A LOW-COMPLEXITY STRATEGY FOR SPEEDING UP THE CONVERGENCE OF CONVEX COMBINATIONS OF ADAPTIVE FILTERS

Vítor H. Nascimento
Department of Electronic Systems Eng.,
University of São Paulo, São Paulo, Brazil.
Email: vitor@lps.usp.br

Rodrigo C. de Lamare
Communications Research Group,
University of York, United Kingdom
Email: rcdl500@ohm.york.ac.uk

ABSTRACT

In this work a low-complexity strategy for accelerating the convergence of convex combinations of adaptive filters is proposed. The idea is based on an instantaneous transfer of coefficients from a fast adaptive filter to a slow adaptive filter, which is performed according to a pre-defined window length. A theoretical model that is capable of predicting the excess mean squared error (EMSE) of the proposed strategy is also presented. Simulation results illustrate the good performance of the proposed strategy and the effectiveness of the proposed model to predict the EMSE.

***Index Terms*—** Adaptive filters, convex combinations, cooperative learning.

1. INTRODUCTION

Cooperative learning schemes may greatly improve the performance of an adaptive filter, since they combine the best features of the cooperating algorithms. For example, using cooperative schemes one may robustly avoid the traditional adaptive filtering compromise between low steady-state misadjustment and fast convergence rate [1]. Several interesting applications have been proposed: convex [1, 2] or affine [3, 4] combinations of adaptive filters with different learning rates, combinations of filters from different families to obtain faster tracking [5], iterative determination of the optimum filter length [6, 7]. In addition, distributed cooperative adaptive schemes have been proposed and analysed in [8, 9].

When using cooperative schemes with filters of different learning rates, the fast filter will quickly converge and track variations of the optimum weight vector, whereas the slow filter will reduce the excess mean-square error in steady-state [1, 2]. During the initial convergence, or after abrupt changes of the optimum weight vector, the slow filter may lag considerably behind the fast filter, slowing down the overall convergence of the combined filter. A solution for this problem was proposed in [10, 11], in which the adaptation of the slow filter is modified by the addition of a small multiple of the fast filter. Although this solution works very well, it is somewhat expensive, requiring $2M$ extra multiplications, M extra sums and one comparison per input sample, where M is the filter length. A different kind of combination, in which the filters are combined in series, instead of in parallel, was recently proposed in [12]. The algorithm proposed in [12] achieves fast initial convergence with low cost, but in its current state, the method does not behave well when tracking fast changes of the optimum filter.

In this work, we propose a simple modification to the solution proposed in [10] which requires only one comparison and one addition more than the standard combination. The strategy is based on an instantaneous transfer of coefficients performed according to a pre-defined window length. We also present a theoretical model that is capable of predicting the excess mean squared error (EMSE) of the proposed strategy. Simulation results illustrate the good performance of the proposed strategy and the effectiveness of the proposed model to predict the EMSE.

This paper is structured as follows. Section 2 reviews the fundamentals of cooperative estimation and states the problem. The proposed strategy to instantaneously transfer coefficients is detailed in Section 3. Section 4 is devoted to the proposed theoretical model, whereas Section 5 presents and discusses the simulations. Section 6 gives the conclusions of the paper.

2. COOPERATIVE ESTIMATION

In the cooperative combination schemes considered here, two adaptive filters are run in parallel, with the same inputs, and their outputs are linearly combined, as shown in Fig. 1. As in [1, 5], we use as coefficients $\eta(n)$ and $1-\eta(n)$ with $0 \leq \eta(n) \leq 1$, so that the combination is convex. If the mixing parameter $\eta(n)$ is properly chosen, the overall filter will perform at least as well as the best filter in the combination at each time instant [2].

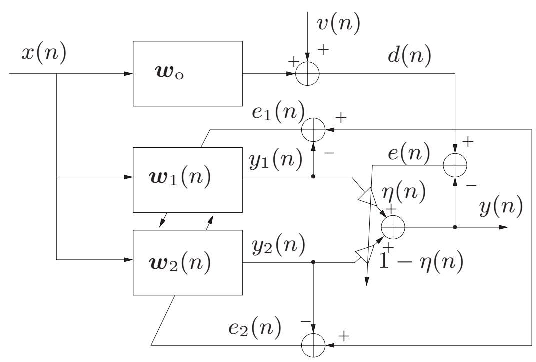

Fig. 1. Convex combination of two adaptive filters.

In Fig. 1, $\{x(n)\}_{n=0}^{\infty}$ and $\{d(n)\}_{n=0}^{\infty}$ are the *regressor* and *desired* sequences, respectively. The output and error for each component filter are, respectively, $y_i(n)$ and $e_i(n)$, $i = 1, 2$. The overall output is

$$y(n) = \eta(n)y_1(n) + (1-\eta(n))y_2(n), \tag{1}$$

and similarly the overall error is $e(n) = \eta(n)e_1(n)+(1-\eta(n))e_2(n)$. In order to smoothly constrain the mixing parameter to the interval $[0, 1]$, we use an auxiliary variable $a(n)$, such that [1]

$$\eta(n) = \frac{1}{1+\mathrm{e}^{-a(n)}}, \tag{2}$$

and $a(n)$ is adapted through [13, 2] (we omit the time index in $\eta(n)$ for concision)

$$a(n+1) = a(n) + \frac{\mu_a}{\epsilon + p(n)} e(n)(y_1(n) - y_2(n))\eta(1-\eta), \tag{3}$$

where $\epsilon > 0$ is a regularization constant and $p(n)$ is a normalization factor, given by the recursion ($0 \ll \nu < 1$ is a forgetting factor)

$$p(n+1) = \nu p(n) + (1-\nu)(y_1(n) - y_2(n))^2. \tag{4}$$

We use two modifications to these recursions, proposed in [1]. In practice, $a(n)$ should be restricted to the interval $[-a_+, a_+]$, otherwise the factor $\eta\,(1-\eta)$ would make adaptation virtually stop when η is close to zero or one. In addition, the gradient noise is reduced if we use $\eta = 1$ when $a(n) = a_+$, and $\eta = 0$ when $a(n) = -a_+$. In general, $a_+ = 4$ gives good results [1, 5].

The component filters may be adapted using any algorithm, such as least-mean squares (LMS), recursive least-squares (RLS), normalized LMS (NLMS). It is not necessary to adapt both filters using the same algorithm (in effect, using different algorithms has advantages for tracking, as shown in [5, 14]). For simplicity, we consider here the combination of two least-mean square (LMS) filters of the same length, and assume that all signals are real.

Define the *regressor vector* as

$$\boldsymbol{x}(n) = [x(n) \quad x(n-1) \quad \ldots \quad x(n-M+1)]^T,$$

where $M > 0$ is the filter length. Using the LMS algorithm, the component filters are updated through the recursion

$$\boldsymbol{w}_i(n+1) = \boldsymbol{w}_i(n) + \mu_i e_i(n)\boldsymbol{x}(n), \quad i = 1, 2, \tag{5}$$

where μ_i are the step-sizes. If the desired and regressor sequences have zero mean and finite variances σ_x^2 and σ_d^2, there exists $\boldsymbol{w}_o(n) \in \mathbb{R}^M$ such that [15]

$$d(n) = \boldsymbol{w}_o^T(n)\boldsymbol{x}(n) + v(n),$$

When running two filters of the same kind, as here, one uses different step-sizes, so that one filter (the *fast* filter, $i = 1$ in (5)) has a relatively large step-size and thus converges quickly and tracks well fast-varying parameters, while the other filter (the *slow* filter, $i = 2$ in (5)) has a relatively small step-size, such that its excess mean-square error (EMSE) will be small during periods in which the optimum weight vector $\boldsymbol{w}_o(n)$ varies slowly.

The combination will thus have both fast tracking and small EMSE. However, as Fig. 2 shows, during the initial adaptation or after an abrupt change of the optimum weight parameters, after the fast filter reaches its steady-state EMSE, the overall filter stops converging, waiting for the slow filter to reach the same performance level. Only after the slow filter catches up with the fast filter will the overall structure resume improving the EMSE. In order to speed-up the overall filter convergence, [10] proposes a so-called *transfer of coefficients* scheme: when $\eta(n)$ is close to 1 (so the fast filter gives better estimates), the fast filter weight vector is used to bias the recursion of the slow filter, as follows. Choose $0 \ll \alpha_0 < 1$, and

$$\begin{aligned} &\text{If } a(n+1) \geq a_+, \text{ then } \alpha(n) = \alpha_0, \text{else } \alpha(n) = 1, \\ &\boldsymbol{w}_2(n+1) = \alpha(n)\left(\boldsymbol{w}_2(n) + \mu_2 e_2(n)\boldsymbol{x}(n)\right) + \\ &\qquad + (1-\alpha(n))\,\boldsymbol{w}_1(n). \end{aligned} \tag{6}$$

This algorithm, which will be called here *linear transfer of coefficients*, works very well, as can be seen in Fig. 2. However, its cost is relatively high: when $\alpha(n+1) = a_+$ and the transfer of coefficients is active, (3) requires $2M$ extra multiplications and M extra additions, with respect to the standard combination. We describe here a method with a better or equivalent performance, which operates at a much smaller cost. The problem we are interested in solving is to find a cost-effective strategy that can accelerate the adaptation of the slow filter during the initial adaptation or after an abrupt change.

3. PROPOSED INSTANTANEOUS TRANSFER STRATEGY

Inspired by the reduced-rank schemes of [16], in which the reduced-rank filter is obtained by switching instantaneously between competing models, we propose here a low-cost alternative to the *linear transfer of coefficients*. The algorithm is as follows. Define a window length N_0. Then, if $n \mod N_0 = 0$ and $a(n+1) = a_+$, instead of using (5) to compute $\boldsymbol{w}_2(n+1)$, do

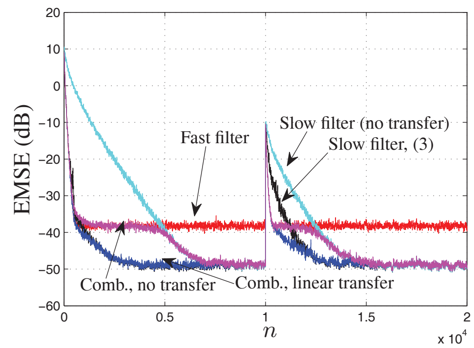

Fig. 2. Ensemble-average learning curves of convex combination of two filters, average of 50 realizations. $\{x(n)\}$ is a stationary first-order AR process with unit power, white Gaussian input and pole at 0.5. Noise with variance $\sigma_v^2 = 10^{-3}$. Filter parameters: $\mu_1 = 0.025$, $\mu_2 = 0.0025$, $\mu_a = 1$, $\epsilon = 10^{-4}$, $\nu = 0.9$, $\alpha_0 = 0.6$, $M = 10$. $\boldsymbol{w}_o$ is kept constant, except for a random perturbation at iteration 10^4. Learning curves smoothed using a 10-coefficient MA filter with unit gain and the filtfilt Matlab function to avoid distortion.

$$\boldsymbol{w}_2(n+1) = \boldsymbol{w}_1(n+1). \tag{7}$$

The additional cost with relation to the standard combination (without transfer) is just one addition and one comparison — in fact, the standard combination already requires checking if $a(n+1) = a_+$, so the only additional operation is the $n \mod N_0$ operation. This can implemented by using an auxiliary variable m initialized with N_0 and decremented at each time instant until it becomes zero, at which point we reset $m = N_0$. When (7) is applied, the cost of this algorithm is actually smaller than that of the original combination, since only one filter is being adapted.

We tested several values of N_0, in different situations. The method is not very sensitive to this choice, but it turns out that the best option is $N_0 = 2$. Small values of N_0 do not affect the performance in a noticeable way, but large values of N_0 make the slow filter lag behind the fast filter (see Sec. 5). The pseudo-code of the proposed algorithm is shown below.

Instantaneous transfer
Initialize both component filters. Let $m \leftarrow N_0$, $a(0) \leftarrow a_+$, $n \leftarrow 0$.
At each step, repeat:

1. Compute $y_i(n)$ and $e_i(n)$ for the component filters, and the overall output using (1).
2. Update fast filter according to (5). Let $m \leftarrow m - 1$.
3. Compute $a(n+1)$ and $\eta(n+1)$ using (2)-(3). If $a(n+1) < -a_+$, let $a(n+1) \leftarrow -a_+$, $\eta(n+1) \leftarrow 0$.
4. If $a(n+1) \geq a_+$, let $a(n+1) \leftarrow a_+$, $\eta(n+1) \leftarrow 1$. If also $m = 0$, let $\boldsymbol{w}_2(n+1) \leftarrow \boldsymbol{w}_1(n+1)$ and $m \leftarrow N_0$.
5. If the conditions in step 4 are not met, update the slow filter according to (5).
6. Let $n \leftarrow n + 1$ and return to step 2.

4. THEORETICAL MODEL

The model derived in [11] can be easily extended to this new algorithm. The model in [11] uses Taylor expansions of η in (2) and (3) to obtain a recursion for the mean of $a(n)$,

$$\bar{a}(n) = \mathrm{E}\{a(n)\}, \quad \text{and} \quad \mathrm{E}\{\eta(n)\} \approx \bar{\eta}(n) = \frac{1}{1+e^{-\bar{a}(n)}}.$$

In order to describe the model, we need the following definitions: for each filter its weight error vector is $\tilde{\boldsymbol{w}}_i(n) = \boldsymbol{w}_\mathrm{o} - \boldsymbol{w}_i(n)$, and the *a-priori* errors are $e_{a,i} = \tilde{\boldsymbol{w}}_i^T(n)\boldsymbol{x}(n)$, $i = 1, 2$.

Models for combinations of adaptive filters need approximations for the *excess mean-square errors* (EMSEs) of each filter, as well as for the *cross*-EMSE [1, 5, 4]

$$\zeta_i(n) = \mathrm{E}\{e_{a,i}^2(n)\}, \qquad \zeta_{12}(n) = \mathrm{E}\{e_{a,1}(n)e_{a,2}(n)\}, \tag{8}$$

where $\mathrm{E}\{\cdot\}$ denotes expected value.

Following [11], we assume that the regressor sequence $\{\boldsymbol{x}(n)\}$ is independent, identically distributed (iid) and independent of the noise sequence $\{v(n)\}$, which is also assumed iid. In this case, the regressor autocorrelation matrix $\boldsymbol{R} = \mathrm{E}\{\boldsymbol{x}(n)\boldsymbol{x}^T(n)\}$ is constant. Its eigenvalue expansion is $\boldsymbol{R} = \boldsymbol{U}^T\boldsymbol{\Lambda}\boldsymbol{U}$, where $\boldsymbol{U}$ is an orthogonal matrix, i.e., $\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{U}\boldsymbol{U}^T = \boldsymbol{I}$. The diagonal matrix $\boldsymbol{\Lambda}$ contains the eigenvalues λ_k, $k = 1 \ldots M$ of $\boldsymbol{R}$, that is, $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_k)$.

In order to take into account the effect of a non-stationary $\boldsymbol{w}_\mathrm{o}(n)$, we assume a simple random-walk model for its evolution [15]

$$\boldsymbol{w}_\mathrm{o}(n+1) = \boldsymbol{w}_\mathrm{o}(n) + \boldsymbol{q}(n), \tag{9}$$

where $\{\boldsymbol{q}(n)\}$ is an iid random vector sequence, independent of $\{v(n)\}$ and of $\{\boldsymbol{x}(n)\}$, with autocorrelation matrix

$$\mathrm{E}\{\boldsymbol{q}(n)\boldsymbol{q}^T(n)\} = \sigma_q^2\boldsymbol{I}.$$

The models for the EMSEs and cross-EMSE can be obtained through recursions for the diagonals of the rotated autocorrelation matrices [11]

$$\begin{aligned} \boldsymbol{s}_{ii}(n) &= \mathrm{diag}\left(\mathrm{E}\{\boldsymbol{U}\tilde{\boldsymbol{w}}_i(n)\tilde{\boldsymbol{w}}_i^T(n)\boldsymbol{U}^T\}\right), i = 1, 2 \\ \boldsymbol{s}_{12}(n) &= \mathrm{diag}\left(\mathrm{E}\{\boldsymbol{U}\tilde{\boldsymbol{w}}_1(n)\tilde{\boldsymbol{w}}_2^T(n)\boldsymbol{U}^T\}\right). \end{aligned} \tag{10}$$

Defining $\boldsymbol{\ell} = [\lambda_1 \quad \ldots \quad \lambda_M]^T$, it can be shown that $\zeta_{ij}(n) = \boldsymbol{\ell}^T\boldsymbol{s}_{ij}(n)$, $i, j = 1, 2$ [11]. A recursion for $\boldsymbol{s}_{11}(n)$, assuming that $x(n)$ is Gaussian, is given by

$$\boldsymbol{s}_{11}(n+1) = \boldsymbol{A}_1\boldsymbol{s}_{11}(n) + \mu_1^2\sigma_v^2\boldsymbol{\ell}, \tag{11}$$

where $\boldsymbol{A}_1 = \boldsymbol{I} - 2\mu_1\boldsymbol{\Lambda} + \mu_1^2\left(2\boldsymbol{\Lambda}^2 + \boldsymbol{\ell}\boldsymbol{\ell}^T\right)$. This is a well-known result in adaptive filtering [17]. The recursions for $\boldsymbol{s}_{22}(n)$ and $\boldsymbol{s}_{12}(n)$ must be modified with respect to what is described in [11] to account for the transfer of coefficients (7). Assuming that $\bar{a}(n+1)$ is available, we have

If $\bar{a}(n+1) < a_+$, then

$$\begin{aligned} \boldsymbol{s}_{12}(n+1) &= \boldsymbol{A}_{12}\boldsymbol{s}_{12}(n) + \mu_1\mu_2\sigma_v^2\boldsymbol{\ell} + \sigma_q^2\boldsymbol{1}, \\ \boldsymbol{s}_{22}(n+1) &= \boldsymbol{A}_2\boldsymbol{s}_{22}(n) + \mu_2^2\sigma_v^2\boldsymbol{\ell} + \sigma_q^2\boldsymbol{1}, \end{aligned} \tag{12}$$

where $\boldsymbol{1} = [1 \quad 1 \quad \ldots \quad 1]^T \in \mathbb{R}^M$, and

$$\boldsymbol{A}_1 = \boldsymbol{I} - (\mu_1 + \mu_2)\boldsymbol{\Lambda} + \mu_1\mu_2\left(2\boldsymbol{\Lambda}^2 + \boldsymbol{\ell}\boldsymbol{\ell}^T\right), \tag{13}$$

$$\boldsymbol{A}_2 = \boldsymbol{I} - 2\mu_2\boldsymbol{\Lambda} + \mu_2^2\left(2\boldsymbol{\Lambda}^2 + \boldsymbol{\ell}\boldsymbol{\ell}^T\right). \tag{14}$$

Otherwise, when $\bar{a}(n+1) \geq a_+$, then

$$\boldsymbol{s}_{12}(n+1) = \boldsymbol{s}_{11}(n+1), \quad \boldsymbol{s}_{22}(n+1) = \boldsymbol{s}_{11}(n+1). \tag{15}$$

The recursions (12)-(15) constitutes the main novelty with respect to the model in [11] — having $\boldsymbol{s}_{12}(n)$ and $\boldsymbol{s}_{22}(n)$, we can use the recursions described in that reference to obtain $\bar{a}(n)$ and $\bar{\eta}(n)$. The overall EMSE is given by

$$\zeta(n) = \mathrm{E}\left\{(\eta(n)e_{a,1}(n) + (1-\eta(n))e_{a,2}(n))^2\right\}.$$

At this point it should be noted that the analytical expression above can be simplified by discarding the contributions of $\sigma_a^2(n)$ and $\sigma_\eta^2(n)$. Therefore, the EMSE can be approximated by the expression

$$\begin{aligned} \zeta(n) \approx\ & \bar{\eta}^2\big[\zeta_{11}(n) - 2\zeta_{12}(n) + \zeta_{22}(n)\big] \\ & + 2\bar{\eta}\big[\zeta_{12}(n) - \zeta_{22}(n)\big] + \zeta_{22}(n). \end{aligned} \tag{16}$$

The recursion for $\bar{a}(n)$ is given in terms of the following functions

$$f_1(a) = -\eta^2(1-\eta), \qquad f_2(a) = \eta(1-\eta)^2, \tag{17a}$$

$$f_3(a) = \eta(2\eta-1)(1-\eta), \qquad f_4(a) = \eta(1-\eta). \tag{17b}$$

Defining $\bar{f}_i = f_i(\bar{a}(n))$, we obtain (note that in (4), $y_1(n) - y_2(n) = e_{a,2}(n) - e_{a,1}(n)$):

$$\bar{a}(n+1) \approx \bar{a}(n) + \mu_a\frac{\left[\bar{f}_1\zeta_{11}(n) + \bar{f}_2\zeta_{22}(n) + \bar{f}_3\zeta_{12}(n)\right]}{\epsilon + \bar{p}(n)}, \tag{18}$$

$$\bar{p}(n+1) = \nu\bar{p}(n) + (1-\nu)\left(\zeta_1(n) - 2\zeta_{12}(n) + \zeta_2(n)\right) \tag{19}$$

Following what is done in the algorithm itself, we restrict $\bar{a}(n+1)$ to the interval $[-a_+, a_+]$.

5. SIMULATIONS

In this section, the performance of the proposed instantaneous transfer strategy and the existing schemes for combinations of adaptive filters are evaluated. In particular, the EMSE expression obtained in (16) is employed to obtain curves that are compared with those computed via simulations.

Fig. 3 shows the performance of the instantaneous transfer (7) in the same stationary ($\sigma_q^2 = 0$) setting as Fig. 2. The EMSE resulting from the instantaneous transfer is shown in green, indistinguishable from that of the linear transfer (3) behind it. The theoretical model of the previous section is also plotted, showing good agreement with the simulations.

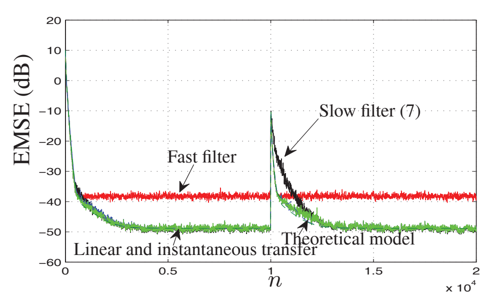

Fig. 3. Ensemble-average learning curves, instantaneous transfer. Conditions are those of Fig. 2, with $N_0 = 2$. Broken line: theoretical model.

The new algorithm works well also if the optimum filter $\boldsymbol{w}_\mathrm{o}$ varies according to (9). Fig. 4 shows the performance of the new algorithm when $M = 10$ and $\boldsymbol{Q} = 10^{-8}\boldsymbol{I}$, and Fig. 6, when $M = 50$ and $\boldsymbol{Q} = 10^{-10}\boldsymbol{I}$. Fig. 5 shows the models for $\bar{a}$, $\bar{\eta}$. As can be seen, the theoretical models approximate well the means.

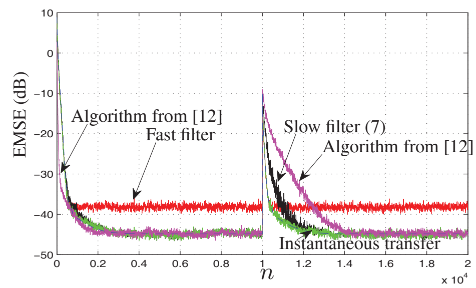

Fig. 4. Ensemble-average learning curves. Conditions same as in Fig. 3, except for $\sigma_q^2 = 10^{-8}$. Broken line: theoretical model.

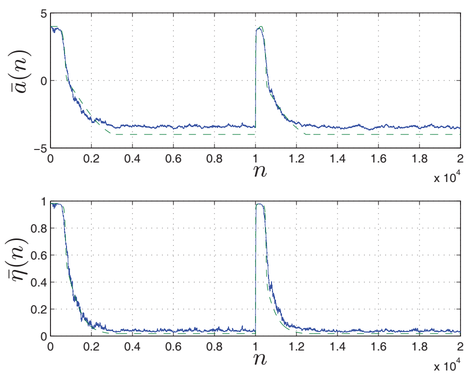

Fig. 5. Ensemble-average and theoretical approximations for $\bar{a}(n)$, $\bar{\eta}(n)$. Conditions as in Fig. 4. Broken lines: theoretical model.

6. CONCLUSION

In this paper we have proposed a new low-cost method for accelerating the convergence of convex combinations of adaptive filters. We have also devised a theoretical model to predict the EMSE of the new method, which agrees well with simulations. Although we exemplified the new method using combinations of LMS filters, our proposal can be used equally with other choices of filters.

7. REFERENCES

[1] J. Arenas-García, 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] V. H. Nascimento, M. T. Silva, R. Candido, and J. Arenas-Garcia, "A transient analysis for the convex combination of adaptive filters," in *Proc. IEEE Workshop on Statistical Signal Process.*, Cardiff, UK, 2009, pp. 53–56.

[3] N. J. Bershad, J. C. M. Bermudez, and J.-Y. 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.

[4] 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, 2010.

[5] 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.

[6] Y. Zhang and J. Chambers, "Convex combination of adaptive filters for a variable tap-length LMS algorithm," *IEEE Signal Process. Lett.*, vol. 13, no. 10, pp. 628–631, Oct. 2006.

[7] M. Zeller, L. A. Azpicueta-Ruiz, and W. Kellermann, "Online estimation of the optimum quadratic kernel size of second-order Volterra filters using a convex combination scheme," in *Proc., ICASSP2009*, 2009, pp. 2965–2968.

[8] C. G. Lopes and A. H. Sayed, "Diffusion least-mean squares over adaptive networks: Formulation and performance analysis," *IEEE Trans. Signal Process.*, vol. 56, no. 7, pp. 3122–3136, 2008.

[9] F. S. Cattivelli, C. G. Lopes, and A. H. Sayed, "Diffusion recursive least-squares for distributed estimation over adaptive networks," *IEEE Trans. Signal Process.*, vol. 56, no. 5, pp. 1865–1877, 2008.

[10] J. Arenas-García, M. Martínez-Ramón, A. Navia-Vázquez, and A. R. Figueiras-Vidal, "Plant identification via adaptive combination of transversal filters," *Signal Processing*, vol. 86, pp. 2430–2438, 2006.

[11] M. T. M. Silva, V. H. Nascimento, and J. Arenas-García, "A transient analysis for the convex combination of two adaptive filters with transfer of coefficients," in *Proc., ICASSP 2010*, Dallas, USA, pp. 3842–3845.

[12] W. B. Lopes and C. G. Lopes, "Incremental-cooperative strategies in combination of adaptive filters," in *Proc. IEEE Int Acoustics, Speech and Signal Processing (ICASSP) Conf*, 2011, pp. 4132–4135.

[13] L. Azpicueta-Ruiz, A. Figueiras-Vidal, and J. Arenas-Garcia, "A normalized adaptation scheme for the convex combination of two adaptive filters," in *Proc., ICASSP 2008*, Las Vegas, NV, USA, pp. 3301–3304.

[14] V. H. Nascimento, M. T. M. Silva, L. A. Azpicueta-Ruiz, and J. Arenas-García, "On the tracking performance of combinations of least mean squares and recursive least squares adaptive filters," in *Proc., ICASSP 2010*, Dallas, USA, pp. 3710–3713.

[15] A. H. Sayed, *Fundamentals of Adaptive Filtering*. Wiley-Interscience, 2003.

[16] R. C. de Lamare and R. Sampaio-Neto, "Adaptive reduced-rank processing based on joint and iterative interpolation, decimation, and filtering," *IEEE Trans. Signal Process.*, vol. 57, no. 7, pp. 2503–2514, Jul. 2009.

[17] P. S. R. Diniz, *Adaptive Filtering: Algorithms and Practical Implementation*, 3rd ed. Springer, 2008.

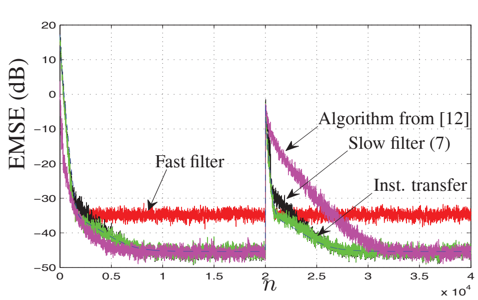

Fig. 6. Ensemble-average learning curves. Conditions as in Fig. 4, except for $M = 50$ and $\sigma_q^2 = 10^{-10}$. $N_0 = 2$. Broken line: theoretical model.