On Parallel-Incremental Combinations of LMS Filters that Outperform the Affine Projection Algorithm

Luiz F. O. Chamon and Cássio G. Lopes

***Abstract*— Data reuse techniques have been successfully applied to improve the performance of adaptive filters, the Affine Projection Algorithm (APA) being one of the most celebrated examples of such achievements. Recently, incremental combinations of LMS filters have been shown to either match or outperform the APA with lower complexity. However, this combination brings forth a trade-off between convergence speed and steady-state error which depends on the number of component filters. This work proposes a parallel-incremental topology with coefficients feedback to allow transient and steady-state performance to be designed separately. Moreover, to address highly correlated environments, a data reusing incremental combination is employed. An approximate analysis is pursued using the data sharing version of this structure. Numerical experiments show that the novel combination can outperform the convergence rate and misadjustment of the APA and combinations of APAs in different scenarios.**

***Keywords*— Adaptive filters, Combination of adaptive filters, Parallel-incremental combination, Affine Projection Algorithm**

I. Introduction

Data reuse—or *data reusing* (DR)—techniques have been successfully employed to improve the performance of adaptive filters (AFs) while maintaining their low computational complexity [1]–[5]. Although their implementations usually incur in additional memory requirements, the resulting algorithms are able to trade-off complexity and performance, a desired characteristic in several applications [5]. One of the most celebrated DR AFs is the Affine Projection Algorithm (APA) [2], especially after fast versions of its update were devised [6], [7]. It remains ubiquitous in applications such as speech echo cancellation, where its performance matches that of Fast Recursive Least Square (FRLS) filters using three times less operations [8].

Combination of AFs is another technique introduced to improve the performance of adaptive algorithms [9]–[16]. It consists of aggregating a pool of AFs—called *components*—by means of a supervisor, adaptive or not, that attempts to achieve universality, i.e., make the overall system at least as good—usually in the mean square sense—as the best filter in the set. Generally, the components run independently while their coefficients are linearly combined by the supervisor. This topology—called *parallel*—has been explored using different AFs, step sizes, orders, and supervising rules [9]–[14].

Signal Processing Laboratory, Dept. of Electronic Systems Engineering, University of São Paulo – Brazil. email: chamon@usp.br, cassio@lps.usp.br. Mr. Chamon's work is supported by the *Coordenação de Aperfeiçoamento de Pessoal de Nível Superior* (CAPES, Brazil).

Parallel combinations, however, present a well-known convergence stagnation that has been addressed in several ways [9], [14], in particular by the *incremental* topology [15]. This last structure has since been extended to take advantage of DR and match—and sometimes outperform—the APA with lower complexity. This improvement in transient performance, however, comes at the cost of a misadjustment increase [16].

This work addresses this last issue by

- introducing a DR parallel-incremental combination of LMS filters;
- using the data sharing incremental combination as an approximation for the behavior of the DR version and analyzing its mean performance;
- illustrating, through simulations, the performance of the new topology and showing that it outperforms the APA—and even APA combinations [17]—with lower complexity.

II. Adaptive filtering

In the widely adopted system identification scenario, consider the measurements

$$d(i) = \boldsymbol{u}_i \boldsymbol{w}^o + v(i), \tag{1}$$

where $\boldsymbol{w}^o$ is an $M \times 1$ vector that models the unknown system, $\boldsymbol{u}_i$ is the $1 \times M$ regressor vector that captures samples $u(i)$ of a zero mean input signal with variance σ_u^2, and $v(i)$ is the realization of a zero mean i.i.d. Gaussian random variable with variance σ_v^2. At iteration i, an AF uses the data pair $\{\boldsymbol{u}_i, d(i)\}$ to update a previous estimate $\boldsymbol{w}_{i-1}$ of $\boldsymbol{w}^o$ through

$$\boldsymbol{w}_i = \boldsymbol{w}_{i-1} + \mu \boldsymbol{p}, \tag{2}$$

in which μ is a step size and $\boldsymbol{p} = -\boldsymbol{B}\,\nabla^* J(\boldsymbol{w}_{i-1})$ is an update direction vector, with $\boldsymbol{B}$ a positive-definite matrix, $J(\boldsymbol{w}_{i-1})$ the underlying cost function of the AF, and * denoting the conjugate transpose operation [18].

Different choices of $\boldsymbol{p}$ lead to different adaptive algorithms, such as the celebrated LMS filter

$$\boldsymbol{w}_i = \boldsymbol{w}_{i-1} + \mu \boldsymbol{u}_i^* e(i), \tag{3}$$

where $e(i) = d(i) - \boldsymbol{u}_i \boldsymbol{w}_{i-1}$ is the output estimation error [18].

A. Data reuse

In adaptive filtering, DR refers either to using $K > 1$ times a single data pair $\{\boldsymbol{u}_i, d(i)\}$ or operating over a set of $K > 1$ data pairs $\{\boldsymbol{U}_i, \boldsymbol{d}_i\}$, where $\boldsymbol{U}_i =$

$\begin{bmatrix} \boldsymbol{u}_i^T & \cdots & \boldsymbol{u}_{i-K+1}^T \end{bmatrix}^T$ is a $K \times M$ regressors matrix and $\boldsymbol{d}_i = \begin{bmatrix} d(i) & \cdots & d(i-K+1) \end{bmatrix}^T$ is a $K \times 1$ measurements vector. The first DR algorithm, the Data Reuse LMS (DR-LMS), was of the former type [1]. Explicitly,

$$\begin{aligned} \boldsymbol{w}_{0,i} &= \boldsymbol{w}_{i-1} \\ \boldsymbol{w}_{k,i} &= \boldsymbol{w}_{k-1,i} + \mu \boldsymbol{u}_i^* [d(i) - \boldsymbol{u}_i \boldsymbol{w}_{k-1,i}] \\ \boldsymbol{w}_i &= \boldsymbol{w}_{K,i}. \end{aligned} \tag{4}$$

The APA, on the other hand, uses the last K data pairs and is given, in its standard regularized form, by

$$\boldsymbol{w}_i = \boldsymbol{w}_{i-1} + \mu \boldsymbol{U}_i^* (\boldsymbol{U}_i \boldsymbol{U}_i^* + \epsilon \boldsymbol{I})^{-1} \boldsymbol{e}_i, \tag{5}$$

where ϵ is a regularization factor and $\boldsymbol{e}_i = \boldsymbol{d}_i - \boldsymbol{U}_i \boldsymbol{w}_{i-1}$ [2], [18]. Specific versions of this algorithm are sometimes presented under different names, such as the Binormalized Data Reusing LMS (BNDR-LMS) for $K = 2$ [4]. For $K = 1$, (5) becomes the Normalized LMS (NLMS) filter [18].

III. Combination of adaptive filters

For the sake of illustration, this section describes combinations of LMS filters, although it is straightforwardly extended to use different adaptive algorithms.

In a combination of N AFs, the components are distinguished by introducing the index $n = 1, \ldots, N$. Hence, the LMS recursion in (3) becomes

$$\boldsymbol{w}_{n,i} = \boldsymbol{w}_{n,i-1} + \mu_n \boldsymbol{u}_{n,i}^* e_n(i), \tag{6}$$

where $e_n(i) = d_n(i) - \boldsymbol{u}_{n,i} \boldsymbol{w}_{n,i-1}$. The components coefficients $\boldsymbol{w}_{n,i}$ are aggregated by the supervisor's parameters $\eta_n(i)$ to obtain the overall—*global*—coefficients $\boldsymbol{w}_i$ that minimize some function of the global estimation error $e(i) = d(i) - \boldsymbol{u}_i \boldsymbol{w}_{i-1}$, usually the mean-square error (MSE) $\mathrm{E}\,|e(i)|^2$.

Different definitions for $\{\boldsymbol{u}_{n,i}, d_n(i)\}$ account for different DR techniques. Generally, combinations of AFs are of the *data sharing* kind, i.e., all components operate over the *same* data pair—$\{\boldsymbol{u}_{n,i}, d_n(i)\} = \{\boldsymbol{u}_i, d(i)\}$ [9]–[15]. Recently, a DR formulation of combinations of AFs was introduced by [16]. Data and component filters were separately indexed using k and n respectively, so that the number of components was not limited by the number of available data pairs, as is the case of the DR-LMS in (4). This *data buffering* approach used $\{\boldsymbol{u}_{n,i}, d_n(i)\} = \{\boldsymbol{u}_{i-k}, d(i-k)\}$, where $k = (n-1) \bmod K$. Hence, the components would use data pairs taken from the k-th row of $\{\boldsymbol{U}_i, \boldsymbol{d}_i\}$, going over the data set as many times as required. A consequence of this formulation is that several DR AFs can be cast and analyzed under a combination framework [16].

The combination topology is induced by the way coefficients are transferred in the structure—i.e., depending on the choice of $\boldsymbol{w}_{n,i-1}$ in (6). The structures mentioned in Section I can therefore be explicitly defined as:

Definition 1 (*Parallel combination* [9], [11], [14])**.**

$$\begin{aligned} \boldsymbol{w}_{n,i-1} &= \delta(i - rL) \boldsymbol{w}_{i-1} + [1 - \delta(i - rL)] \boldsymbol{w}_{n,i-1} \\ \boldsymbol{w}_{n,i} &= w_{n,i-1} + \mu_n \boldsymbol{u}_{n,i}^* [d_n(i) - \boldsymbol{u}_{n,i} \boldsymbol{w}_{n,i-1}] \\ \boldsymbol{w}_i &= \sum_{n=1}^{N} \eta_n(i) \boldsymbol{w}_{n,i} \end{aligned} \tag{7}$$

where $\delta(\cdot)$ is Kronecker's delta and $r \in \mathbb{N}$. The first equation accounts for the cyclic feedback strategy, an effective alternative to transfers of coefficients [9] that addresses the stagnation issue of parallel combinations by feeding back the global coefficients to all components every L iterations, where L is a constant called the *cycle length* [14]. For $N = 2$, data sharing, and $L \to \infty$, the usual parallel-independent structure is recovered [9], [11].

Definition 2 (*Incremental combination* [15], [16])**.**

$$\begin{aligned} \boldsymbol{w}_{0,i} &= \boldsymbol{w}_{i-1} \\ \boldsymbol{w}_{n,i} &= \boldsymbol{w}_{n-1,i} + \eta_n(i) \mu_n \boldsymbol{u}_{n,i}^* [d_n(i) - \boldsymbol{u}_{n,i} \boldsymbol{w}_{n-1,i}] \\ \boldsymbol{w}_i &= \boldsymbol{w}_{N,i}, \end{aligned} \tag{8}$$

Prior to introducing the new topology, it is convenient to introduce a *notation* that uniquely defines combination topologies and avoids the use of long cryptic acronyms. Inspired by transfer function composition in system theory [19], parallel combinations are represented by addition, whereas incremental combinations are represented by multiplication. Intuitive extensions of these can be used as shorthands for larger combinations. Hence, for two LMS filters, their parallel combination can be written as LMS + LMS or $2\{\text{LMS}\}$ and their incremental combination, as LMS · LMS or $\{\text{LMS}\}^2$.

IV. Parallel-incremental combination

Fig. 1a shows the trade-off between transient and steady-state performance in the incremental combination depending on the number of filters N [16]. Still, the combination is able to match the APA. Moreover, although DR yields negligible improvements for i.i.d. inputs—the curves were omitted in Fig. 1a for clarity—, it greatly impacts the convergence rate when data are correlated (Fig. 1b).

These observations motivate a parallel combination between the fast DR-$\{\text{LMS}\}^N$ and a more accurate algorithm to address steady-state performance. Although other AFs would be suitable, a single LMS filter with small step size is chosen due to its low complexity and robustness [18]. In order to avoid the stagnation effect, a cyclic coefficients feedback strategy is employed [14]. Thus, this parallel-incremental combination—LMS + DR-$\{\text{LMS}\}^N$—is described as (see Fig. 2)

i. *Cyclic coefficients feedback*

$$\boldsymbol{w}_{n,i-1} = \delta(i - rL) \boldsymbol{w}_{i-1} + [1 - \delta(i - rL)] \boldsymbol{w}_{n,i-1} \tag{9}$$

ii. *LMS branch*

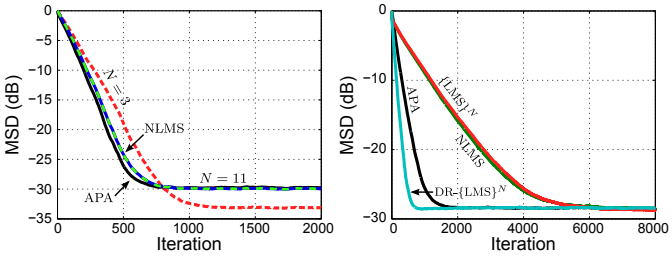
$$\boldsymbol{w}_{1,i} = \boldsymbol{w}_{1,i-1} + \mu_1 \boldsymbol{u}_i^* [d(i) - \boldsymbol{u}_i \boldsymbol{w}_{1,i-1}] \tag{10}$$

iii. *Incremental LMS with DR branch* (DR-$\{\text{LMS}\}^N$)

$$\begin{aligned} \boldsymbol{w}'_{0,i} &= \boldsymbol{w}_{2,i-1} \\ \boldsymbol{w}'_{n,i} &= \boldsymbol{w}'_{n-1,i} + \eta_n(i) \mu_n \boldsymbol{u}_{i-k}^* [d(i-k) - \boldsymbol{u}_{i-k} \boldsymbol{w}'_{n-1,i}] \\ \boldsymbol{w}_{2,i} &= \boldsymbol{w}'_{N,i} \end{aligned} \tag{11}$$

iv. *Parallel combination*

$$\boldsymbol{w}_i = \eta(i) \boldsymbol{w}_{1,i} + [1 - \eta(i)] \boldsymbol{w}_{2,i} \tag{12}$$



(a) White input (b) Correlated input

Fig. 1. **APA, NLMS, $\{$LMS$\}^N$, and DR-$\{$LMS$\}^N$** ($M = 100$, $K = 10$, $\eta_n = 1$, $\mu_n = \mu$). (a) **APA**: $\mu = 0.3$; **NLMS**: $\mu = 1$; and **$\{$LMS$\}^N$**: $\mu = 3 \cdot 10^{-3}$, $N = 20$; (b) **APA**: $\mu = 0.015$; **NLMS**: $\mu = 0.9$; **$\{$LMS$\}^N$**: $\mu = 1.5 \cdot 10^{-3}$, $N = 30$; and **DR-$\{$LMS$\}^N$**: $\mu = 2 \cdot 10^{-3}$, $N = 15$.

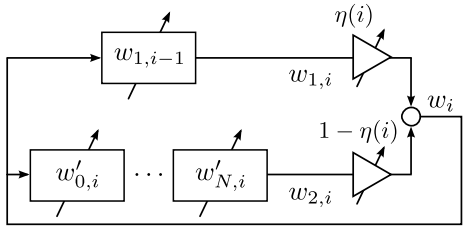


Fig. 2. The novel parallel-incremental combination: LMS + DR-$\{$LMS$\}^N$

where again $n = 1, \ldots, N$, $k = (n-1) \bmod K$, and $r \in \mathbb{N}$. Different methods can be used to adapt $\eta(i)$ [9], [11], [20]. For illustration purposes, a normalized convex supervisor is employed, namely

$$\begin{aligned} p(i) &= \beta p(i-1) + (1-\beta)[y_1(i) - y_2(i)]^2 \\ a(i) &= a(i-1) \\ &\quad + \frac{\mu_a}{p(i)+\epsilon} e(i)[y_1(i) - y_2(i)]\eta(i)[1-\eta(i)] \\ \eta(i) &= \frac{1}{1+e^{-a(i)}}, \end{aligned} \tag{13}$$

where $y_n(i) = \boldsymbol{u}_i \boldsymbol{w}_{n,i-1}$ is the n-th component output and ϵ is a regularization factor [20].

The novel parallel-incremental LMS requires $(N+1)(2M+1) + \mathcal{O}[\eta]$ multiplications per iteration, where $\mathcal{O}[\eta]$ represents the supervisor complexity. For (13), $\mathcal{O}[\eta] = \mathcal{O}[1]$—assuming the sigmoidal function is implemented using a lookup table [21]. Therefore, the complexity of this combination is only slightly higher than that from [16], remaining lower than that of the APA while definitely outperforming it (see Section VI).

V. Brief on Analysis

The LMS branch (10) of the combination is designed to have a low misadjustment, so that the supervisor (13) will track its output once the AF has converged [9], [14], [20]. However, this filter will adapt slowly and the transient phase of the parallel-incremental combination will be dictated by the incremental branch (11).

More detailed developments would require the transient mean-square analysis of the DR-$\{$LMS$\}^N$—or its data sharing version $\{$LMS$\}^N$ as an approximation for white inputs (Fig. 1a). This analysis, however, involves the combined effect of several AFs and is therefore challenging. Nevertheless, the following derivations help understanding the effect of the incremental combination on the convergence rate of AFs and suggest that its use in (11) is indeed appropriate.

A. Mean convergence of the $\{LMS\}^N$

First the overall recursion of the $\{$LMS$\}^N$—data sharing—is derived. Combining the equations in (8) yields

$$\boldsymbol{w}_i = \boldsymbol{w}_{i-1} + \boldsymbol{u}_i^* \sum_{n=1}^{N} \eta_n(i)\mu_n[d(i) - \boldsymbol{u}_i \boldsymbol{w}_{n-1,i}]. \tag{14}$$

Noticing that for $n = 1$, $\boldsymbol{w}_{n-1,i} = \boldsymbol{w}_{i-1}$, and using the coefficients recursion, the estimation errors in (14) expand as

$$[d(i) - \boldsymbol{u}_i \boldsymbol{w}_{n-1,i}] = \prod_{k=1}^{n-1} (1 - \eta_k(i)\mu_k \|\boldsymbol{u}_i\|^2) e(i). \tag{15}$$

Substituting (15) and defining the global coefficients error $\widetilde{\boldsymbol{w}} = \boldsymbol{w}^o - \boldsymbol{w}_{i-1}$, (14) becomes

$$\begin{aligned} \widetilde{\boldsymbol{w}}_i &= \widetilde{\boldsymbol{w}}_{i-1} - (\bar{\mu} + \mu')\, \boldsymbol{u}_i^* e(i) \\ \mu' &= \sum_{k=2}^{N} (-1)^{k-1} \|\boldsymbol{u}_i\|^{2(k-1)} \left[\sum \binom{\{\eta_n \mu_n\}}{k} \right], \end{aligned} \tag{16}$$

where $\bar{\mu} = \sum_{n=1}^{N} \eta_n \mu_n$ and $\binom{\{\eta_n \mu_n\}}{k}$ is taken as the set of $\binom{N}{k}$ products of k-combinations of $\{\eta_n \mu_n\}$. The time index on the supervisor parameters were omitted for clarity's sake. Notice that (16) is algebraically similar to a variable step size (VSS) algorithm. Its conceptual motivation, however, comes from incremental strategies in distributed optimization[1] [22], [23].

In order to proceed, the following assumptions are made:

A.1 (*Data independence*) $\{\boldsymbol{u}_i\}$ constitutes an i.i.d. sequence of vectors independent of the noise $v(j)$, $\forall\, i, j$. Consequently, $\boldsymbol{u}_i$ is independent of $\widetilde{\boldsymbol{w}}_j$, $\forall\, i > j$.

A.2 (*Supervisor separation principle*) $\eta_n(i)$ varies slowly when compared to $\boldsymbol{u}_i$ and, consequently, $e_a(i) = \boldsymbol{u}_i \widetilde{\boldsymbol{w}}_{i-1}$, the *a priori* error. Hence, the expected value of these variables can be separated as in $\mathrm{E}[\eta_n(i)\boldsymbol{u}_i] = \mathrm{E}\,\eta_n(i)\,\mathrm{E}\,\boldsymbol{u}_i$ and $\mathrm{E}[\eta_n(i)e_a(i)] = \mathrm{E}\,\eta_n(i)\,\mathrm{E}\,e_a(i)$.

The former is commonly adopted in the adaptive filtering literature and even though it is usually taken as an approximation, it accurately describes the behavior of AF's inputs in some applications [18]. The latter was inspired by the separation principle [18] and has been employed in the analysis of both convex and affine combinations [24]–[27].

From the data model (1), $e(i) = e_a(i) + v(i)$. Thus, under A.1 and A.2, the expected value of (16) becomes

$$\mathrm{E}\,\widetilde{\boldsymbol{w}}_i = \mathrm{E}\,\widetilde{\boldsymbol{w}}_{i-1} - \mathrm{E}(\bar{\mu} + \mu')\boldsymbol{u}_i^* e_a(i),$$

which yields the mean global coefficients error recursion

$$\begin{aligned} \mathrm{E}\,\widetilde{\boldsymbol{w}}_i &= [\boldsymbol{I} - \mathrm{E}\,\bar{\mu}\boldsymbol{R}_{\boldsymbol{u}} + \boldsymbol{\mathcal{M}}]\,\mathrm{E}\,\widetilde{\boldsymbol{w}}_{i-1} \\ \boldsymbol{\mathcal{M}} &= \sum_{k=2}^{N} \mathrm{E}(-\boldsymbol{u}_i^* \boldsymbol{u}_i)^k\, \mathrm{E}\left[\sum \binom{\{\eta_n \mu_n\}}{k} \right], \end{aligned} \tag{17}$$

with $\boldsymbol{R}_{\boldsymbol{u}} = \mathrm{E}\,\boldsymbol{u}_i^* \boldsymbol{u}_i$, the correlation matrix of the input signal. Even though (17) is a closed-form recursion dependent only on the data and the supervisor, the higher order moments required to calculate $\boldsymbol{\mathcal{M}}$ can be intricate to evaluate, especially for

[1] A similar interplay takes place in the NLMS recursion [18]

large N. Nevertheless, assuming the step sizes are small—more precisely, for $\eta_n\mu_n \ll 1$—, the terms for $k > 2$ are negligible and (17) can be approximated using

$$\boldsymbol{\mathcal{M}}' = \sum_{m\neq n} \mu_m\mu_n \,\mathrm{E}\,\eta_m\eta_n \,\mathrm{E}\,\boldsymbol{u}_i^*\boldsymbol{u}_i\boldsymbol{u}_i^*\boldsymbol{u}_i. \tag{18}$$

Empirical observations suggest that this approximation is accurate even for a large number of components—$N \gg 10$ (Fig. 3).

Finally, for Gaussian inputs, (18) can be evaluated analytically. Taking the eigenvalue decomposition $\boldsymbol{R_u} = \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^*$, where $\boldsymbol{\Lambda} = \mathrm{diag}\{\lambda_\ell\}$ is the diagonal eigenvalue matrix and $\boldsymbol{Q}$ is the unitary eigenvector matrix, define the transformed quantities $\bar{\boldsymbol{u}}_i = \boldsymbol{u}_i\boldsymbol{Q}^*$ and $\bar{\boldsymbol{w}}_i = \boldsymbol{Q}\tilde{\boldsymbol{w}}_i$. Notice that $\mathrm{E}\,\bar{\boldsymbol{u}}_i^*\bar{\boldsymbol{u}}_i = \boldsymbol{\Lambda}$ and that since $\boldsymbol{Q}$ is an isometry, $\bar{\boldsymbol{w}}_i$ will vanish iff $\tilde{\boldsymbol{w}}_i$ vanishes as well [18]. Hence, using the fourth-order moment result from [18], (18) becomes

$$\begin{aligned} \mathrm{E}\,\bar{\boldsymbol{w}}_i &= \boldsymbol{A}\cdot\mathrm{E}\,\bar{\boldsymbol{w}}_{i-1} \\ \boldsymbol{A} &= \boldsymbol{I} - \mathrm{E}\,\bar{\mu}\boldsymbol{\Lambda} \\ &\quad + \sum_{m\neq n} \mu_m\mu_n \,\mathrm{E}\,\eta_m\eta_n \left[\boldsymbol{\Lambda}\,\mathrm{Tr}(\boldsymbol{\Lambda}) + \gamma\boldsymbol{\Lambda}^2\right], \end{aligned} \tag{19}$$

with $\gamma = 2$ for real-valued inputs and $\gamma = 1$ for complex-valued and circular $\boldsymbol{u}_i$. For $N = 2$—as in [15]—(19) is exact.

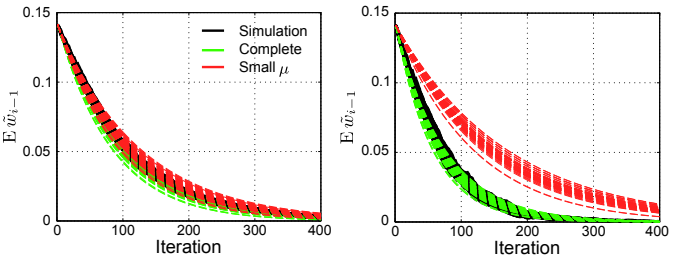
Comparing (19)—or (17)—with the model for a single LMS with step size μ—explicitly $\mathrm{E}\,\bar{\boldsymbol{w}}_i = [1-\mu\boldsymbol{\Lambda}]\,\mathrm{E}\,\bar{\boldsymbol{w}}_{i-1}$ [18]—it is clear that the incremental combination is able to improve the convergence rate by operating as though a larger step size ($\bar{\mu}$) were used. However, the reprocessing of the same data pair yields ever smaller gains in convergence speed, so that the net effect of the $\{\mathrm{LMS}\}^N$ is not equivalent to a single LMS filter with step size $\bar{\mu}$. This difference is quantified by $\boldsymbol{\mathcal{M}}$—or approximated by $\boldsymbol{\mathcal{M}}'$.

Finally, notice from (19) that the mean transient performance depends directly on the eigenvalues of $\boldsymbol{R_u}$, in contrast to the NLMS or the APA where $\boldsymbol{\Lambda}$ is normalized by some data non-linearity which reduces the eigenvalue spread [18], [28]. Hence, correlated inputs can worsen the convergence of the $\{\mathrm{LMS}\}^N$, motivating the use of the DR-$\{\mathrm{LMS}\}^N$ [16] in the parallel-incremental combination.

VI. Simulations

The data for the following simulations are taken from the zero mean Gaussian i.i.d. sequences $\{x(i)\}$ and $\{v(i)\}$, with $\sigma_x^2 = 1$ and $\sigma_v^2 = 10^{-3}$—SNR $= 30$ dB. White input experiments use $u(i) = x(i)$, whereas correlated inputs are generated using $u(i) = \alpha u(i-1) + \sqrt{1-\alpha^2}x(i)$ with $\alpha = 0.95$, which results in a highly correlated signal. All curves are averaged over 200 independent realizations.

Fig. 3 illustrates the accuracy of the mean convergence models (17)—*complete*—and (19)—*small* μ. Reduced step sizes and large N are employed to emphasize the effects of the small step sizes approximation. The higher order moments in (17) are evaluated using 10^4 Monte Carlo ensemble averages. Notice that (19) is accurate even when the number of components is large as long as the μ_n are small enough (Fig. 3a). In this case, due to the large value of N, a small change in the step sizes is enough to invalidate the model (Fig. 3b). Still, the complete model remains accurate until the adaptation is too fast for assumption A.1 to be valid.



(a) $\mu_n = 0.01$, $N = 75$ (b) $\mu_n = 0.02$, $N = 75$

Fig. 3. Mean convergence models ($M = 50$, $\eta_n = 1$, white inputs)

Simulations comparing the incremental combination with the APA and the NLMS were presented in Fig. 1[2]. The misadjustment of the AFs were equalized so as to better compare their performance. For white inputs (Fig. 1a), the DR-$\{\mathrm{LMS}\}^N$ was omitted as its curves almost match those of the $\{\mathrm{LMS}\}^N$. Nevertheless, DR combinations become clearly advantageous when the input data are correlated (Fig. 1b).

The DR-$\{\mathrm{LMS}\}^N$ has already been shown to outperform the APA with lower complexity [16]. Hence, the proposed LMS + DR-$\{\mathrm{LMS}\}^N$ is compared to a combination of APA with different K. Explicitly, the APA and NLMS—APA with $K = 1$—combination proposed in [17] and an improved version using the coefficients feedback from [14], both employing the normalized convex supervising rule (13).

Fig. 4a shows the combinations in a white input scenario. The new combination clearly outperforms the APA from Fig. 1a, but also the APA combination from [17]. Moreover, though the coefficients feedback improves the performance of the APA combination to the point of matching that of the parallel-incremental LMS, the complexity of the latter remains much lower (see Fig. 4a). Fig. 4b and 4c illustrate the components behavior under cyclic feedback and the evolution of the parallel supervisors $\eta(i)$, respectively.

Last, Fig. 5 presents the new combination under colored inputs. Although the performance of the DR-$\{\mathrm{LMS}\}^N$ in the correlated case was already noteworthy, the parallel-incremental combination further improves its misadjustment, again outperforming the APA combination from [17] and matching the coefficients feedback version—with considerably lower complexity. The components behaviors and the average supervisor parameters are presented in Fig. 5b and 5c.

VII. Conclusion

A parallel-incremental combination was presented motivated by the transient–steady-state trade-offs of the incremental topology. A simplified analysis was provided to demonstrate the effect on the convergence rate of incrementally combining AFs and the resulting model showed good agreement with numerical experiments. The new combination was shown to outperform the APA and combinations of APA in different scenarios. Future work includes the extension of this combination to more than two branches and mean-square analyses.

[2]Using $\eta_n\mu_n = \mu$ yields the faster convergence rates.

REFERENCES

[1] S. Shaffer and C.S. Williams, "Comparison of LMS, alpha LMS, and data reusing LMS algorithms," in *Asilomar Conference on Circuits, Systems, and Computers*, 1983, pp. 260–264.

[2] K. Ozeki and T. Umeda, "An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties," *Elec. Comm. Japan*, vol. J67-A, pp. 126–132, 1984.

[3] B.A. Schnaufer and W.K. Jenkins, "New data-reusing LMS algorithms for improved convergence," in *Asilomar Conf. on Signals, Syst. and Comput.*, 1993, pp. 1584–1588.

[4] J. Apolinário Jr., M.L.R. Campos, and P.S.R. Diniz, "Convergence analysis of the binormalized data-reusing LMS algorithm," *IEEE Trans. Signal Process.*, vol. 48[11], pp. 3235–3242, 2000.

[5] R.A. Soni, K.A. Gallivan, and W.K. Jenkins, "Low-complexity data reusing methods in adaptive filtering," *IEEE Trans. Signal Process.*, vol. 52[2], pp. 394–405, 2004.

[6] S.L. Gay and S. Tavathia, "The fast affine projection algorithm," in *International Conference in Acoustic, Speech, and Signal Processing*. IEEE, 1995, pp. 3023–3026.

[7] M.C. Tsakiris and P.A. Naylor, "Fast exact affine projection algorithm using displacement structure theory," in *International Conference on Digital Signal Processing*, 2009.

[8] Y.A. Huang, J. Benesty, and J. Chen, *Acoustic MIMO Signal Processing*, Springer, 2006.

[9] 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[3], pp. 1078–1090, 2006.

[10] C.G. Lopes, E.H. Satorius, P. Estabrook, and A.H. Sayed, "Adaptive carrier tracking for mars to earth communications during entry, descent, and landing," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 46[4], pp. 1865–1879, 2010.

[11] 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[5], pp. 1853—-1864, 2008.

[12] M.T.M. Silva and V.H. Nascimento, "Improving the tracking capability of adaptive filters via convex combination," *IEEE Trans. Signal Process.*, vol. 56[7], pp. 3137—-3149, 2008.

[13] L.F.O. Chamon and C.G. Lopes, "Combination of adaptive filters for relative navigation," in *EUSIPCO*, 2011, pp. 1771–1775.

[14] L.F.O. Chamon, W.B. Lopes, and C.G. Lopes, "Combination of adaptive filters with coefficients feedback," in *ICASSP*, 2012, pp. 3785–3788.

[15] W.B. Lopes and C.G. Lopes, "Incremental-cooperative strategies in combination of adaptive filters," in *ICASSP*, 2011, pp. 4132–4135.

[16] L.F.O. Chamon, H.F. Ferro, and C.G. Lopes, "A data reusage algorithm based on incremental combination of LMS filters," in *Asilomar Conference on Signals, Systems and Computers*, 2012.

[17] A. Gonzalez M. Ferrer, M. de Diego and G. Piñero, "Convex combination of affine projection algorithms," in *EUSIPCO*, 2009, pp. 431–435.

[18] A.H. Sayed, *Adaptive filters*, Wiley-IEEE Press, 2008.

[19] K. Ogata, *Modern Control Engineering*, Prentice Hall, 2009.

[20] L.A. Azpicueta-Ruiz, A.R. Figueiras-Vidal, and J. Arenas-García, "A normalized adaptation scheme for the convex combination of two adaptive filters," in *ICASSP*, 2008, pp. 3301–3304.

[21] F. Vahid and T.D. Givargis, *Embedded System Design: A Unified Hardware/Software Introduction*, John Wiley & Sons, 2001.

[22] D.P. Bertsekas, "A new class of incremental gradient methods for least squares problems," *SIAM J. Optim.*, vol. 7[4], pp. 913–926, 1997.

[23] C.G. Lopes and A.H. Sayed, "Incremental adaptive strategies over distributed networks," *IEEE Trans. Signal Process.*, vol. 55[8], pp. 4064–4077, 2007.

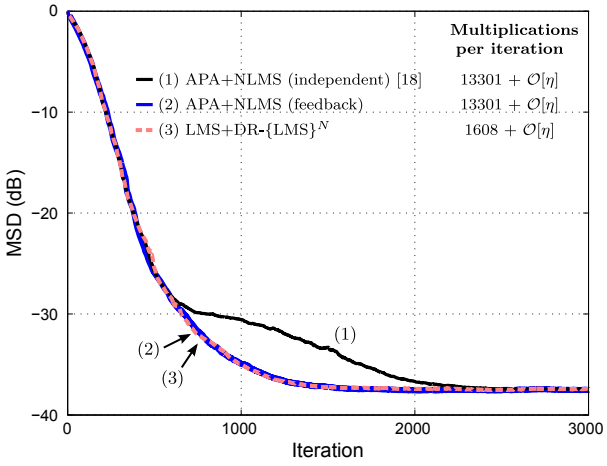
[24] M.T.M. Silva, V.H. Nascimento, and J. Arenas-García, "A transient analysis for the convex combination of two adaptive filters with transfer of coefficients," in *ICASSP*, 2010, pp. 3842—-3845.

[25] V.H. Nascimento, M.T.M. Silva, R. Candido, and J. Arenas-García, "A transient analysis for the convex combination of adaptive filters," in *SSP*, 2009, pp. 53–56.

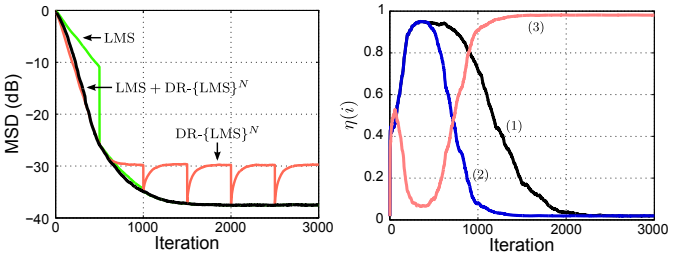
[26] R. Candido, M.T.M. Silva, and V.H. Nascimento, "Affine combinations of adaptive filters," in *Asilomar Conference on Signals, Syst. and Comput.*, 2008, pp. 236–240.

[27] S.S. Kozat, A.T. Erdogan, A.C. Singer, and A.H. Sayed, "Transient analysis of adaptive affine combinations," *IEEE Trans. Signal Process.*, vol. 59[12], pp. 6227–6232, 2011.

[28] H.-C. Shin and A.H. Sayed, "Mean-square performance of a family of affine projection algorithms," *IEEE Trans. Signal Process.*, vol. 52[1], pp. 90–102, 2004.

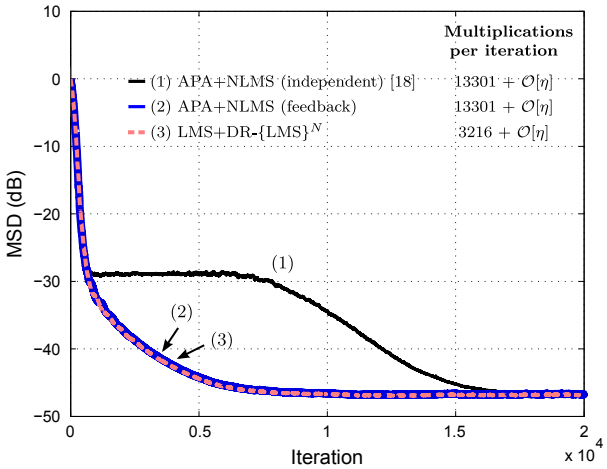


(a) Overall performance

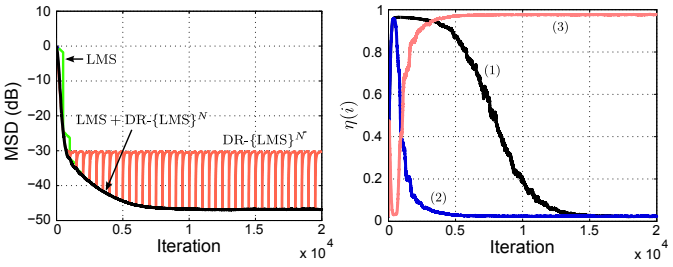


(b) Components (c) Supervisor

Fig. 4. **Parallel-incremental combination (white input)** ($M = 100$, $K = 10$, $\eta_n = 1$, and $\mu_n = \mu$). **APA+NLMS**: $\mu_{\text{APA}} = 0.3$, $\mu_{\text{NLMS}} = 0.3$, $\mu_a = 0.8$, $\beta = 0.95$, $\epsilon = 10^{-4}$, $L = 400$; **LMS+DR-{LMS}^{N}**: $\mu_{\text{LMS}} = 3 \cdot 10^{-3}$, $\mu_{\text{DR-\{LMS\}}^N} = 6 \cdot 10^{-3}$, $N = 7$, $\mu_a = 0.7$, $\beta = 0.95$, $\epsilon = 10^{-4}$, $L = 400$.



(a) Overall performance



(b) Components (c) Supervisor

Fig. 5. **Parallel-incremental combination (correlated input)** ($M = 100$, $K = 10$, $\eta_n = 1$, and $\mu_n = \mu$). **APA+NLMS**: $\mu_{\text{APA}} = 1$, $\mu_{\text{NLMS}} = 0.04$, $\mu_a = 0.8$, $\beta = 0.95$, $\epsilon = 10^{-4}$, $L = 400$; **LMS+DR-{LMS}^{N}**: $\mu_{\text{LMS}} = 4 \cdot 10^{-4}$, $\mu_{\text{DR-\{LMS\}}^N} = 2 \cdot 10^{-3}$, $N = 15$, $\mu_a = 0.7$, $\beta = 0.95$, $\epsilon = 10^{-4}$, $L = 500$.