t) - e_{a,1}(t)$ [8]. Assuming convergence, we have limiting value $\theta_o = \lim_{t \to \infty} E[\theta(t)] = \frac{J_{\mathrm{ex},22} - J_{\mathrm{ex},12}}{J_{\mathrm{ex},11} + J_{\mathrm{ex},22} - 2J_{\mathrm{ex},12}}$, from [8].

Remark 1: To extend the affinely constrained algorithm to combine outputs of m constituent adaptive filters, one can define the input regressor as $\boldsymbol{y}(t) \triangleq [\hat{d}_1(t), \ldots, \hat{d}_m(t)]^T$, the final output as $\hat{d}(t) = \boldsymbol{\theta}^T(t)\boldsymbol{y}(t)$, with the constraint $\boldsymbol{\theta}^T(t)\mathbf{1} = 1$. Note that by taking $\theta^{(m)}(t) = 1 - \sum_{k=1}^{m-1} \theta^{(k)}(t)$, this is an unconstrained combination with the weight vector $\boldsymbol{\beta}(t) \triangleq [\theta^{(1)}(t), \ldots, \theta^{(m-1)}(t)]$, the input regressor $\boldsymbol{\kappa}(t) = [\hat{d}_1(t) - \hat{d}_m(t), \ldots, \hat{d}_{m-1}(t) - \hat{d}_m(t)]^T$ and the desired signal $d(t) - \hat{d}_m(t)$. Then, the stochastic gradient update is given by: $\boldsymbol{\beta}(t+1) = \boldsymbol{\beta}(t) + \mu e(t)\boldsymbol{\kappa}(t)$ where $e(t) = d(t) - \boldsymbol{\beta}^T(t)\boldsymbol{\kappa}(t)$ [4]. The corresponding limiting values $\lim_{t \to \infty} E[\boldsymbol{\beta}(t)]$ and $\lim_{t \to \infty} E[e^2(t)]$ for certain scenarios are given in [4] assuming convergence.

Remark 2: With the same notation as in Remark 1, one can define an unconstrained update directly on $\boldsymbol{\theta}(t)$, without the constraint $\boldsymbol{\theta}^T(t)\mathbf{1} = 1$, as $\boldsymbol{\theta}(t+1) = \boldsymbol{\theta}(t) + \mu e(t)\boldsymbol{y}(t)$, where $e(t) = d(t) - \boldsymbol{\theta}^T(t)\boldsymbol{y}(t)$, to combine m constituent filters [4]. The corresponding limiting values $\lim_{t \to \infty} E[\boldsymbol{\theta}(t)]$ and $\lim_{t \to \infty} E[e^2(t)]$, and their relation to affine or convex combinations are given in [4] assuming convergence under different scenarios.

We next provide a transient analysis under this system description.

III. Transient Analysis

We next analyze the transient behavior of the stochastic gradient update on the affine combination parameter given in (1) with the input regressor $[e_{a,2}(t) - e_{a,1}(t)]$ and the desired signal $n(t) + e_{a,2}(t)$. Note that the time-varying optimal combination weight that minimizes the estimation MSE at each time instant can be written

$$\begin{aligned} \theta_o(t) &\triangleq \min_{\theta} E\left(\left\{ [n(t) + e_{a,2}(t)] - \theta[e_{a,2}(t) - e_{a,1}(t)] \right\}^2 \right) \\ &= \frac{E\left\{ [n(t) + e_{a,2}(t)][e_{a,2}(t) - e_{a,1}(t)] \right\}}{E\left\{ [e_{a,2}(t) - e_{a,1}(t)]^2 \right\}} \\ &= \frac{J_{\mathrm{ex},22}(t) - J_{\mathrm{ex},12}(t)}{J_{\mathrm{ex},11}(t) + J_{\mathrm{ex},22}(t) - 2J_{\mathrm{ex},12}(t)}. \end{aligned} \tag{2}$$

Subtracting the time-varying optimal combination weight from both sides of (1) yields

$$[\theta_o(t) - \theta(t+1)] = [\theta_o(t) - \theta(t)] - \mu e(t)[e_{a,2}(t) - e_{a,1}(t)].$$

After defining *a priori* and *a posteriori* errors for the second stage as

$$e_a(t) = [\theta_o(t) - \theta(t)][e_{a,2}(t) - e_{a,1}(t)] \tag{3}$$

$$e_p(t) = [\theta_o(t) - \theta(t+1)][e_{a,2}(t) - e_{a,1}(t)], \tag{4}$$

we get the energy conservation relation

$$\begin{aligned} &\|\theta_o(t) - \theta(t+1)\|^2 + \frac{\|e_a(t)\|^2}{\|[e_{a,2}(t) - e_{a,1}(t)]\|^2} \\ &\qquad = \|\theta_o(t) - \theta(t)\|^2 + \frac{\|e_p(t)\|^2}{\|[e_{a,2}(t) - e_{a,1}(t)]\|^2}, \end{aligned} \tag{5}$$

after some algebra. We point out that in [8], the *a priori* error is defined with respect to $\boldsymbol{w}_o$ and $\boldsymbol{u}(t)$ which yields $e_a(t) = \theta(t)e_{a,1}(t) + (1-\theta(t))e_{a,2}(t)$ in [8]. Note that this *a priori* error definition is different than our definition of *a priori* error in (3). By defining *a priori* and *a posteriori* errors with respect to the time-varying optimal combination parameter $\theta_o(t)$ and nonstationary input regressor, $[e_{a,2}(t) - e_{a,1}(t)]$, we are able to derive and use the energy conservation relation to perform transient analysis which can be readily extended to mixtures having more than two constituent filters or using Newton based updates. Defining

$$e_o(t) \triangleq n(t) + e_{a,2}(t) - \theta_o(t)[e_{a,2}(t) - e_{a,1}(t)], \tag{6}$$

i.e., $d(t) - \hat{d}_2(t) = \theta_o(t)[e_{a,2}(t) - e_{a,1}(t)] + e_o(t)$, and after some algebra, (5) yields

$$\begin{aligned}||\theta_o(t) - \theta(t+1)||^2 = &||\theta_o(t) - \theta(t)||^2 \\ &+\mu^2[e_{a,2}(t)-e_{a,1}(t)]^4||\theta_o(t)-\theta(t)||^2 \\ &+ \mu^2[e_{a,2}(t) - e_{a,1}(t)]^2 e_o^2(t) \\ &-2\mu||\theta_o(t)-\theta(t)||^2[e_{a,2}(t)-e_{a,1}(t)]^2 \\ &+ 2\mu^2[e_{a,2}(t) - e_{a,1}(t)]^2 e_a(t)e_o(t) \\ &- 2\mu e_o(t)e_a(t)\end{aligned} \tag{7}$$

and

$$\begin{aligned}&||\theta_o(t+1) - \theta(t+1)||^2 \\ &\quad + 2[\theta_o(t) - \theta_o(t+1)][\theta_o(t+1) - \theta(t+1)] \\ &\quad + ||\theta_o(t) - \theta_o(t+1)||^2 \\ &\quad = ||\theta_o(t)-\theta(t)||^2+\mu^2[e_{a,2}(t)-e_{a,1}(t)]^4||\theta_o(t)-\theta(t)||^2 \\ &\qquad + \mu^2[e_{a,2}(t) - e_{a,1}(t)]^2 e_o^2(t) \\ &\qquad - 2\mu||\theta_o(t) - \theta(t)||^2[e_{a,2}(t) - e_{a,1}(t)]^2 \\ &\qquad + 2\mu^2[e_{a,2}(t) - e_{a,1}(t)]^2 e_a(t)e_o(t) \\ &\qquad - 2\mu e_o(t)e_a(t).\end{aligned} \tag{8}$$

The recursion in (8) is different from (16) of [8] since we have defined the recursion with respect to $\theta_o(t+1) - \theta(t+1)$ unlike in (16) where $\delta\eta(t+1) = \theta_o(t) - \theta(t+1)$. Assuming $E[\theta(t+1)] = \theta_o(t+1)$, we have $E[||\theta_o(t) - \theta(t+1)||^2] = E[||\theta_o(t+1) - \theta(t+1)||^2] + E[||\theta_o(t+1) - \theta_o(t)||^2]$. To take the expectation of both sides of (8), we make separation assumptions for the terms related to $[e_{a,1}(t) - e_{a,2}(t)]$ and $\theta(t)$ in (8) similar to the separation assumption discussed in [1], [5]. We point out that $e_o(t)$ is uncorrelated with $[e_{a,2}(t) - e_{a,1}(t)]$ from (2) and (6). However, we also assume independence of $e_o(t)$ from $[e_{a,2}(t) - e_{a,1}(t)]$ and $\theta(t)$ yielding

$$\begin{aligned}E[||\tilde{\theta}(t+1)||^2] = &\left\{1 - 2\mu E[(e_{a,1}(t) - e_{a,2}(t))^2]\right. \\ &\left.+\mu^2 E[(e_{a,1}(t) - e_{a,2}(t))^4]\right\} E[||\tilde{\theta}(t)||^2] \\ &+ \mu^2 E[(e_{a,1}(t) - e_{a,2}(t))^2]E[e_o^2(t)] - \alpha(t)\end{aligned} \tag{9}$$

where we define $\tilde{\theta}(t) \triangleq \theta_o(t) - \theta(t)$ and $\alpha(t) \triangleq ||\theta_o(t+1) - \theta_o(t)||^2$. Note that since $\lim_{t\to\infty}\theta_o(t) = \frac{J_{\mathrm{ex},22} - J_{\mathrm{ex},12}}{J_{\mathrm{ex},11}+J_{\mathrm{ex},22}-2J_{\mathrm{ex},12}}$, we have $\lim_{t\to\infty}\alpha(t) = 0$ assuming convergence. We also have $E[(e_{a,1}(t) - e_{a,2}(t))^2] = J_{\mathrm{ex},11}(t) + J_{\mathrm{ex},22}(t) - 2J_{\mathrm{ex},12}(t)$ in (9). In order to calculate $E[(e_{a,1}(t) - e_{a,2}(t))^4]$, we assume that $e_{a,i}(t)$ are Gaussian distributed. With this assumption, (9) yields (10), shown at the bottom of the page. This recursion will converge if

$$\left|\left\{1 - 2\mu[J_{\mathrm{ex},11}(t) + J_{\mathrm{ex},22}(t) - 2J_{\mathrm{ex},12}(t)] + 3\mu^2[J_{\mathrm{ex},11}(t) + J_{\mathrm{ex},22}(t) - 2J_{\mathrm{ex},12}(t)]^2\right\}\right| < 1$$

which yields

$$0 < \mu < \frac{2}{3\left[J_{\mathrm{ex},11}(t) + J_{\mathrm{ex},22}(t) - 2J_{\mathrm{ex},12}(t)\right]}. \tag{11}$$

Based on the recursion in (10), the norm of the weight error vector can be written as [5], [11]

$$\begin{aligned}E[||\tilde{\theta}(t)||^2] = &\Phi(t-1,0)E[||\tilde{\theta}(0)||^2] \\ &+\sum_{k=1}^{t-1}\Phi(t-1,k)\varphi(k-1) + \varphi(t-1)\end{aligned}$$

where $\Phi(t_1,t_2) \triangleq A(t_1)A(t_1-1)\ldots A(t_2+1)A(t_2)$. Note that the time-varying formulations for $J_{\mathrm{ex},ii}(t)$ are given in [5, Ch. 9] for a wide range of adaptation methods under the studied model, including the LMS and RLS updates. The time varying cross correlation function between the constituent filters $J_{\mathrm{ex},12}(t)$ can be derived for two LMS filters (or for an LMS filter and an RLS filter) following similar lines to [5, Ch. 9].

$$\begin{aligned}E[||\tilde{\theta}(t+1)||^2] = &\underbrace{\left\{1 - 2\mu[J_{\mathrm{ex},11}(t) + J_{\mathrm{ex},22}(t) - 2J_{\mathrm{ex},12}(t)] + 3\mu^2[J_{\mathrm{ex},11}(t) + J_{\mathrm{ex},22}(t) - 2J_{\mathrm{ex},12}(t)]^2\right\}}_{\triangleq A(t)} E[||\tilde{\theta}(t)||^2] \\ &+ \underbrace{\mu^2[J_{\mathrm{ex},11}(t)J_{\mathrm{ex},22}(t) - J_{\mathrm{ex},12}^2(t)] + \mu^2\sigma_n^2[J_{\mathrm{ex},11}(t) + J_{\mathrm{ex},22}(t) - 2J_{\mathrm{ex},12}(t)] - \alpha(t)}_{\triangleq \varphi(t)}.\end{aligned} \tag{10}$$

Given this recursion for the affine combination weight error, one can derive the corresponding expression for the excess MSE of the affine combination. Using the separation assumption, we have

$$E[e_a^2(t)] = E\left\{[\theta_o(t)-\theta(t)]^2[e_{a,1}(t)-e_{a,2}(t)]^2\right\} = E\left\{[\theta_o(t)-\theta(t)]^2\right\} E\left\{[e_{a,1}(t)-e_{a,2}(t)]^2\right\} \quad (12)$$

where the second line follows from the separation assumption. With μ satisfying (11), we have

$$\lim_{t\longrightarrow\infty} E[e_a^2(t)] = \lim_{t\longrightarrow\infty} E\left\{[\theta_o(t)-\theta(t)]^2\right\} \lim_{t\longrightarrow\infty} E\left\{[e_{a,1}(t)-e_{a,2}(t)]^2\right\} \quad (13)$$

$$= \frac{\mu\left(\sigma_n^2+\frac{J_{\mathrm{ex},11}J_{\mathrm{ex},22}-J_{\mathrm{ex},12}^2}{J_{\mathrm{ex},11}+J_{\mathrm{ex},22}-2J_{\mathrm{ex},12}}\right)}{2-3\mu(J_{\mathrm{ex},11}+J_{\mathrm{ex},22}-2J_{\mathrm{ex},12})} \times (J_{\mathrm{ex},11}+J_{\mathrm{ex},22}-2J_{\mathrm{ex},12}). \quad (14)$$

To compare (7) with the excess MSE of a converged affine combination filter trained using the LMS update with input regressor $[e_{a,2}(t)-e_{a,1}(t)]$ and desired signal $[e_{a,2}(t)+n(t)]$ given in [4], under different assumptions, we obtain

$$\frac{\mu\left(\sigma_n^2+\frac{J_{\mathrm{ex},11}J_{\mathrm{ex},22}-J_{\mathrm{ex},12}^2}{J_{\mathrm{ex},11}+J_{\mathrm{ex},22}-2J_{\mathrm{ex},12}}\right)(J_{\mathrm{ex},11}+J_{\mathrm{ex},22}-2J_{\mathrm{ex},12})}{2-\mu(J_{\mathrm{ex},11}+J_{\mathrm{ex},22}-2J_{\mathrm{ex},12})} \quad (15)$$

since $(J_{\mathrm{ex},11}+J_{\mathrm{ex},22}-2J_{\mathrm{ex},12})$ is the power of the input regressor, i.e., $[e_{a,2}(t)-e_{a,1}(t)]$, in the limit, and

$$\lim_{t\longrightarrow\infty} E[e_o^2(t)] = \left(\frac{\sigma_n^2+J_{\mathrm{ex},11}J_{\mathrm{ex}},22-J_{\mathrm{ex},12}^2}{J_{\mathrm{ex},11}+J_{\mathrm{ex},22}-2J_{\mathrm{ex},12}}\right)$$

from [5] and [8]. We emphasize that the transient analysis resulting in (7) is slightly different from (8) since the transient analysis carried out in this correspondence is more involved and requires stronger assumptions.

Remark 3: When Newton (or quasi-Newton) based methods are used instead of the LMS update to train the combination weights, the norms in (5) should be changed to weighted norms in terms of

$$p(t) \triangleq \frac{1}{\lambda^t\delta+\sum_{i=1}^t \lambda^{t-i}[e_{a,2}(i)-e_{a,1}(i)]^2}.$$

Here, $0<\lambda\leq 1$ is the forgetting factor and δ^{-1} is the initial value for $p(t)$. If we use assume that $p(t)$ is uncorrelated with the mixture weights and regressor vectors [5], then (9) yields

$$E[\|\tilde{\theta}(t+1)\|^2] = \{1-2\mu E[p(t)]E[(e_{a,1}(t)-e_{a,2}(t))^2] + \mu^2 E[p^2(t)]E[(e_{a,1}(t)-e_{a,2}(t))^4]\} E[\|\tilde{\theta}(t)\|^2] + \mu^2 E[p^2(t)]E[(e_{a,1}(t)-e_{a,2}(t))^2]E[e_o^2(t)]-\alpha(t). \quad (16)$$

To calculate $E[p(t)]$ and $E[p^2(t)]$, we use approximations similar to ones used in [5] as

$$E[p(t)] \approx \frac{1}{\lambda^t\delta+E\{\sum_{i=1}^t \lambda^{t-i}[e_{a,2}(i)-e_{a,1}(i)]^2\}}$$

and $E[p^2(t)] \approx E[p(t)]^2$.

Remark 4: When we use an affine combination of m constituent filters of the same length performing unbiased estimation, the norm of the input regressor, $[e_{a,2}(t)-e_{a,1}(t)]^2$, in the energy conservation relation (8) should be replaced by $\|\boldsymbol{\kappa}(t)\|^2$. The time-varying optimal affine combination, i.e., $\boldsymbol{\beta}_o(t)=\arg\min_{\boldsymbol{\beta}} E\left\{[n(t)+e_{a,m}(t)-\boldsymbol{\beta}^T\boldsymbol{\kappa}(t)]^2\right\} = \boldsymbol{R}^{-1}(t)\boldsymbol{p}(t)$, where $\boldsymbol{R}(t) \triangleq E[\boldsymbol{\kappa}(t)\boldsymbol{\kappa}^T(t)]$, $\boldsymbol{p}(t) \triangleq E\{[n(t)+e_{a,m}(t)]\boldsymbol{\kappa}(t)\}$, can be derived as in [4] including $E[e_o^2(t)] = \sigma_n^2 + J_{\mathrm{ex},mm}(t) - \boldsymbol{p}^T(t)\boldsymbol{R}^{-1}(t)\boldsymbol{p}(t)$ and $\lim_{t\longrightarrow\infty} E[e_o^2(t)]$. For equal length filters performing unbiased estimation, we have $R^{(i,j)}(t) = J_{\mathrm{ex},ij}(t) - J_{\mathrm{ex},im}(t) - J_{\mathrm{ex},jm}(t) + J_{\mathrm{ex},mm}(t)$ and $p^{(i)}(t) = J_{\mathrm{ex},im}(t) - J_{\mathrm{ex},mm}(t)$. For this configuration, the variance relation yields

$$E[\|\tilde{\boldsymbol{\beta}}(t+1)\|_{\boldsymbol{\Sigma}}^2] = E[\|\tilde{\boldsymbol{\beta}}(t)\|_{\boldsymbol{\Sigma}}^2]+\mu^2 E[e_o^2(t)]E[\|\boldsymbol{\kappa}(t)\|_{\boldsymbol{\Sigma}}^2]-\nu(t) \quad (17)$$

where $\boldsymbol{\Sigma} \triangleq \boldsymbol{\Sigma} - \mu\boldsymbol{\Sigma}E[\boldsymbol{\kappa}(t)\boldsymbol{\kappa}^T(t)]-\mu E[\boldsymbol{\kappa}(t)\boldsymbol{\kappa}^T(t)]\boldsymbol{\Sigma} + \mu^2 E[\|\boldsymbol{\kappa}(t)\|_{\boldsymbol{\Sigma}}^2\boldsymbol{\kappa}(t)\boldsymbol{\kappa}^T(t)]$, $\boldsymbol{\Sigma}$ is a positive definite weighting matrix, $\tilde{\boldsymbol{\beta}}(t) \triangleq \boldsymbol{\beta}_o(t)-\boldsymbol{\beta}(t)$ and $\nu(t) \triangleq \|\boldsymbol{\beta}_o(t+1)-\boldsymbol{\beta}_o(t)\|_{\boldsymbol{\Sigma}}^2$. We point out that although $\boldsymbol{\kappa}(t)$ can be assumed to be Gaussian, $E[\boldsymbol{\kappa}(t)\boldsymbol{\kappa}^T(t)]$ is time varying such that (10) cannot be readily diagonalized while preserving time evaluation. Hence the state space representation for(10) should be derived similar to the non-Gaussian regressor case as in [5]. Furthermore, while constructing the corresponding state space representation, the left hand side of (10) should be weighted using $\boldsymbol{R}(t+1)$ and the right hand side with $\boldsymbol{R}(t)$. We observe that since $\|\tilde{\boldsymbol{\beta}}(t+1)\|$ can be assumed to be bounded and $\|\boldsymbol{R}(t+1)-\boldsymbol{R}(t)\|$ is relatively small, using $\boldsymbol{R}(t)$ on both sides of (10) yields the time evaluation of the corresponding state space representation (which yields $E[\|\tilde{\boldsymbol{\beta}}(t)\|^2]$) and provides satisfactory results in our simulations.

Moreover, instead of using the full state space representation, we observe that using $\mathrm{tr}(\boldsymbol{AB}) \approx \mathrm{tr}(\boldsymbol{A})\mathrm{tr}(\boldsymbol{B})$ for affine combination of m filters (9) yields[2]

$$E[\|\tilde{\boldsymbol{\beta}}(t+1)\|^2] = \left\{1-2\mu\mathrm{tr}[\boldsymbol{R}(t)]+\mu^2\{\mathrm{tr}[\boldsymbol{R}(t)]^2+2\mathrm{tr}[\boldsymbol{R}^2(t)]\}\right\}E[\|\tilde{\boldsymbol{\beta}}(t)\|^2] + \mu^2\mathrm{tr}[\boldsymbol{R}(t)][\sigma_n^2+J_{\mathrm{ex},mm}(t)-\boldsymbol{p}^T(t)\boldsymbol{R}^{-1}(t)\boldsymbol{p}(t)]-\nu'(t) \quad (18)$$

assuming $e_{a,i}(t)$ are Gaussian distributed and $\nu(t) \triangleq \|\boldsymbol{\beta}_o(t+1)-\boldsymbol{\beta}_o(t)\|^2$. In our experiments, we observe a close agreement between (11) and simulations. For unconstrained combination, $[e_{a,2}(t)-e_{a,1}(t)]^2$ in (18) should be replaced by $\|\boldsymbol{y}(t)\|^2$ and $\boldsymbol{\theta}_o(t)=\arg\min_{\boldsymbol{\theta}} E\left\{[d(t)-\boldsymbol{\theta}^T\boldsymbol{y}(t)]^2\right\}$, which can be derived as in [4], including $E[e_o^2(t)]$ and $\lim_{t\longrightarrow\infty} E[e_o^2(t)]$. After these replacements the derivations follows similar lines as the affine combination of m filters to yield recursion as in (10), where $\tilde{\boldsymbol{\beta}}(t)$ is replaced by $\tilde{\boldsymbol{\theta}}(t) = \boldsymbol{\theta}_o(t)-\boldsymbol{\theta}(t)$, $\boldsymbol{R}(t)$ is replaced by $E[\boldsymbol{y}(t)\boldsymbol{y}^T(t)]$ and $\nu(t)$

[2]For a Gaussian distributed vector $\boldsymbol{\kappa}$ with zero mean, $E[\boldsymbol{\kappa}^T\boldsymbol{\kappa}\boldsymbol{\kappa}^T\boldsymbol{\kappa}] = \mathrm{tr}\{E[\boldsymbol{\kappa}\boldsymbol{\kappa}^T]\}+2\mathrm{tr}\{E[\boldsymbol{\kappa}\boldsymbol{\kappa}^T]^2\}$ [5].

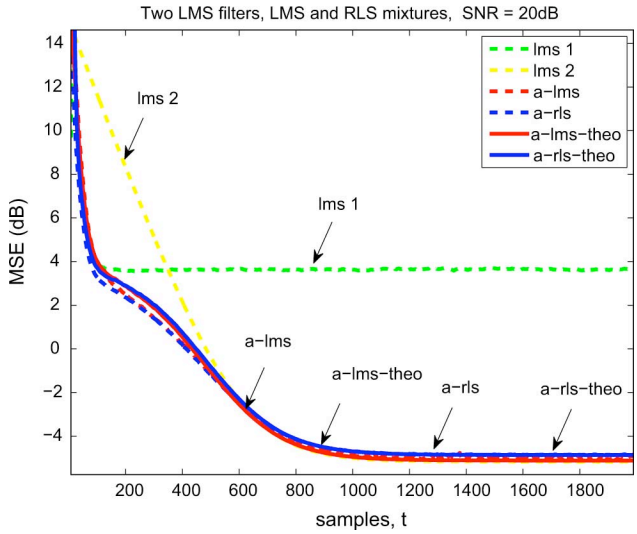

Fig. 2. System identification. The learning rate for the first LMS filter, i.e., "lms 1", is $\mu_1 = 0.1$, for the second LMS filter, i.e., "lms 2", is $\mu_2 = 0.002$, for the adaptive combination using the LMS update "a-lms", is $\mu = 0.012$ and using the RLS update "a-rls", is $1-\lambda = 0.12$. The theoretical curves are also plotted, i.e., "a-lms-theo" and "a-rls-theo".

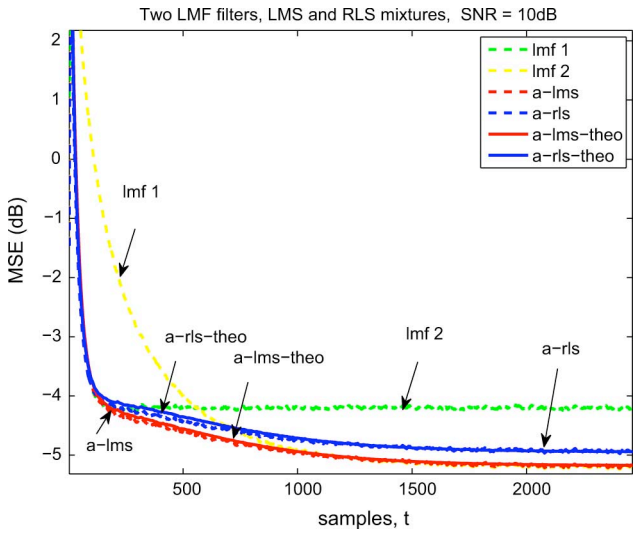

Fig. 3. System identification. The learning rate for the first LMF filter, i.e., "lmf 1", is $\mu_1 = 0.03$, for the second LMF filter, i.e., "lmf 2", is $\mu_2 = 0.0015$, for the adaptive combination using the LMS update "a-lms", is $\mu = 0.1$ and using the RLS update "a-rls", is $1-\lambda = 0.1$. The theoretical curves are also plotted, i.e., "a-lms-theo" and "a-rls-theo".

is replaced by $\|\boldsymbol{\theta}_o(t+1) - \boldsymbol{\theta}_o(t)\|^2_{\boldsymbol{\Sigma}}$. Note that if the constituent filters are not of the equal length, then the terms $E[\|\boldsymbol{\kappa}(t)\|^2]$ and $E[\|\boldsymbol{\kappa}(t)\|^4]$ should be evaluated accordingly. However, the derivations still hold after these replacements. When RLS based updates are used to train a mixture of m constituent filters, we also need to include $\boldsymbol{\Phi}^{-1}(t) \triangleq \{\lambda^t \delta \boldsymbol{I} + \sum_{i=1}^{t}(\lambda^{t-i}\boldsymbol{\kappa}(i)\boldsymbol{\kappa}^T(i))\}$ in the state space evaluation. Again, one can use $\boldsymbol{\Phi}(t+1) \approx \boldsymbol{\Phi}(t)$ approximation to derive the corresponding transient analysis.

Remark 5: For complex-valued data, where $d(t) \triangleq n(t) + \boldsymbol{u}^H(t)\boldsymbol{w}_o$ and $\boldsymbol{w}(t+1) = \boldsymbol{w}(t) + \mu \boldsymbol{u}(t)[d(t) - \boldsymbol{u}^H(t)\boldsymbol{w}(t)]$, defining for a vector $\boldsymbol{v}$, $\|\boldsymbol{v}\|^2 = \boldsymbol{v}^H\boldsymbol{v}$, (9) yields

$$E[\|\tilde{\theta}(t+1)\|^2] = \{1 - 2\mu E[\|e_{a,1}(t) - e_{a,2}(t)\|^2] + \mu^2 E[\|e_{a,1}(t) - e_{a,2}(t)\|^4]\} E[\|\tilde{\theta}(t)\|^2] + \mu^2 E[\|e_{a,1}(t) - e_{a,2}(t)\|^2] E[\|e_o(t)\|^2] - \alpha(t).$$

The derivations follow the case with the real valued data after this replacement.

We next provide numerical examples.

IV. Numerical Examples

The first set of experiments involve estimating a linear model of 7th order with $\boldsymbol{w}_o = [0.25, -0.47, -0.37, 0.045, -0.18, 0.78, 0.147]^T$ as in Fig. 1 [1]. Here, $\sigma_n^2 = 0.3$, the input regressor $\boldsymbol{u}(t)$ is i.i.d. with variance equal to 1 for each entry and the norm of $\boldsymbol{w}_o$ is scaled to yield SNR $= 20$dB. As the constituent filters, we use two 7th order linear filters with the LMS update to train $\boldsymbol{w}_1(t)$ and $\boldsymbol{w}_2(t)$, i.e., $\boldsymbol{w}_i(t+1) = \boldsymbol{w}_i(t) + \mu_i e_i(t)\boldsymbol{u}(t)$, $i = 1, 2$. For the first LMS filter, labeled "lms 1" in Fig. 2, the learning rate is selected as $\mu_1 = 0.2$, for the second filter, labeled "lms 2", as $\mu_2 = 0.004$, for the adaptive affine combination using the LMS update, labeled "a-lms", as $\mu = 0.012$ and for the adaptive affine combination using the RLS update, labeled "a-rls", as $1-\lambda = 0.12$. In Fig. 2, we plot the corresponding MSEs for all algorithms. The learning rates of the adaptive affine combinations are selected such that the mixtures initially follow the rapidly converging filter and then follow the slowly converging filter with the lower steady-state MSE [2]. The results are averaged over 2×10^5 independent trials and smoothed using 10 consecutive samples. We also plot the theoretically derived MSE curves for the adaptive affine combination using the LMS update, labeled "a-lms-theo", using (10) and (12), and the RLS update labeled "a-rls-theo", using (16). For all simulations, as the initial condition, we set $E[\|\tilde{\boldsymbol{\beta}}(0)\|^2] = 1$. The theoretical curves in Fig. 2 are produced using $J_{\text{ex},ii}(t)$, $J_{\text{ex},ij}(t)$ and $\boldsymbol{\theta}_o(t)$ that are calculated from the simulations, since our goal is to illustrate the validity of derived equations. We observe a close agreement between the derivations and simulations for these trials. We observe that our transient analysis gets more accurate as the sample size increases. As highlighted in the correspondence, our derivations are generic with respect to how the constituent filters are trained provided that they produce unbiased estimates. To demonstrate this, in the next set of experiments, we try to learn the same linear model using the least-mean fourth (LMF) [12] algorithm, i.e., $\boldsymbol{w}_i(t+1) = \boldsymbol{w}_i(t) + \mu_i e_i^3(t)\boldsymbol{u}(t)$, $i = 1, 2$. The learning rates are selected as $\mu_1 = 0.03$, $\mu_2 = 0.0015$ for the LMF filters, $\mu = 0.1$ for the LMS mixture, $1-\lambda = 0.1$ for the RLS mixture and SNR $= 10$ dB. We plot in Fig. 3 MSE curves for the constituent filters, labeled "lmf 1" and "lmf 2", for the adaptive affine filter using the LMS update, labeled "a-lms", using the RLS update "a-rls" and theoretically derived curves, labeled "a-lms-theo" and "a-rls-theo", respectively. We also note that the performance of the combination typically lies close to the best performing filter. However, due to the "diversity combining" [4] property the performance of the combination can be better than the individual branches, especially for the regions where branches have comparable performance. To test the validity of the assumption $E[\theta(t)] = \theta_o(t)$, heavily used in the derivations, we plot in Fig. 4 the curves for $E[\theta(t)]$ and $\theta_o(t)$ for the mixtures using the LMS and RLS updates under the setup of Fig. 2. We observe that the assumption is rather accurate in the initial phase of the transient and gets better as the data length increases.

As the last set of experiments, we apply the affinely constrained mixture to combine outputs of three constituent filters. Here, $\sigma_n^2 = 1$, the input regressor $\boldsymbol{u}(t)$ is i.i.d. with variance equal to 1 for each entry and the norm of $\boldsymbol{w}_o$ is scaled to yield SNR $= 0$ dB. In Fig. 5, we plot

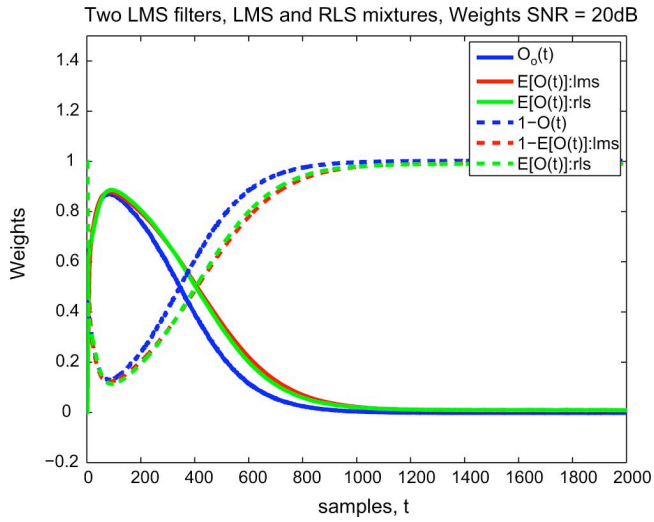

Fig. 4. $E[\theta(t)]$ and $\theta_o(t)$. The curves for $E[\theta(t)]$ (and $1 - E[\theta(t)]$) are plotted for mixtures using the LMS and RLS updates.

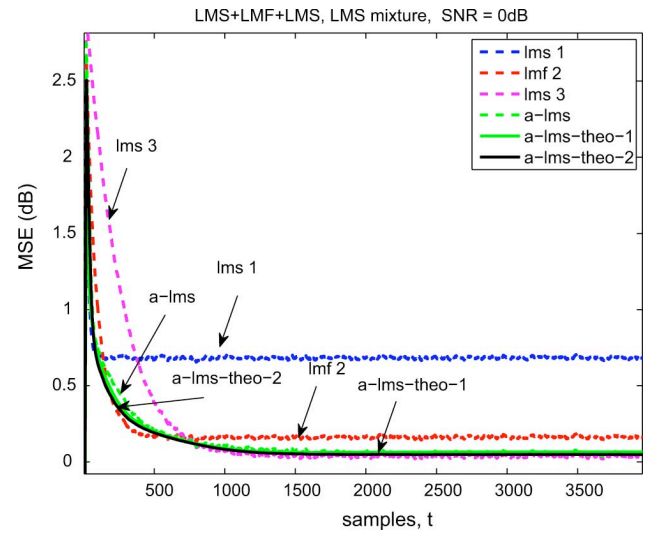

Fig. 5. System identification. Learning rate for the first LMS filter, i.e., "lms 1", is $\mu_1 = 0.04$, for the second LMF filter, i.e., "lmf 2", is $\mu_2 = 0.002$, for the third LMS filter, i.e., "lms 3", is $\mu_3 = 0.0025$, and for the adaptive combination using the LMS update, i.e., "a-lms", is $\mu = 0.06$. The theoretically derived MSE of the mixtures are also plotted.

the MSE curves corresponding to a linear filter using the LMS update with $\mu_1 = 0.04$, labeled "lms 1", a linear filter using the LMF update with $\mu_1 = 0.002$, labeled "lmf 2", a linear filter using the LMS update with $\mu_3 = 0.0025$, labeled "lms 3", affine mixture using the LMS update $\mu = 0.06$, labeled "a-lms', theoretically derived curve based on the state space representation from (10), labeled "a-lms-theo-1" and theoretically derived curve based on(11) "a-lms-theo-2". Note that the learning rates of the constituent filters are selected such that the first filter converges quickly with a relatively high final MSE, the third filter converges slowly with the lowest final MSE and the second filter yields a MSE curve in between. As in the previous examples, we observe that the agreement between the simulations and theory gets better as the data length grows for these trials.

V. Conclusion

In this correspondence, we perform a transient analysis of an affinely constrained mixture method that adaptively combines the outputs of two adaptive linear filters of the same length running in parallel to model a linear system. The analysis is generic with respect to how the constituent filters are trained as long as they perform unbiased estimation of the underlying linear model. We also demonstrate how this analysis can be generalized to mixtures having more than two constituent filters under affine constraints or unconstrained configurations, using Newton based updates to train the mixture weights or working on complex valued data. We observe a close agreement with the theory and our simulations.

References

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

[2] N. J. Bershad, J. C. M. Bermudez, and J. Tourneret, "An affine combination of two LMS adaptive filters: Transient mean-square analysis," *IEEE Trans. Signal Process.*, vol. 56, no. 5, pp. 1853–1864, May 2008.

[3] M. T. M. Silva and V. H. Nascimento, "Improving the tracking capability of adaptive filters via convex combination," *IEEE Trans. Signal Process.*, vol. 56, no. 7, pp. 3137–3149, Jul. 2008.

[4] S. S. Kozat, A. T. Erdogan, A. C. Singer, and A. H. Sayed, "Steady state MSE performance analysis of mixture approaches to adaptive filtering," *IEEE Trans. Signal Process.*, vol. 58, pp. 4050–4063, Aug. 2010.

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

[6] M. Niedzwiecki, "Identification of nonstationary stochastic systems using parallel estimation schemes," *IEEE Trans. Autom. Control*, vol. 35, no. 3, pp. 329–334, Mar. 1990.

[7] L. A. Azpicueta-Ruiz, A. R. Figueiras-Vidal, and J. Arenas-Garcia, "A new least-squares adaptation scheme for the affine combination of two adaptive filters," in *Proc. IEEE Workshop Mach. Learn. Signal Process.*, 2008, pp. 327–332.

[8] R. Candido, M. T. M. Silva, and V. H. Nascimento, "Affine combination of adaptive filters," in *Proc. Asilomar Conf.*, 2008, pp. 236–240.

[9] J. C. M. Bermudez, N. l. Bershad, and J. Y. Tourneret, "An affine combination of two NLMS adaptive filters—Transient mean-square analysis," in *Proc. Asilomar Conf.*, 2008, pp. 230–235.

[10] R. Candido, M. T. M. Silva, and V. H. Nascimento, "Transient and steady-state analysis of the affine combination of two adaptive filters," *IEEE Trans. Signal Process.*, vol. 58, no. 8, pp. 4064–4078, Aug. 2010.

[11] T. Trump, "Transient analysis of an output signal based combination of two adaptive filters," in *Proc. IEEE Int. Workshop Mach. Learn. Signal Process.*, Aug. 2010, pp. 244–249.

[12] J. C. M. Bermudez, P. I. Hubscher, and V. H. Nascimento, "A mean-square stability analysis of the least mean fourth (LMF) adaptive algorithm," *IEEE Trans. Signal Process.*, vol. 55, no. 8, pp. 4018–4028, Aug. 2007.