



# The Krylov-proportionate normalized least mean fourth approach: Formulation and performance analysis

Muhammed O. Sayin<sup>a</sup>, Yasin Yilmaz<sup>b</sup>, Alper Demir<sup>c</sup>, Suleyman S. Kozat<sup>a,\*</sup>

<sup>a</sup> Department of Electrical and Electronics Engineering, Bilkent University, Ankara, Turkey

<sup>b</sup> Department of Electrical Engineering, Columbia University, New York, USA

<sup>c</sup> Department of Electrical and Computer Engineering, Koc University, Istanbul, Turkey

## ARTICLE INFO

### Article history:

Received 29 January 2014

Received in revised form

9 October 2014

Accepted 13 October 2014

Available online 4 November 2014

### Keywords:

Krylov subspace

NLMF

Proportional update

Transient analysis

Steady-state analysis

Tracking performance

## ABSTRACT

We propose novel adaptive filtering algorithms based on the mean-fourth error objective while providing further improvements on the convergence performance through proportionate update. We exploit the sparsity of the system in the mean-fourth error framework through the proportionate normalized least mean fourth (PNLMF) algorithm. In order to broaden the applicability of the PNLMF algorithm to dispersive (non-sparse) systems, we introduce the Krylov-proportionate normalized least mean fourth (KPNLMF) algorithm using the Krylov subspace projection technique. We propose the Krylov-proportionate normalized least mean mixed norm (KPNLMMN) algorithm combining the mean-square and mean-fourth error objectives in order to enhance the performance of the constituent filters. Additionally, we propose the stable-PNLMF and stable-KPNLMF algorithms overcoming the stability issues induced due to the usage of the mean fourth error framework. Finally, we provide a complete performance analysis, i.e., the transient and the steady-state analyses, for the proportionate update based algorithms, e.g., the PNLMF, the KPNLMF algorithms and their variants; and analyze their tracking performance in a non-stationary environment. Through the numerical examples, we demonstrate the match of the theoretical and ensemble averaged results and show the superior performance of the introduced algorithms in different scenarios.

© 2014 Elsevier B.V. All rights reserved.

## 1. Introduction

Many signal processing problems such as noise removal, e.g., recent works [1–3], echo cancellation, e.g., recent works [4–7], and channel equalization, e.g., recent works [8,9], can be formulated in the general system-identification framework depicted in Fig. 1. In this framework, we model the unknown system adaptively by minimizing a certain statistical measure of the error  $e_t$  between the output of the

unknown system  $d_t$  and the model system  $\hat{d}_t$ . Minimization in the mean square error (MSE) sense is the most widely known and used technique providing tractability and relative ease of analysis. As an alternative, we consider the minimization of the mean-fourth error, which is shown to improve performance compared to the conventional MSE objective with a considerable margin in certain scenarios [10–12]. In this context, the normalized least mean fourth (NLMF) algorithm is shown to achieve faster convergence performance through the independence of the input data correlation statistics in certain settings [13–15].

In this paper, we seek to enhance the performance of the NLMF algorithm further. We first derive the proportionate normalized least mean fourth (PNLMF) algorithm

\* Corresponding author. Tel.: +90 312 290 2336.

E-mail addresses: [sayin@ee.bilkent.edu.tr](mailto:sayin@ee.bilkent.edu.tr) (M.O. Sayin),

[yasin@ee.columbia.edu](mailto:yasin@ee.columbia.edu) (Y. Yilmaz), [aldemir@ku.edu.tr](mailto:aldemir@ku.edu.tr) (A. Demir),

[kozat@bilkent.edu.tr](mailto:kozat@bilkent.edu.tr) (S.S. Kozat).

based on the proportionate update and the mean fourth error framework. The proportionate update exploits the sparsity of the underlying system by updating each component of the estimate  $\mathbf{w}_t$  independently [6]. In the echo-cancellation framework, the proportionate least mean-square (PNLMS) algorithms are shown to converge faster for the sparse echo paths [6,16]. We note that the convergence performance of the conventional PNLMS algorithm degrades significantly in the dispersive systems. In [17], authors propose an improved PNLMS (IPNLMS) algorithm providing enhanced performance independent of the sparsity of the impulse response of the system. Hence, in the derivation of the PNLMF algorithm we follow a similar approach with [17] to increase the reliability of our novel algorithms and our algorithm PNLMF further improves the convergence performance of the IPNLMS algorithm for certain scenarios.

Furthermore, we introduce the Krylov-proportionate normalized least mean fourth (KPNLMF) algorithm [18]. Here, the Krylov subspace projection technique is incorporated within the framework of the PNLMF algorithm. The Krylov-proportionate normalized least mean square (KPNLMS) algorithm, introduced in [19–21], extends the use of the IPNLMS algorithm to the identification of dispersive systems. Our KPNLMF algorithm inherits the advantageous features of the KPNLMS for the dispersive systems in addition to the benefits of the mean-fourth error objective. We note that a mixture combination of the mean-square and the mean-fourth error objectives is shown to outperform both of the constituent filters [22]. Hence, we propose the Krylov-proportionate normalized least mean mixed norm (KPNLMMN) algorithm having a convex combination of the mean-square and the mean-fourth error objectives. In addition, we point out that the stability of the mean-fourth error based algorithms depends on the initial value of the adaptive filter weights, the input and noise power [23–25]. In order to enhance the stability of the introduced algorithms, we further introduce the stable-PNLMF and the stable-KPNLMF algorithms [24,25]. Finally we provide a complete performance analysis for the introduced algorithms, i.e., the transient, the steady-state and the tracking performance analyses. We evaluate convergence performance of our algorithms and compare them with the well-known example algorithms under several different configurations through numerical examples. We observe that the introduced algorithms achieve superior performance in different scenarios.

Our main contributions include the following: (1) We derive the PNLMF algorithm suitable for sparse systems such as for echo-cancellation frameworks based on the natural gradient descent framework and propose the stable-PNLMF algorithm avoiding the stability issues induced due to the mean-fourth error objective. (2) We derive the KPNLMF algorithm utilizing the Krylov projection technique, which broadens the applicability of the PNLMF algorithm to the non-sparse systems. (3) We introduce the KPNLMMN and the stable-KPNLMF algorithms achieving better trade-off in terms of the transient and steady-state performance under certain settings. (4) We provide a complete performance analysis, i.e., the transient and the steady state analyses; and analyze the tracking performance in a non-stationary environment. (5)

We demonstrate the improved convergence performance of the proposed algorithms through several numerical examples under different scenarios.

The paper is organized as follows. In Section 2, we describe the system identification framework for the mean-square and the mean-fourth error objectives. We formulate the PNLMF and KPNLMF algorithms, and their variants in Sections 3 and 4, respectively. We propose a new simplification scheme reducing the computational complexity of the Krylov-proportionate update based algorithms further in Section 5. We carry out a complete performance analysis of the algorithms in Section 6. Section 7 contains the simulation results for the different configurations followed by the concluding remarks in Section 8.

*Notation:* All vectors are column vectors represented by boldface lowercase letters,  $[\cdot]^T$ ,  $\|\cdot\|$  and  $|\cdot|$  are the transpose,  $l_2$ -norm and the absolute value operators, respectively. For a vector  $\mathbf{x}$ ,  $x^{(i)}$  is the  $i$ th entry. Matrices are represented with boldface capital letters. For a random variable  $x$  (or vector  $\mathbf{x}$ ),  $E[x]$  (or  $E[\mathbf{x}]$ ) is the expectation. Time index appears as a subscript, e.g.,  $\mathbf{x}_t$ .

## 2. System description

Consider the system identification task given in Fig. 1. The output of the unknown system is given by

$$d_t = \mathbf{w}_0^T \mathbf{x}_t + v_t, \quad t \in \mathbb{N},$$

where  $\mathbf{x}_t \in \mathbb{R}^M$  is the zero-mean input regressor vector,  $\mathbf{w}_0 \in \mathbb{R}^M$  is the coefficient vector of the unknown system to be identified and  $v_t \in \mathbb{R}$  is the zero-mean noise assumed to be independent and identically distributed (i.i.d.) with variance  $\sigma_v^2$ . Although we assume a time invariant desired vector  $\mathbf{w}_0$  here, we also provide the tracking performance analysis for certain non-stationary models later in the paper. We assume that the input regressor  $\mathbf{x}_t$  and the noise  $v_t$  are independent as is common in the analysis of traditional adaptive schemes [33]. We note that the system identification task also models the conventional high-level echo-cancellation framework where the signal  $\mathbf{x}_t$  denotes the far-end signal that excites the echo path,  $v_t$  is the near-end noise signal,  $d_t$  corresponds to the near-end signal, and  $\mathbf{w}_0$  represents the unknown echo-path impulse response [16].

Given the input regressor, the estimate of the output signal is given by

$$\hat{d}_t = \mathbf{w}_t^T \mathbf{x}_t, \quad t \in \mathbb{N},$$

where  $\mathbf{w}_t = [w_t^{(1)}, w_t^{(2)}, \dots, w_t^{(M)}]^T$  is the adaptive weight

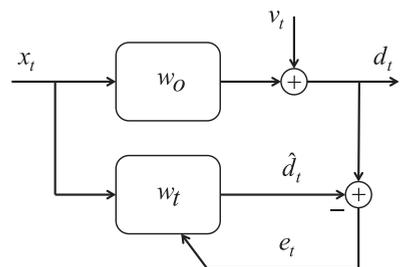


Fig. 1. Block diagram of the system identification task.

vector that estimates  $\mathbf{w}_0$ . In this framework, we aim to minimize a specific statistical measure of the error between the output signal  $d_t$  and the estimate produced by the adaptive algorithm  $\hat{d}_t$ , i.e.,  $e_t \triangleq d_t - \hat{d}_t$ . The mean square error (MSE),  $E[e_t^2]$ , and the mean fourth error (MFE),  $E[e_t^4]$ , are two popular choices to minimize.

In the next sections, we introduce several adaptive filtering algorithms in the system identification framework that are constructed based on the MSE and MFE criteria through the proportional update idea and the Krylov-subspace-projection technique.

### 3. Proportionate update approach

In the well-known and popular gradient descent method, we seek to converge to a local minimum of a given cost function, e.g.,  $J(d_t, \mathbf{x}_t, \mathbf{w}) = E[(d_t - \mathbf{x}_t^T \mathbf{w})^4]$ , irrespective of the unknown parameter space [33]. However, in the proportionate update approach, we consider the cases where the unknown parameters are sparse or quasi-sparse, where most of the terms in the true parameter vector, i.e.,  $\mathbf{w}_0$ , are close to zero. For such cases, different from the conventional gradient descent methods, the natural gradient adaptation aims to exploit the near sparseness of the parameter space for faster convergence to the local minimum [26]. Instead of an Euclidean space, the natural gradient descent adaptation utilizes a Riemannian metric structure, which is introduced in [27]. Assume that  $S = \{\mathbf{w} \in \mathbb{R}^M\}$  is a Riemannian parameter space on which we define the cost function  $J(d, \mathbf{x}, \mathbf{w})$ . Then, the distance between the current parameter vector  $\mathbf{w}_t$  and the next parameter vector  $\mathbf{w}_{t+1}$  is defined as

$$D(\mathbf{w}_{t+1}, \mathbf{w}_t) \triangleq (\mathbf{w}_{t+1} - \mathbf{w}_t)^T \Theta_t (\mathbf{w}_{t+1} - \mathbf{w}_t), \quad (1)$$

$$\triangleq \|\mathbf{w}_{t+1} - \mathbf{w}_t\|_{\Theta_t}^2,$$

where  $\Theta_t \in \mathbb{R}^{M \times M}$  denotes the Riemannian metric tensor describing the local curvature of the parameter space and depends on  $\mathbf{w}_t$  in general [26]. A formulation of the proportionate update based algorithms using the natural gradient descent adaptation has been studied in [28,29]. Particularly, in this paper, we define  $\Theta_t \triangleq \mathbf{G}_t^{-1}$  and  $\mathbf{G}_t$  is given by

$$\mathbf{G}_t \triangleq \text{diag}(\phi_t^{(1)}, \phi_t^{(2)}, \dots, \phi_t^{(M)}), \quad (2)$$

$$\phi_t^{(k)} \triangleq (1-\gamma) \frac{1}{M} + \gamma \frac{|w_t^{(k)}|}{\|\mathbf{w}_t\|_1 + \kappa}, \quad k = 1, \dots, M,$$

where  $\gamma \in (0, 1)$  is the proportionality factor and  $\kappa$  is a small regularization constant [17]. However, note that  $\gamma = (\alpha + 1)/2$  for the  $\alpha$  used in [17].

We note that we can derive most of the conventional adaptive filtering algorithms through the following generic update [30,31]:

$$\mathbf{w}_{t+1} = \arg \min_{\mathbf{w}} \{D(\mathbf{w}, \mathbf{w}_t) + \eta J(d_t, \mathbf{x}_t, \mathbf{w})\}. \quad (3)$$

Hence, after some algebra for the Riemannian metric tensor  $\Theta_t$  and the stochastic cost function  $J(d_t, \mathbf{x}_t, \mathbf{w})$ , the natural gradient descent algorithm yields

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \eta \Theta_t^{-1} \nabla_{\mathbf{w}} J(d_t, \mathbf{x}_t, \mathbf{w})|_{\mathbf{w} = \mathbf{w}_t}, \quad (4)$$

where  $\eta > 0$  is the step size. As an example, for  $J_1(d_t, \mathbf{x}_t, \mathbf{w}) \triangleq (d_t - \mathbf{x}_t^T \mathbf{w})^2$ , (4) yields the IPNLMS algorithm [17] as

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \mu e_t \frac{\mathbf{G}_t \mathbf{x}_t}{\mathbf{x}_t^T \mathbf{G}_t \mathbf{x}_t + \epsilon} \quad (5)$$

by letting  $\eta = \mu / (\mathbf{x}_t^T \mathbf{G}_t \mathbf{x}_t + \epsilon)$  and  $\epsilon > 0$  denotes the regularization factor. Note that for a stationary regression signal and given signal-to-noise ratio (SNR) which is defined as  $E[\mathbf{w}_0^T \mathbf{x}_t \mathbf{x}_t^T \mathbf{w}_0] / E[v_t^2]$ , we can choose the regularization factor as [32]

$$\epsilon = \frac{1 + \sqrt{1 + \text{SNR}}}{\text{SNR}} \sigma_x^2.$$

However, when any *a priori* information on the SNR is not available, the determination of the regularization constant requires special care.

We emphasize that the proportionate update (5) distinguishes frequently used, rarely used and unused coefficients; and updates them separately with different step sizes. In particular, we update each filter coefficient based on the absolute value in a proportional manner. Hence, we seek to employ the proportionate update idea in the MFE framework. To this end, for the stochastic cost function  $J_2(d_t, \mathbf{x}_t, \mathbf{w}) \triangleq (d_t - \mathbf{x}_t^T \mathbf{w})^4$ , we obtain the PNLMF algorithm [18], given by

$$\mathbf{w}_{t+1} = \mathbf{w}_t + 2\mu e_t^3 \frac{\mathbf{G}_t \mathbf{x}_t}{\mathbf{x}_t^T \mathbf{G}_t \mathbf{x}_t + \epsilon}.$$

We point out that the PNLMF algorithm outperforms the NLMS and NLMF algorithms when the system to be identified is sparse. However, the PNLMF algorithm has stability issues due to the mean-fourth error objective. In order to overcome this issue, we propose the stable-PNLMF algorithm defined as

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \frac{2\mu \mathbf{G}_t \mathbf{x}_t e_t^3}{\mathbf{x}_t^T \mathbf{G}_t \mathbf{x}_t + (\mathbf{x}_t^T \mathbf{G}_t \mathbf{x}_t + e_t^2)}, \quad (6)$$

similar to the stable-NLMF algorithm [24,25]. In practice, in order to avoid a division by zero, we also propose the regularized stable-PNLMF algorithm modifying (6) such that

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \frac{2\mu \mathbf{G}_t \mathbf{x}_t e_t^3}{(\mathbf{x}_t^T \mathbf{G}_t \mathbf{x}_t + \epsilon)(\mathbf{x}_t^T \mathbf{G}_t \mathbf{x}_t + e_t^2)}.$$

We note that the stable-PNLMF algorithm (6) updates its coefficients similar to the IPNLMS algorithm at the initial stages of the adaptation where the estimation error is relatively large. However, for small error values, the stable-PNLMF algorithm updates akin to the PNLMF algorithm, which yields smaller steady-state error.

In the next section, we extend the enhanced performance of the proportionate update idea to dispersive (non-sparse) systems using the Krylov subspace projection technique in the mean fourth error framework.

### 4. Projection onto the Krylov subspace

We can utilize the proportionate update approach in a dispersive system ( $S = \{\mathbf{w} \in \mathbb{R}^M\}$  is an Euclidean parameter space) through the projection of the unknown system onto

the Krylov subspace. To this end, we define

$$K_M(\hat{\mathbf{R}}, \hat{\mathbf{p}}) \triangleq [\hat{\mathbf{p}}, \hat{\mathbf{R}}\hat{\mathbf{p}}, \hat{\mathbf{R}}^2\hat{\mathbf{p}}, \dots, \hat{\mathbf{R}}^{M-1}\hat{\mathbf{p}}], \quad (7)$$

whose column vectors span the Krylov subspace [19]. We denote the estimates of the autocorrelation matrix of the regressor and the cross-correlation vector between the input regressor  $\mathbf{x}_t$  and the output  $d_t$  through  $\hat{\mathbf{R}}$  and  $\hat{\mathbf{p}}$ , respectively. We construct the orthogonal matrix  $\mathbf{Q} \in \mathbb{R}^{M \times M}$  by orthonormalizing the columns of  $K_M(\hat{\mathbf{R}}, \hat{\mathbf{p}})$ . Through the orthogonal matrix  $\mathbf{Q}$ , in [20], the author shows that the projected system  $\mathbf{w}_0^* \triangleq \mathbf{Q}^T \mathbf{w}_0$  has a sparse structure provided that the input regressor  $\mathbf{x}_t$  is nearly white, i.e.,  $\hat{\mathbf{R}} \approx \mathbf{I}$ . In particular, if the autocorrelation matrix  $\hat{\mathbf{R}}$  of the input regressor  $\mathbf{x}_t$  has clustered eigenvalues or a condition number that is close to one, then any unknown system will have a sparse representation under the new Krylov subspace coordinates [21]. However, for the colored input signal, we can use a preconditioning, i.e., whitening, process before the projection onto the Krylov subspace [19].

We define the projected weight vector as  $\hat{\mathbf{w}}_t \triangleq \mathbf{Q}^T \mathbf{w}_t$ . Then, the projected parameter space  $\hat{S} = \mathbf{Q}^T \mathbf{w} \in \mathbb{R}^M$  is a Riemannian parameter space and we can use the natural gradient descent update as follows:

$$\hat{\mathbf{w}}_{t+1} = \hat{\mathbf{w}}_t + \eta \hat{\Theta}_t^{-1} \nabla_{\hat{\mathbf{w}}} J(d_t, \mathbf{Q}^T \mathbf{x}_t, \hat{\mathbf{w}}) \Big|_{\hat{\mathbf{w}} = \hat{\mathbf{w}}_t}, \quad (8)$$

where we also project the regression signal onto the Krylov subspace so that the error is given by  $e_t = d_t - (\mathbf{Q}^T \mathbf{x}_t)^T (\mathbf{Q}^T \mathbf{w}_t) = d_t - \mathbf{x}_t^T \mathbf{w}_t$  since  $\mathbf{Q}$  is an orthonormal matrix, i.e.,  $\mathbf{Q}^T \mathbf{Q} = \mathbf{I}$ . However, we note that  $\hat{\Theta}_t \triangleq \hat{\mathbf{G}}_t^{-1}$  and  $\hat{\mathbf{G}}_t$  is given by

$$\hat{\mathbf{G}}_t \triangleq \text{diag}(\hat{\phi}_t^{(1)}, \hat{\phi}_t^{(2)}, \dots, \hat{\phi}_t^{(M)}),$$

$$\hat{\phi}_t^{(k)} \triangleq (1 - \gamma) \frac{1}{M} + \gamma \frac{|\hat{w}_t^{(k)}|}{\|\hat{\mathbf{w}}_t\|_1 + \kappa}, \quad k = 1, \dots, M. \quad (9)$$

In the original coordinates by multiplying both sides of (8) from left with  $\mathbf{Q}$ , we obtain the following update:

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \eta \mathbf{Q} \hat{\Theta}_t^{-1} \nabla_{\hat{\mathbf{w}}} J(d_t, \mathbf{Q}^T \mathbf{x}_t, \hat{\mathbf{w}}) \Big|_{\hat{\mathbf{w}} = \hat{\mathbf{w}}_t}.$$

By letting  $\eta = \mu / (\mathbf{x}_t^T \mathbf{Q} \hat{\mathbf{G}}_t \mathbf{Q}^T \mathbf{x}_t + \epsilon)$  and for the square error cost  $J_1(d_t, \mathbf{x}_t, \mathbf{w})$ , we obtain the KPNLMS algorithm [21], given by

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \mu e_t \frac{\mathbf{Q} \hat{\mathbf{G}}_t \mathbf{Q}^T \mathbf{x}_t}{\mathbf{x}_t^T \mathbf{Q} \hat{\mathbf{G}}_t \mathbf{Q}^T \mathbf{x}_t + \epsilon}.$$

Correspondingly, the fourth error cost  $J_2(d_t, \mathbf{x}_t, \mathbf{w})$  yields the KPNLMF algorithm [18] as

$$\mathbf{w}_{t+1} = \mathbf{w}_t + 2\mu e_t^3 \frac{\mathbf{Q} \hat{\mathbf{G}}_t \mathbf{Q}^T \mathbf{x}_t}{\mathbf{x}_t^T \mathbf{Q} \hat{\mathbf{G}}_t \mathbf{Q}^T \mathbf{x}_t + \epsilon}. \quad (10)$$

In [22], the authors demonstrate that a mixture combination of the mean-square and mean-fourth error objectives achieve superior performance with respect to both of the constituent filter. In that sense, we propose the KPNLMMN algorithm given by

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \mu (\delta e_t + 2(1 - \delta) e_t^3) \frac{\mathbf{Q} \hat{\mathbf{G}}_t \mathbf{Q}^T \mathbf{x}_t}{\mathbf{x}_t^T \mathbf{Q} \hat{\mathbf{G}}_t \mathbf{Q}^T \mathbf{x}_t + \epsilon},$$

where  $\delta \in [0, 1]$  is the combination weight. Finally, the extension of the stable-PNLMF algorithm to be used in the dispersive systems through the Krylov-subspace projection technique leads to the following algorithm, i.e., the stable-KPNLMF algorithm, as

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \frac{2\mu \mathbf{Q} \hat{\mathbf{G}}_t \mathbf{Q}^T \mathbf{x}_t e_t^3}{\mathbf{x}_t^T \mathbf{Q} \hat{\mathbf{G}}_t \mathbf{Q}^T \mathbf{x}_t (\mathbf{x}_t^T \mathbf{Q} \hat{\mathbf{G}}_t \mathbf{Q}^T \mathbf{x}_t + e_t^2)}.$$

We point out that we can estimate  $\mathbf{R} = E[\mathbf{x}_t \mathbf{x}_t^T]$  and  $\mathbf{p} = E[\mathbf{x}_t d_t]$ , recursively, in the initial stages of the adaptation such that

$$\hat{\mathbf{R}}_{t+1} = \hat{\mathbf{R}}_t + \mathbf{x}_t \mathbf{x}_t^T,$$

$$\hat{\mathbf{p}}_{t+1} = \hat{\mathbf{p}}_t + \mathbf{x}_t d_t,$$

for  $t \in \{1, \dots, T_0\}$ . During the estimation stage we can update  $\mathbf{w}_t$  through the NLMF algorithm, i.e.,  $\hat{\mathbf{G}}_t = \mathbf{I}$ . Once we have estimated  $\mathbf{R}$  and  $\mathbf{p}$ , we can construct the Krylov vectors. However, the explicit generation of Krylov vectors is an ill-conditioned numerical operation. The well-known *Gram-Schmidt method* does not help here as it first generates the Krylov vectors and then orthonormalizes them. We can perform the orthonormalization via *Arnoldi's method* since it does not explicitly generate Krylov vectors [34,35]. Furthermore, we construct  $\mathbf{Q}$  only once in the algorithm, hence this calculation does not bring significant additional computational burden for the updates.

In the sequel, we discuss the approaches to reduce the computational complexity of the introduced algorithms.

## 5. Algorithms with reduced computational complexity

In this section, we examine several approaches to reduce the computational complexity of the update for  $\mathbf{w}_t$ . We note that at each time  $t$  computing  $\hat{\mathbf{G}}_t$  (9) and then  $\mathbf{Q} \hat{\mathbf{G}}_t \mathbf{Q}^T \mathbf{x}_t$ , in general, have a complexity of  $O(M^2)$  unless the matrix  $\mathbf{Q} \hat{\mathbf{G}}_t \mathbf{Q}^T$  has a special structure. Hence, the algorithm given in (10) is computationally intensive. However, we can attain linear computational complexity per iteration, i.e.,  $O(M)$ , as follows.

In [21], the authors demonstrate that whenever the projected vector  $\mathbf{Q}^T \mathbf{w}_0$  is sparse (i.e.,  $\hat{\mathbf{R}}$  has one of the properties:  $\hat{\mathbf{p}}$  is an eigenvector of  $\hat{\mathbf{R}}$  or eigenvalues of  $\hat{\mathbf{R}}$  are clustered or eigenvalue-spread of  $\hat{\mathbf{R}}$  is close to 1), the nonzero entries are concentrated in the first few elements in terms of the  $l_2$ -norm (Euclidean norm). Similarly, the projected weight vector  $\hat{\mathbf{w}}_t$  has its nonzero entries mainly in the first few elements. Hence, in [20], the author approximates  $\hat{\mathbf{G}}_t$  with the following simplified matrix:

$$\tilde{\mathbf{G}}_t \triangleq \text{diag}(\tilde{\phi}_t^{(1)}, \dots, \tilde{\phi}_t^{(\lambda)}, \psi_t, \dots, \psi_t),$$

$$\tilde{\phi}_t^{(k)} \triangleq (1 - \gamma) \frac{1}{M} + \gamma \frac{|\hat{w}_t^{(k)}|}{\delta_t + \kappa},$$

$$\psi_t \triangleq (1 - \gamma) \frac{1}{M} + \gamma \frac{\zeta \tau_t}{\delta_t + \kappa}, \quad (11)$$

where

$$\tau_t \triangleq \frac{1}{\lambda} \sum_{l=1}^{\lambda} |\hat{w}_t^{(l)}|, \quad \delta_t \triangleq (\lambda + \zeta(M - \lambda)) \tau_t$$

and  $\zeta$  is a pre-specified small constant. However, in this paper, we seek to achieve computationally more efficient

algorithms. To this end, instead of (11), we approximate  $\hat{\mathbf{G}}_t$  with

$$\begin{aligned} \bar{\mathbf{G}}_t &\triangleq \text{diag}\{\bar{\phi}_t^{(1)}, \dots, \bar{\phi}_t^{(\lambda)}, \psi, \dots, \psi\}, \\ \bar{\phi}_t^{(k)} &\triangleq (1-\gamma)\frac{1}{M} + \gamma\frac{|\hat{w}_t^{(k)}|}{\sum_{l=1}^{\lambda} |\hat{w}_t^{(l)}| + \kappa}, \quad k=1, \dots, \lambda, \end{aligned} \quad (12)$$

where  $\psi = (1-\gamma)/M > 0$ , i.e., we assume that  $\hat{w}_t^{(k)} \approx 0$  for all  $k \in \{\lambda+1, \dots, M\}$ . Then, as in [20], we define  $\bar{\mathbf{\Omega}}_t \triangleq \mathbf{Q}\bar{\mathbf{G}}_t\mathbf{Q}^T$  and  $\mathbf{Q}_\lambda \in \mathbb{R}^{M \times \lambda}$  as the first  $\lambda$  columns of  $\mathbf{Q}$  such that  $\mathbf{Q} = [\mathbf{Q}_\lambda \mathbf{Q}_{M-\lambda}]$ . Then, we can compute  $\bar{\mathbf{\Omega}}_t \mathbf{x}_t$  through

$$\begin{aligned} \bar{\mathbf{\Omega}}_t \mathbf{x}_t &= [\mathbf{Q}_\lambda \mathbf{Q}_{M-\lambda}] \begin{bmatrix} \bar{\mathbf{G}}_{t,\lambda} & \mathbf{0} \\ \mathbf{0} & \psi \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{Q}_\lambda^T \\ \mathbf{Q}_{M-\lambda}^T \end{bmatrix} \mathbf{x}_t \\ &= \mathbf{Q}_\lambda (\bar{\mathbf{G}}_{t,\lambda} \mathbf{Q}_\lambda^T \mathbf{x}_t - \psi \mathbf{Q}_{M-\lambda}^T \mathbf{x}_t) + \psi \mathbf{x}_t, \end{aligned} \quad (13)$$

where we define  $\bar{\mathbf{G}}_{t,\lambda} \in \mathbb{R}^{M \times \lambda}$  as the first  $\lambda$  columns of  $\bar{\mathbf{G}}_t$  [20]. Note from (13) that we do not need  $\mathbf{Q}_{M-\lambda}$  to compute  $\bar{\mathbf{\Omega}}_t \mathbf{x}_t$ . On the contrary, we need to compute the elements of  $\bar{\mathbf{G}}_{t,\lambda}$  (12) since  $\hat{\mathbf{w}}_t = \mathbf{Q}^T \mathbf{w}_t$ . However, we emphasize that only the first  $\lambda$  entries of  $\hat{\mathbf{w}}_t$ , i.e.,  $\hat{w}_{t,\lambda}$ , are needed since only  $\{\bar{\phi}_t^{(k)} : k=1, \dots, \lambda\}$  are computed in our computationally more efficient algorithm. Hence, we update the sub-vector  $\hat{\mathbf{w}}_{t,\lambda}$  as

$$\hat{\mathbf{w}}_{t+1,\lambda} = \hat{\mathbf{w}}_{t,\lambda} + 2\mu e_t^3 \frac{\bar{\mathbf{G}}_{t,\lambda} \mathbf{Q}_\lambda^T \mathbf{x}_t}{\mathbf{x}_t^T \bar{\mathbf{\Omega}}_t \mathbf{x}_t + \epsilon} \quad (14)$$

and the update for  $\mathbf{w}_t$  is given by

$$\mathbf{w}_{t+1} = \mathbf{w}_t + 2\mu e_t^2 \frac{\bar{\mathbf{\Omega}}_t \mathbf{x}_t}{\mathbf{x}_t^T \bar{\mathbf{\Omega}}_t \mathbf{x}_t + \epsilon}. \quad (15)$$

At each time  $t$  the sub-matrix  $\bar{\mathbf{G}}_{t,\lambda}$  is computed, and using (13) the sub-vector  $\hat{\mathbf{w}}_{t,\lambda}$  and the weight vector  $\mathbf{w}_t$  are updated as in (14) and (15), respectively. Note that the computational complexity of (13) is only  $O(\lambda M)$ , i.e.,  $O(M)$ , so are those of (14) and (15). Therefore, using this approach, given in (12)–(15), we can attain linear computational complexity per iteration.

Since  $\mathbf{Q}_{M-\lambda}$  is not used, in the new scheme we can compute only the first  $\lambda \ll M$  columns of  $\mathbf{Q}$  beforehand. In [20], the author suggests that  $\lambda \approx 5$  is enough to achieve acceptable performance in general. Additionally, we can choose the smallest  $\lambda$  satisfying that  $\hat{\mathbf{R}} \hat{\mathbf{p}}$  is within the subspace spanned by the first  $\lambda$  columns of  $K_M(\hat{\mathbf{R}}, \hat{\mathbf{p}})$  [20]. To this end, a threshold  $\delta=0.01$  yields reasonable performance in the selection of the smallest  $\lambda$  in general [20].

In the next section, we provide a complete performance analysis for the proposed algorithms.

## 6. Performance analysis

We can write the proportionate update based algorithms in the following form:

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \mu \frac{\Phi_t \mathbf{x}_t}{\mathbf{x}_t^T \Phi_t \mathbf{x}_t + \epsilon} f(e_t) \quad (16)$$

where  $\Phi_t$  denotes  $\bar{\mathbf{G}}_t$  for the PNLMF variant algorithms while  $\Phi_t$  corresponds to  $\bar{\mathbf{\Omega}}_t$  for the KPNLMF variant algorithms. We note that  $\Phi$  is a symmetric positive definite matrix for both of the cases. Additionally,  $f(e_t)$  is the error nonlinearity function, e.g.,  $f(e_t) = 2e_t^3$ .

We define *a priori* and the weighted *a priori* estimation error as follows:

$$e_{a,t} \triangleq \mathbf{x}_t^T (\mathbf{w}_o - \mathbf{w}_t) \quad \text{and} \quad e_{a,t}^\Sigma \triangleq \mathbf{x}_t^T \Sigma (\mathbf{w}_o - \mathbf{w}_t),$$

where  $\Sigma$  is a symmetric positive definite weighting matrix. We utilize the weighting matrix  $\Sigma$  later in the analysis. The deviation parameter vector is defined as  $\tilde{\mathbf{w}} = \mathbf{w}_o - \mathbf{w}_t$ . Then, the weighted energy recursion of (16) leads to

$$\begin{aligned} E[\|\tilde{\mathbf{w}}_{t+1}\|_{\Sigma}^2] &= E[\|\tilde{\mathbf{w}}_t\|_{\Sigma}^2] - 2\mu E\left[\mathbf{x}_t^T \left(\frac{\Phi_t \Sigma}{\mathbf{x}_t^T \Phi_t \mathbf{x}_t + \epsilon}\right) \tilde{\mathbf{w}} f(e_t)\right] \\ &\quad + \mu^2 E\left[\mathbf{x}_t^T \left(\frac{\Phi_t \Sigma \Phi_t}{\mathbf{x}_t^T \Phi_t \mathbf{x}_t + \epsilon}\right) \mathbf{x}_t f^2(e_t)\right], \\ &= E[\|\tilde{\mathbf{w}}_t\|_{\Sigma}^2] - 2\mu E[e_{a,t}^\Sigma f(e_t)] + \mu^2 E[\|\mathbf{x}_t\|_{\Sigma_2}^2 f^2(e_t)], \end{aligned} \quad (17)$$

where

$$\Sigma_1 \triangleq \frac{\Phi_t \Sigma}{\mathbf{x}_t^T \Phi_t \mathbf{x}_t + \epsilon} \quad \text{and} \quad \Sigma_2 \triangleq \frac{\Phi_t \Sigma \Phi_t}{\mathbf{x}_t^T \Phi_t \mathbf{x}_t + \epsilon}.$$

In the subsequent analysis of (17), we employ the following assumptions:

**Assumption 1.** The observation noise  $v_t$  is a zero-mean independently and identically distributed (i.i.d.) Gaussian random variable and independent from  $\mathbf{x}_t$ . The regressor signal  $\mathbf{x}_t$  is also zero-mean i.i.d. Gaussian random vector with the auto-correlation matrix  $\mathbf{R}_x \triangleq \sigma_x^2 \mathbf{I}$ .

**Assumption 2.** The *a priori* estimation error  $e_{a,t}$  has Gaussian distribution and it is jointly Gaussian with the weighted *a priori* estimation error  $e_{a,t}^\Sigma$ . The assumption is reasonable for long filters, i.e.,  $p$  is large, sufficiently small step size  $\mu$  and by Assumption 1 [36].

**Assumption 3.** The random variables  $\|\mathbf{x}_t\|_{\Sigma_2}^2$  and  $f^2(e_t)$  are uncorrelated, which enables the following split as

$$E[\|\mathbf{x}_t\|_{\Sigma_2}^2 f^2(e_t)] = E[\|\mathbf{x}_t\|_{\Sigma_2}^2] E[f^2(e_t)].$$

**Assumption 4.** The coefficients of the mean of the estimation vector  $\mathbf{w}_t$  are far larger than the corresponding variance such that the matrix  $\Phi_t$  and the deviation vector  $\tilde{\mathbf{w}}_t$  are uncorrelated and

$$E[e_{a,t}^\Sigma e_{a,t}] = E\left[\tilde{\mathbf{w}}_t^T E\left[\frac{\mathbf{x}_t \mathbf{x}_t^T \Phi_t \Sigma}{\mathbf{x}_t^T \Phi_t \mathbf{x}_t + \epsilon}\right] \tilde{\mathbf{w}}_t\right].$$

**Remark 6.1.** By Assumption 1, we can express the relation between the various performance measures, i.e., the mean-square deviation (MSD)  $E[\|\tilde{\mathbf{w}}_t\|^2]$  denoted by  $\xi$ , the excess mean square error (EMSE)  $E[e_{a,t}^2]$  denoted by  $\zeta$  and the mean square error (MSE)  $E[e_t^2] = \sigma_e^2$  as follows:

$$\sigma_e^2 = \zeta + \sigma_v^2 = \sigma_x^2 \xi + \sigma_v^2. \quad (18)$$

Hence, once we evaluate one of those performance measures, we can obtain the other results through (18).

We next provide the mean square convergence performance of the introduced algorithms.

### 6.1. Transient analysis

By **Assumptions 1 and 2**, and Price's result [37–39], we obtain

$$E[e^{\sum_i f(e_t)}] = E[e^{\sum_i e_{a,t}}] \frac{E[e_{a,t} f(e_{a,t} + v_t)]}{E[e_{a,t}^2]} \quad (19)$$

We can evaluate the first term on the right hand side of (19) through the generalized Abelian integral functions [40,41]. By **Assumption 4**, we replace  $\Phi_t$  with its mean  $\bar{\Phi}_t \triangleq E[\Phi_t]$  in

$$E \left[ \frac{\mathbf{x}_t \mathbf{x}_t^T \bar{\Phi}_t \Sigma}{\mathbf{x}_t^T \bar{\Phi}_t \mathbf{x}_t + \epsilon} \right] = E \left[ \frac{\mathbf{x}_t \mathbf{x}_t^T}{\mathbf{x}_t^T \bar{\Phi}_t \mathbf{x}_t + \epsilon} \right] \bar{\Phi}_t \Sigma.$$

Then, we have

$$E \left[ \frac{\mathbf{x}_t \mathbf{x}_t^T}{\mathbf{x}_t^T \bar{\Phi}_t \mathbf{x}_t + \epsilon} \right] = \frac{1}{(2\pi)^{M/2} \sigma_x^M} \int \dots \int \frac{\mathbf{x}_t \mathbf{x}_t^T}{\mathbf{x}_t^T \bar{\Phi}_t \mathbf{x}_t + \epsilon} \exp\left(-\frac{\mathbf{x}_t^T \bar{\Phi}_t \mathbf{x}_t}{2\sigma_x^2}\right) d\mathbf{x}_t. \quad (20)$$

In order to evaluate (20), as in [41], we define

$$F(\beta) \triangleq \frac{1}{(2\pi)^{M/2} \sigma_x^M} \int \dots \int \frac{\mathbf{x}_t \mathbf{x}_t^T e^{-\beta(\epsilon + \mathbf{x}_t^T \bar{\Phi}_t \mathbf{x}_t)}}{\mathbf{x}_t^T \bar{\Phi}_t \mathbf{x}_t + \epsilon} e^{-\mathbf{x}_t^T \bar{\Phi}_t \mathbf{x}_t / 2\sigma_x^2} d\mathbf{x}_t$$

and the derivative of  $F(\beta)$  with respect to  $\beta$  yields

$$\frac{dF(\beta)}{d\beta} = -\frac{e^{-\beta\epsilon}}{(2\pi)^{M/2} \sigma_x^M} \int \dots \int \mathbf{x}_t \mathbf{x}_t^T e^{-(1/2)\mathbf{x}_t^T \bar{\Phi}_t^{-1} \mathbf{x}_t} d\mathbf{x}_t, \quad (21)$$

where

$$\mathbf{B}_t \triangleq \left( \frac{1}{\sigma_x^2} + 2\beta \bar{\Phi}_t \right)^{-1}.$$

Then, after some algebra we obtain (21) as

$$\frac{dF(\beta)}{d\beta} = \mathbf{B}_t \frac{e^{-\beta\epsilon} |\mathbf{B}_t|^{1/2}}{\sigma_x^M}, \quad (22)$$

where  $|\mathbf{B}_t|$  denotes the determinant of  $\mathbf{B}_t$ .

We point out that  $\Phi_t = \mathbf{G}_t$  has a diagonal structure, however,  $\Phi_t = \Omega_t = \mathbf{Q} \hat{\mathbf{G}}_t \mathbf{Q}^T$  may not necessarily be diagonal. Hence, consider that the eigenvalue decomposition of  $\Phi_t = \mathbf{U} \Lambda_t \mathbf{U}^T$  where  $\Lambda_t = \text{diag}\{\lambda_t^{(1)}, \dots, \lambda_t^{(M)}\}$  so that we can write  $\mathbf{B}_t = \mathbf{U} \mathbf{D}_t \mathbf{U}^T$  where

$$\mathbf{D}_t = \left( \frac{1}{\sigma_x^2} + 2\beta \Lambda_t \right)$$

Then, we obtain

$$|\mathbf{B}_t| = \prod_{l=1}^M \left( \frac{1}{\sigma_x^2} + 2\beta \lambda_t^{(l)} \right)^{-1}. \quad (23)$$

Since  $F(0)$  yields (20), through (22) and (23), we get

$$E \left[ \frac{\mathbf{x}_t \mathbf{x}_t^T}{\mathbf{x}_t^T \bar{\Phi}_t \mathbf{x}_t + \epsilon} \right] = \mathbf{U} \mathbf{D}_\Lambda \mathbf{U}^T \sigma_x^2, \quad (24)$$

where  $\mathbf{D}_\Lambda = \text{diag}\{I_1(\Lambda), \dots, I_M(\Lambda)\}$  and

$$I_k(\Lambda) = \int_0^\infty e^{-\beta\epsilon} \prod_{l=1}^M (1 + 2\beta \lambda^{(l)})^{-1/2} (1 + 2\beta \lambda^{(k)})^{-1} d\beta,$$

which is in the form of a generalized Abelian integral function and can be evaluated numerically. Note that we can

approximate  $\lambda^{(k)}$  as

$$\lambda^{(k)} = \frac{1 - \gamma}{M} + \gamma \frac{|W_o^{(k)}|}{\|\mathbf{w}_o\|_1 + \kappa}$$

or

$$\lambda^{(k)} = \frac{1 - \gamma}{M} + \gamma \frac{|W_o^{*(k)}|}{\|\mathbf{w}_o^*\|_1 + \kappa}$$

for the PNLMF and the KPNLMF algorithms, respectively.

Next, we evaluate the second term on the right hand side of (17). To this end, we define

$$\mathbf{A} \triangleq E \left[ \frac{\mathbf{x}_t \mathbf{x}_t^T \bar{\Phi}_t \Sigma}{\mathbf{x}_t^T \bar{\Phi}_t \mathbf{x}_t + \epsilon} \right] = \sigma_x^2 \mathbf{U} \mathbf{D}_\Lambda \mathbf{U}^T \bar{\Phi}_t \Sigma.$$

Taking derivative of  $\mathbf{A}$  with respect to  $\epsilon$ , we get

$$\begin{aligned} \frac{\partial \mathbf{A}}{\partial \epsilon} &= -E \left[ \frac{\mathbf{x}_t \mathbf{x}_t^T \bar{\Phi}_t \Sigma}{(\mathbf{x}_t^T \bar{\Phi}_t \mathbf{x}_t + \epsilon)^2} \right] \\ &= -\sigma_x^2 \mathbf{U} \tilde{\mathbf{D}}_\Lambda \mathbf{U}^T \bar{\Phi}_t \Sigma, \end{aligned}$$

where  $\tilde{\mathbf{D}}_\Lambda \triangleq \text{diag}\{\tilde{I}_1(\Lambda), \dots, \tilde{I}_M(\Lambda)\}$  and

$$\tilde{I}_k(\Lambda) = \int_0^\infty \beta e^{-\beta\epsilon} \prod_{l=1}^M (1 + 2\beta \lambda^{(l)})^{-1/2} (1 + 2\beta \lambda^{(k)})^{-1} d\beta.$$

We point out that

$$\begin{aligned} E[\|\mathbf{x}_t\|_{\Sigma_2}^2] &= -\text{Tr} \left\{ \frac{\partial \mathbf{A}}{\partial \epsilon} \bar{\Phi}_t \right\} \\ &= \sigma_x^2 \text{Tr} \left\{ \mathbf{U} \tilde{\mathbf{D}}_\Lambda \mathbf{U}^T \bar{\Phi}_t \Sigma \bar{\Phi}_t \right\}. \end{aligned} \quad (25)$$

By (24) and (25), the weighted energy recursion (17) yields

$$\begin{aligned} E[\|\tilde{\mathbf{w}}_{t+1}\|_{\Sigma}^2] &= E[\|\tilde{\mathbf{w}}_t\|_{\Sigma}^2] - 2\mu \sigma_x^2 E[\|\tilde{\mathbf{w}}_t\|_{\Sigma}^2] \frac{E[e_{a,t} f(e_t)]}{E[e_{a,t}^2]} \\ &\quad + \mu^2 \sigma_x^2 \text{Tr} \left\{ \tilde{\mathbf{Y}} \Sigma \bar{\Phi}_t \right\} E[f^2(e_t)], \end{aligned} \quad (26)$$

$\underbrace{h_G(e_{a,t}, v_t)}_{h_U(e_{a,t}, v_t)}$

where  $\mathbf{Y} \triangleq \mathbf{U} \mathbf{D}_\Lambda \mathbf{U}^T \bar{\Phi}_t$  and  $\tilde{\mathbf{Y}} \triangleq \mathbf{U} \tilde{\mathbf{D}}_\Lambda \mathbf{U}^T \bar{\Phi}_t$ . In **Table 1**, we tabulate  $h_G(e_{a,t}, v_t)$  and  $h_U(e_{a,t}, v_t)$  for the mean-square, mean-fourth and mixture norm updates [36]. We note that by **Assumption 1**, we have  $\sigma_{e_a}^2 = \sigma_x^2 E[\|\tilde{\mathbf{w}}_t\|^2]$ .

We point out that by the Cayley–Hamilton theorem, we can write

$$\mathbf{Y}^M = -c_0 \mathbf{I} - c_1 \mathbf{Y} - \dots - c_{M-1} \mathbf{Y}^{M-1},$$

where  $c_i$ 's are the coefficients of the characteristic polynomial of  $\mathbf{Y}$  as follows:

$$\det(y\mathbf{I} - \mathbf{Y}) = y^M + c_{M-1} y^{M-1} + \dots + c_1 y + c_0.$$

Hence, the transient behavior of the proportionate update based algorithms is given by the following theorem.

**Theorem 1.** Consider a proportionate update based algorithm with the error nonlinearity function  $f(e_t)$ . Then, assuming the adaptive filter is mean-square stable and through **Assumptions 1–4**, the mean-square convergence behavior of the filter is characterized by the state space recursion

$$\mathcal{W}_{t+1} = \mathcal{A}_t \mathcal{W}_t + \mu^2 \sigma_x^2 \mathcal{Y}_t$$

**Table 1**  
 $h_G(e_t)$  and  $h_U(e_t)$  functions in terms of  $\sigma_{e_a}^2$  and  $\sigma_v^2$ .

$f(e_t)$	$h_G(e_{a,t}, v_t)$	$h_U(e_{a,t}, v_t)$
$e_t$	1	$\sigma_{e_a}^2 + \sigma_v^2$
$2e_t^2$	$6(\sigma_{e_a}^2 + \sigma_v^2)$	$60(\sigma_{e_a}^2 + \sigma_v^2)^3$
$\delta e_t + 2(1-\delta)e_t^3$	$\delta + 6(1-\delta)(\sigma_{e_a}^2 + \sigma_v^2)$	$\delta^2(\sigma_{e_a}^2 + \sigma_v^2) + 12\delta(1-\delta)(\sigma_{e_a}^2 + \sigma_v^2)^2 + 60(1-\delta)^2(\sigma_{e_a}^2 + \sigma_v^2)^3$

where the state vectors are defined as

$$\mathcal{W}_t \triangleq \begin{bmatrix} E[\|\tilde{\mathbf{w}}_t\|^2] \\ \vdots \\ E[\|\tilde{\mathbf{w}}_t\|_{\mathbf{Y}^{M-1}}^2] \end{bmatrix}, \quad \mathcal{Y}_t \triangleq h_U(e_{a,t}, v_t) \begin{bmatrix} \text{Tr}\{\tilde{\mathbf{Y}}\bar{\Phi}_t\} \\ \vdots \\ \text{Tr}\{\tilde{\mathbf{Y}}\mathbf{Y}^{M-1}\bar{\Phi}_t\} \end{bmatrix}$$

and the coefficient matrix  $\mathcal{A}_t$  is given by

$$\mathcal{A}_t \triangleq \begin{bmatrix} 1 & -2\mu\sigma_x^2 h_G & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 2\mu c_0 \sigma_x^2 h_G & 2\mu c_1 \sigma_x^2 h_G & \cdots & 1 + 2\mu c_{M-1} \sigma_x^2 h_G \end{bmatrix}.$$

Note that we have removed the argument of  $h_G(e_{a,t}, v_t)$  for notational simplicity.

In the sequel, we analyze the steady-state behavior of the algorithms.

## 6.2. Steady-state analysis

In the steady-state we assume that

$$\lim_{t \rightarrow \infty} E[\|\mathbf{w}_{t+1}\|_{\Sigma}^2] = \lim_{t \rightarrow \infty} E[\|\mathbf{w}_t\|_{\Sigma}^2].$$

Then, by (26), at steady state we have

$$E[\|\tilde{\mathbf{w}}_t\|_{\mathbf{Y}\Sigma}^2] = \frac{\mu}{2} \text{Tr}\{\tilde{\mathbf{Y}}\Sigma\bar{\Phi}_t\} \frac{h_U(e_{a,t}, v_t)}{h_G(e_{a,t}, v_t)}. \quad (27)$$

Since  $\mathbf{Y}$  is a positive definite matrix, the steady state mean square deviation (MSD) yields

$$\begin{aligned} \xi &\triangleq \lim_{t \rightarrow \infty} E[\|\tilde{\mathbf{w}}_t\|^2], \\ &= \frac{\mu}{2} \text{Tr}\{\tilde{\mathbf{Y}}\mathbf{Y}^{-1}\bar{\Phi}_t\} \frac{h_U(e_{a,t}, v_t)}{h_G(e_{a,t}, v_t)}. \end{aligned}$$

Then, the steady-state behavior of the proportionate update based algorithms is given by the following theorem.

**Theorem 2.** Consider the same setting of Theorem 1. Then, the steady-state MSD denoted by  $\xi$  of the adaptive filter satisfies

$$\xi = \frac{\mu}{2} \text{Tr}\{\bar{\mathbf{Y}}\} \frac{h_U(e_{a,t}, v_t)}{h_G(e_{a,t}, v_t)}, \quad (28)$$

where  $\bar{\mathbf{Y}} \triangleq \mathbf{U}\bar{\mathbf{D}}_{\Lambda}\mathbf{A}\mathbf{U}^T$  and  $\bar{\mathbf{D}}_{\Lambda} \triangleq \bar{\mathbf{D}}_{\Lambda}\mathbf{D}_{\Lambda}^{-1} = \text{diag}\{\bar{l}_1(\Lambda), \dots, \bar{l}_M(\Lambda)\}$  and

$$\bar{l}_k(\Lambda) = \frac{\int_0^{\infty} \beta e^{-\beta e} \prod_{l=1}^M (1+2\beta\lambda^{(l)})^{-1/2} (1+2\beta\lambda^{(k)})^{-1} d\beta}{\int_0^{\infty} e^{-\beta e} \prod_{l=1}^M (1+2\beta\lambda^{(l)})^{-1/2} (1+2\beta\lambda^{(k)})^{-1} d\beta}.$$

Through (28), we can calculate the steady-state MSD of the introduced algorithms exactly. Then, the steady-state

MSD of the proportionate update based algorithms with mean-square error objective, i.e., the IPNLMS and KPNLMS algorithms, is given by

$$\xi_s = \frac{\mu\sigma_v^2 \text{Tr}\{\bar{\mathbf{Y}}\}}{2 - \mu\sigma_x^2 \text{Tr}\{\bar{\mathbf{Y}}\}}. \quad (29)$$

In addition, the steady-state MSD for the mean-fourth error objective, i.e., the PNLMF and KPNLMF algorithms, is found as

$$\xi_f = \frac{1 - 10\mu\sigma_x^2\sigma_v^2 \text{Tr}\{\bar{\mathbf{Y}}\} \pm \sqrt{1 - 20\mu\sigma_x^2\sigma_v^2 \text{Tr}\{\bar{\mathbf{Y}}\}}}{10\mu\sigma_x^4 \text{Tr}\{\bar{\mathbf{Y}}\}},$$

where the smaller root coincides with the ensemble averaged results. Furthermore, in the following, we provide the steady-state MSD of the mixed-norm algorithms under the assumption that the estimation error gets so small that we can neglect the relatively high order error terms. Since  $\sigma_{e_a}^2 = \sigma_x^2 \xi$ , (28) for mixed-norm error objective yields

$$\xi_m = \frac{\mu\sigma_v^2 \delta \text{Tr}\{\bar{\mathbf{Y}}\} (\delta + 12(1-\delta)\sigma_v^2)}{2\delta + 12(1-\delta)\sigma_v^2 - \mu\sigma_x^2 \text{Tr}\{\bar{\mathbf{Y}}\} \delta_o}, \quad (30)$$

where  $\delta_o \triangleq \delta^2 + 24\delta(1-\delta)\sigma_v^2 + 180(1-\delta)^2\sigma_v^4$ . We note that for  $\delta=1$ , (30) coincides with (29).

**Remark 6.2.** We note that for the stable-PNLMF and the stable-KPNLMF algorithms, we have

$$h_G(e_{a,t}, v_t) = \frac{1}{E[e_{a,t}^2]} E \left[ e_{a,t} \frac{2e_t^3}{\mathbf{x}_t^T \Phi_t \mathbf{x}_t + e_t^2} \right]$$

and

$$h_U(e_{a,t}, v_t) = E \left[ \frac{4e_t^6}{(\mathbf{x}_t^T \Phi_t \mathbf{x}_t + e_t^2)^2} \right].$$

We assume that the estimation error  $e_t$  gets relatively small at the steady state such that

$$f(e_t) = \frac{2e_t^3}{\mathbf{x}_t^T \Phi_t \mathbf{x}_t + e_t^2} \rightarrow \frac{2e_t^3}{\mathbf{x}_t^T \Phi_t \mathbf{x}_t}$$

and similarly

$$f^2(e_t) = \frac{4e_t^6}{(\mathbf{x}_t^T \Phi_t \mathbf{x}_t + e_t^2)^2} \rightarrow \frac{4e_t^6}{(\mathbf{x}_t^T \Phi_t \mathbf{x}_t)^2}$$

as  $t \rightarrow \infty$ . Then, by [Assumption 3](#), at the steady-state, for the proposed stable algorithms we obtain

$$\xi_s = \frac{\mu}{2} \text{Tr}\{\bar{\mathbf{Y}}\} \underbrace{\frac{E[4e_t^6]E[e_{a,t}^2]}{E[2e_{a,t}e_t^3]}}_{\text{braced term}} \frac{E\left[\frac{1}{(\mathbf{x}_t^T \Phi_t \mathbf{x}_t)^2}\right]}{E\left[\frac{1}{\mathbf{x}_t^T \Phi_t \mathbf{x}_t}\right]}. \quad (31)$$

We point out that with the involvement of the braced term only on the right hand side of (31), (31) yields the steady-state MSD of the algorithms with the mean-fourth error cost function. Hence, the steady-state performance of the proposed stable algorithms might differ from the steady-state performance of the conventional least mean fourth algorithms based on the statistics of the regressor signal.

Additionally, at the initial stages of the adaptation where the estimation error is relatively large, the error nonlinearity is approximately given by

$$f(e_t) = \frac{2e_t^3}{\mathbf{x}_t^T \Phi_t \mathbf{x}_t + e_t^2} \approx 2e_t$$

implying that the proposed stable algorithms demonstrate similar learning rate with the least mean square algorithms in the transient stage.

**Remark 6.3.** We note that a mixture of the mean square and the mean fourth error cost functions outperforms both of the constituent filters [22,42]. In [42], the authors show that the optimum error nonlinearity for the adaptive filters without data normalization is an optimal mixture of different order of error measures. Hence, a mixture of the mean-square error and the mean-fourth error objectives can better approximate the optimum error nonlinearity also for the proportionate update algorithms. At the steady-state by (27) and setting  $\Sigma = \sigma_x^2 \mathbf{Y}_t^{-1}$ , we obtain

$$\zeta = \frac{\mu}{2} \sigma_x^2 \text{Tr}\{\bar{\mathbf{Y}}\} \frac{E[f^2(e_t)]E[e_{a,t}^2]}{E[e_{a,t}f(e_t)]}, \quad (32)$$

where  $\zeta = \lim_{t \rightarrow \infty} E[e_{a,t}^2]$  denotes the steady-state excess mean square error. Then, through [Assumptions 1 and 2](#), and Price's result [42], we get

$$E[e_{a,t}f(e_t)] = E[e_{a,t}^2]E[f'(e_t)],$$

where  $f'(e_t)$  is the derivative of  $f(e_t)$  with respect to  $e_t$ . Then, (32) yields

$$\zeta = \frac{\mu}{2} \sigma_x^2 \text{Tr}\{\bar{\mathbf{Y}}\} \frac{E[f^2(e_t)]}{E[f'(e_t)]}. \quad (33)$$

However, the excess mean square error is lower bounded by the Cramer–Rao lower bound denoted by  $C$  [43]. Hence, (33) leads to

$$\frac{E[f^2(e_t)]}{E[f'(e_t)]} \geq \frac{2C}{\underbrace{\mu \sigma_x^2 \text{Tr}\{\bar{\mathbf{Y}}\}}_{\alpha}}$$

and with equality for

$$f(e_t) = -\alpha \frac{p'_e(e_t)}{p_e(e_t)}, \quad (34)$$

where  $p_e(e_t)$  is the probability density function of the estimation error  $e_t$  [42]. For a given error distribution, we

can derive the optimum error nonlinearity through (34). Additionally, after some algebra, through the Edgeworth expansion of the distribution, we obtain

$$f_{\text{opt}}(e_t) = \sum_{j=0}^{\infty} c_{2j+1} e_t^{2j+1},$$

where  $c_{2j+1}$ 's are the combination weights. Hence, we emphasize that through the mixture of mean-square and the mean-fourth error objectives we can approximate the optimum error nonlinearity better than the constituent filters.

In the next subsection, we analyze the tracking performance of the introduced algorithms in a non-stationary environment.

### 6.3. Tracking performance

We model the non-stationary system through a first-order random walk model, in which the parameter vector of the unknown system changes in time as follows:

$$\mathbf{w}_{\mathbf{o}t+1} = \mathbf{w}_{\mathbf{o}t} + \mathbf{q}_t, \quad (35)$$

where  $\mathbf{q}_t \in \mathbb{R}^M$  is a zero-mean vector process which is independent of the regressor  $\mathbf{x}_t$  and the noise  $v_t$  and has a covariance matrix  $\mathbf{C} = E[\mathbf{q}_t \mathbf{q}_t^T]$ . Since the definitions of *a priori* estimation error does not change under the first-order random walk model, the new weighted energy recursion is given by

$$E[\|\tilde{\mathbf{w}}_{t+1}\|_{\Sigma}^2] = E[\|\tilde{\mathbf{w}}_t\|_{\Sigma}^2] - 2\mu\sigma_x^2 E[\|\tilde{\mathbf{w}}_t\|_{\Sigma}^2] h_G(e_{a,t}, v_t) + \mu^2 \sigma_x^2 \text{Tr}\{\tilde{\mathbf{Y}} \Sigma \bar{\Phi}_t\} h_U(e_{a,t}, v_t) + E[\mathbf{q}_t^T \Sigma \mathbf{q}_t].$$

Then, at steady-state we have

$$E[\|\tilde{\mathbf{w}}_t\|_{\Sigma}^2] = \frac{\mu\sigma_x^2 \text{Tr}\{\tilde{\mathbf{Y}} \Sigma \bar{\Phi}_t\} h_U(e_{a,t}, v_t) + \mu^{-1} \text{Tr}\{\mathbf{C}\Sigma\}}{2\sigma_x^2 h_G(e_{a,t}, v_t)}.$$

Hence, we obtain the following theorem.

**Theorem 3.** Consider the same setting of [Theorems 1 and 2](#) in a non-stationary environment modeled with the first-order random walk model through (35). Then, at the steady-state the following equality holds:

$$\xi' = \frac{\mu\sigma_x^2 \text{Tr}\{\bar{\mathbf{Y}}\} h_U(e_{a,t}, v_t) + \mu^{-1} \text{Tr}\{\mathbf{C}\mathbf{Y}^{-1}\}}{2\sigma_x^2 h_G(e_{a,t}, v_t)}, \quad (36)$$

where  $\xi'$  is the steady-state MSD of the algorithm.

By (36), the steady state MSD in the non-stationary environment for  $f(e_t) = e_t$  leads to

$$\xi'_s = \frac{\mu\sigma_v^2 \text{Tr}\{\bar{\mathbf{Y}}\} + \mu^{-1} \sigma_x^{-2} \text{Tr}\{\mathbf{C}\mathbf{Y}^{-1}\}}{2 - \mu\sigma_x^2 \text{Tr}\{\bar{\mathbf{Y}}\}}.$$

Correspondingly, the tracking performance of the mean-fourth error objective is roughly given by

$$\xi'_f \approx \frac{\text{Tr}\{\mathbf{C}\mathbf{Y}^{-1}\}}{12\mu\sigma_x^2 \sigma_v^2 - 180\mu\sigma_x^2 \sigma_v^4 \text{Tr}\{\bar{\mathbf{Y}}\}}.$$

Assuming the higher order measure of the estimation error is negligibly small at the steady-state, we obtain the

steady-state MSD for  $f(e_t) = \delta e_t + 2(1 - \delta)e_t^3$  as

$$\xi_m^e = \frac{\mu \sigma_v^2 \delta \text{Tr}\{\bar{\mathbf{Y}}\}(\delta + 12(1 - \delta)\sigma_v^2) + \mu^{-1} \sigma_x^{-2} \text{Tr}\{\mathbf{C}\mathbf{Y}^{-1}\}}{2\delta + 12(1 - \delta)\sigma_v^2 - \mu \sigma_x^2 \text{Tr}\{\bar{\mathbf{Y}}\}\delta_o}$$

**Remark 6.4.** We point out that under the first order random walk model, since the system impulse response, i.e.,  $\mathbf{w}_{o,t}$ , changes in time, the system statistics, e.g.,  $\mathbf{p}$ , changes even if  $\mathbf{q}_t$  is a zero mean vector process independent of the regression signal. Hence, the performance of the Krylov-proportionate update based algorithms degrades in the non-stationary environments. However, for the sufficiently slow change in the environment, the Krylov-proportionate algorithms can provide good tracking performance [20]. In Section 7, we substantiate this by several different numerical examples.

In the next section, we provide numerical examples comparing the convergence performance of the proposed algorithms in several different configurations.

## 7. Numerical examples

In this section, we examine the mean-square convergence performance of the proposed algorithms in various examples. In the first experimental setup, we consider an echo cancellation scenario. We observe a near-end signal  $d_t$  with a linear model such that

$$d_t = \mathbf{w}_o^T \mathbf{x}_t + v_t,$$

where  $\mathbf{x}_t \in \mathbb{R}^M$  denotes the far-end signal,  $v_t \in \mathbb{R}$  is a white Gaussian noise signal and  $\mathbf{w}_o \in \mathbb{R}^M$  represents the unknown echo path. We choose an artificial male voice sample<sup>1</sup> having the average characteristics of comprehensive human voice with a smaller variability relative to the real speech samples [44]. We generate the normalized echo path seen in Fig. 2 based on the Allen and Berkley's image source method for small-room acoustics [45,46]. The reflection coefficient of each wall is 0.7 and the room dimensions are  $4 \times 4 \times 2.5$  in metres. The sink, i.e., the microphone, locates at (2 m, 2 m, 1 m). In order to examine the performance of the algorithms against abrupt changes in the echo path, the source locates at (2 m, 1 m, 2 m) for  $t$  in [0, 5 s) and at (2 m, 2 m, 2 m) in [5 s, 10 s]. Correspondingly, we have sparse echo paths  $\mathbf{w}_o$  of length  $M=256$  and we use the same length for the adaptive filter  $\mathbf{w}_t$ . The sampling rate<sup>2</sup> is 8 kHz and noise variance is  $\sigma_v^2 = 10^{-3}$ . We measure the convergence rate of the algorithms in terms of thenormalized misalignment defined as  $\|\mathbf{w}_o - \mathbf{w}_t\|^2 / \|\mathbf{w}_o\|^2$ . In Fig. 3, we compare the time evolution of the system mismatch of the PNLMS, the IPNLMS, the sparse-NLMF introduced in [47], and the regularized stable-PNLMF algorithms for  $\mu_{\text{PNLMS}} = \mu_{\text{IPNLMS}} = \mu_{\text{sNLMF}} = \mu_{\text{sPNLMF}} = 0.1$ ,  $\gamma = 0.5$ , and  $\kappa = 10^{-4}$ . For the PNLMS algorithm, we set  $\delta = 0.1$  and  $\rho = 0.2$ . We note that we use a regularized version of the sparse-NLMF and set the threshold as 10 and  $\lambda = 10^{-6}$ , as suggested in [47]. Due to stability concerns, the

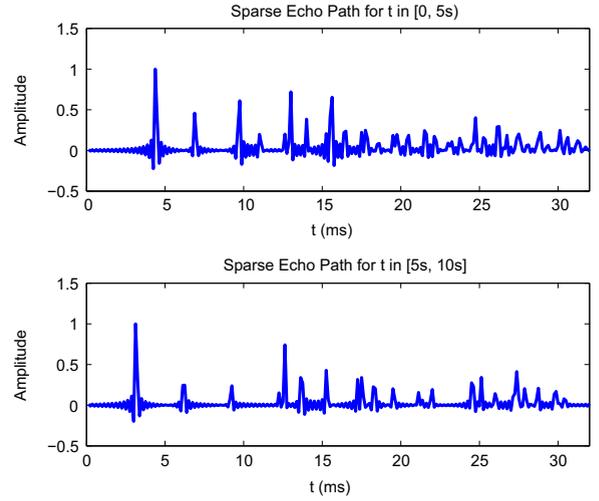


Fig. 2. Unknown echo paths.

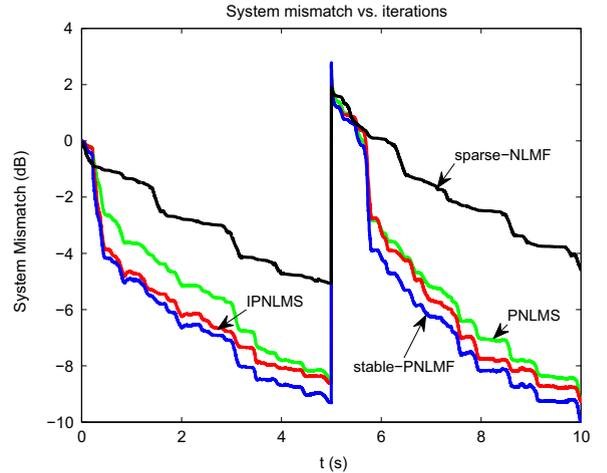


Fig. 3. Time evolution of the system mismatch in the realistic echo cancellation scenario.

regularization constants are chosen as  $\epsilon_{\text{PNLMS}} = 10^{-1}$ ,  $\epsilon_{\text{IPNLMS}} = \epsilon_{\text{sPNLMF}} = 10^{-3}$ , and  $\epsilon_{\text{sNLMF}} = 10^{-2}$ . Additionally, in Fig. 4, for  $t$  in [0, 1.5 s], we compare the echo return loss enhancement (ERLE), which is defined as [48]

$$\text{ERLE} = 10 \log_{10} \left( \frac{E[d_t^2]}{E[e_t^2]} \right).$$

For presentation purposes, we filter the ERLE curves with a moving average of length 1000. Through Figs. 3 and 4, we point out that the stable-PNLMF algorithm achieves enhanced performance relative to the other algorithms.

In the second example, we examine the performance of the algorithms for a dispersive system  $\mathbf{w}_o \in \mathbb{R}^{256}$  whose coefficients are chosen from a normal distribution. In Fig. 5, we plot the first 25 out of 256 coefficients of the system. We use simplified  $O(M)$  versions, introduced in [20], of the KPNLMS and the KPNLMF algorithms. The SNR is 30 dB. We set  $\mu = 0.1$ ,  $\gamma = 0.5$ ,  $\kappa = 10^{-5}$ ,  $\zeta = 0.001$ ,  $T_o = 2M$  and  $c = 0.065$ . We choose the threshold as  $\delta = 10^{-4}$  and we

<sup>1</sup> Artificial male voice sample [44] which can be found at <http://www.itu.int/net/itu-t/sigdb/genaudio/Pseries.htm>.

<sup>2</sup> We resample the input voice sample, which is originally 16 kHz.

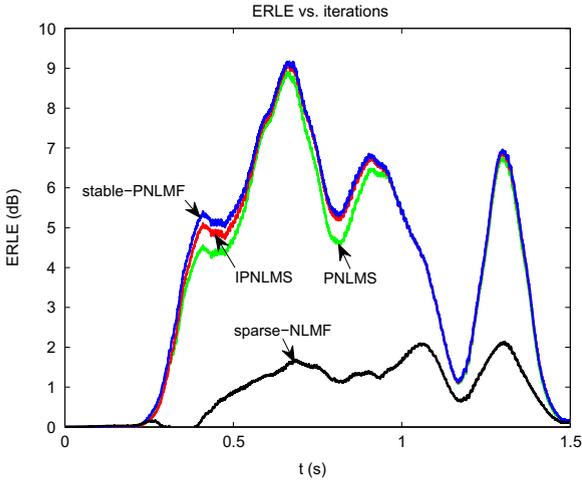


Fig. 4. Time evolution of the ERLE.

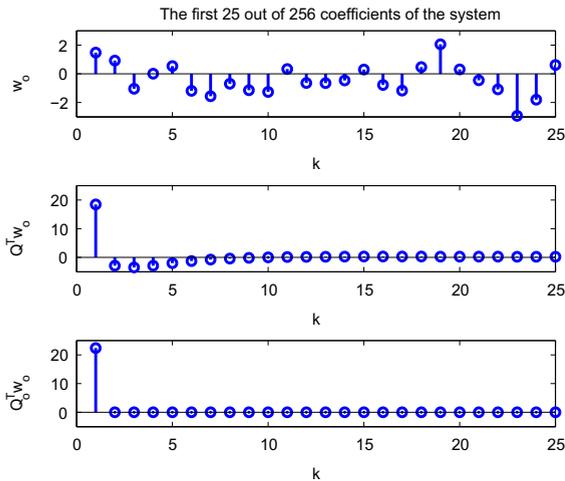


Fig. 5. After the projection of the dispersive system  $\mathbf{w}_0$  onto the Krylov subspace, we obtain sparse  $\mathbf{Q}^T \mathbf{w}_0$  and  $\mathbf{Q}_0^T \mathbf{w}_0$  generated through the estimated parameters  $\hat{\mathbf{R}}$  and  $\hat{\mathbf{p}}$ , and the true statistics, respectively.

select the smallest  $\lambda$  satisfying that threshold, i.e.,  $\lambda=7$ . In Fig. 5, we demonstrate the sparse structure of the system constructed by the projection of the true system onto the Krylov-subspace. Hence, we can employ proportionate update approach in the dispersive systems through the Krylov-subspace projection techniques. In Fig. 6, we compare the time evolution of the MSD of the KPNLMS, the IPNLMS, and the proposed stable-PNLMF and the stable-KPNLMF algorithms. As in Fig. 3, we can achieve better trade-off in terms of the transient and the steady-state performances through the stable-KPNLMF algorithm while avoiding the stability issues induced due to the mean-fourth error framework. In Fig. 7, we evaluate the performance of the proposed simplification scheme for different  $\lambda$  values. We observe that the proposed simplification scheme demonstrates almost identical performance with the simplification scheme introduced in [20]. Additionally, the learning rate and the computational complexity of the algorithm increases with the  $\lambda$  values. However, note that  $\lambda=10$  and  $\lambda=256$  (full dimension)

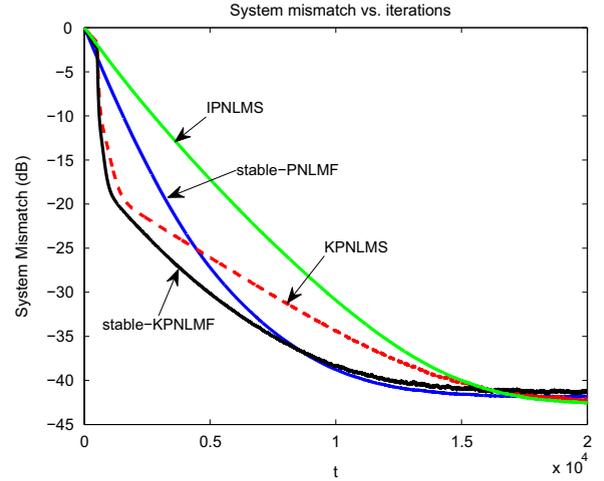


Fig. 6. Time evolution of the system mismatch of the KPNLMS, the proposed stable-PNLMF and stable-KPNLMF algorithms in a dispersive system.

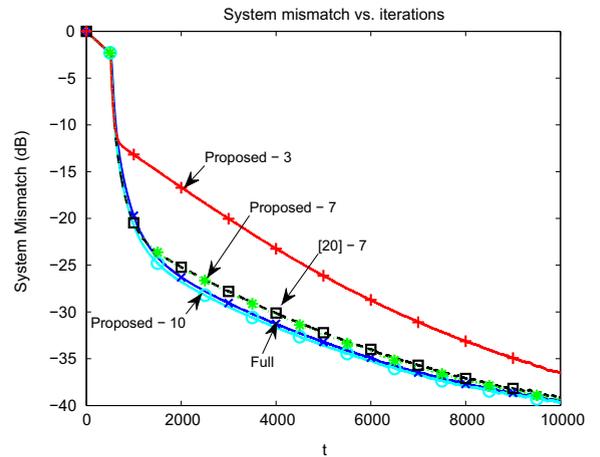


Fig. 7. Comparison of the proposed simplification scheme for the computational complexity with different  $\lambda$  choices in terms of the system mismatch.

achieve similar convergence performance while the computational complexities are  $O(\lambda M)$  and  $O(M^2)$ , respectively.

In the third example, we examine the performance of the proposed KPNLMMN algorithm with respect to the KPNLMS and KPNLMF algorithms. Different from the example 2, we set  $M=10$ ,  $\mu=0.05$ ,  $\epsilon=0.41$ , and SNR is 10 dB. The regularization constant  $\epsilon$  is determined according to [32]. We choose the unknown system coefficients randomly from a normal distribution and normalize it. For the threshold  $\delta=10^{-4}$ , the smallest  $\lambda$  satisfying that threshold is  $\lambda=4$ . Note that we compare the performance with the KPNLMF algorithm and in order to avoid stability issues we have set relatively short filter length instead of far smaller step sizes resulting longer convergence duration. In Fig. 8, we plot the time evolution of the MSD of the KPNLMMN algorithm where the combination weight  $\delta=0.5$  with the KPNLMS and the KPNLMF algorithms. We observe that the KPNLMMN algorithm has smaller steady-state MSD than the KPNLMF algorithm for

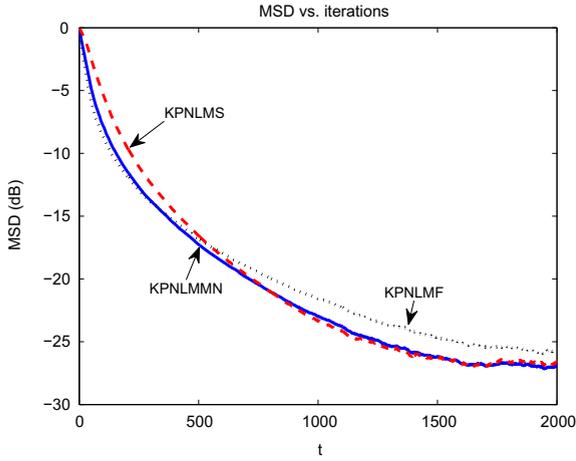


Fig. 8. Time evolution of the MSD of the KPNLMS, the KPNLMF and the proposed KPNLMMN ( $\delta = 0.5$ ) algorithms in a dispersive system.

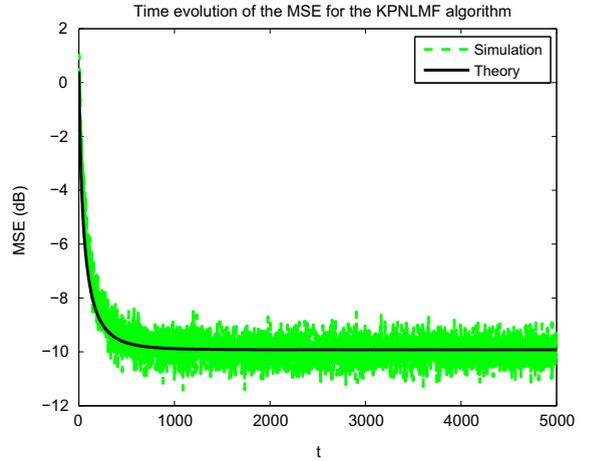


Fig. 10. Time evolution of the MSE of the KPNLMF algorithm.

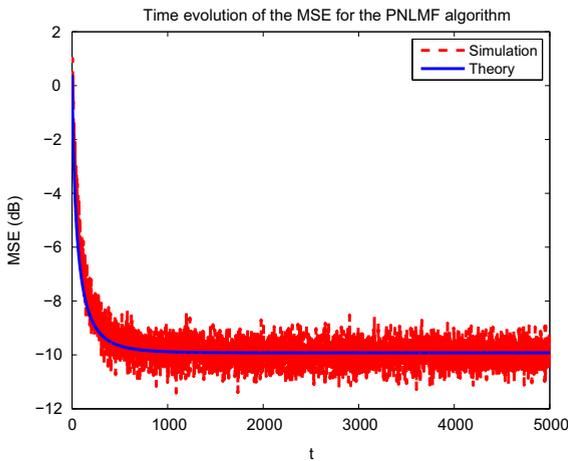


Fig. 9. Time evolution of the MSE of the PNLMF algorithm.

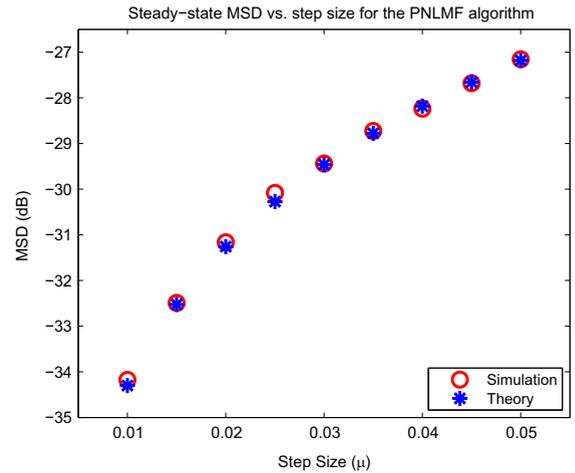


Fig. 11. Dependence of the steady-state MSD on the step size  $\mu$  for the PNLMF algorithm.

similar convergence rate and has a faster convergence rate with respect to the KPNLMS algorithm for the same steady-state MSD. Hence through the mixture norm approach, we can achieve superior performance relative to the constituent filters in the proportionate update based algorithms.

Finally, for the same system configuration with the example 3, we demonstrate that the theoretical results and the ensemble averaged simulation results match. In Figs. 9 and 10, we plot the time evolution of the MSE of the PNLMF and the KPNLMF algorithms respectively. In addition, in Figs. 11 and 12, we plot the steady-state MSD versus the adaptation step size for the PNLMF and the KPNLMF algorithms, respectively. We note that we use the system statistics, i.e.,  $\mathbf{R}$  and  $\mathbf{p}$ , directly. In Figs. 9–12, we observe that the theoretical and ensemble averaged results match. Furthermore, we evaluate the steady-state performance of the KPNLMMN algorithm in a non-stationary environment with the first-order random walk model. We choose  $\mathbf{q}_t$  randomly satisfying  $\|\mathbf{q}_t\|_1 \cong 10^{-4}$  for relatively slow change of the system impulse response. In Fig. 13, we observe that the theoretical results accurately match with

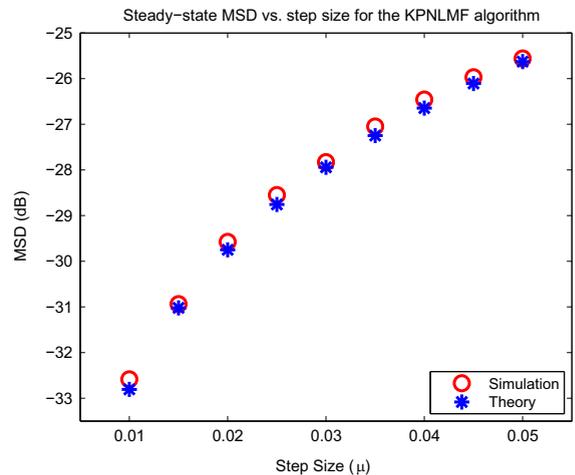
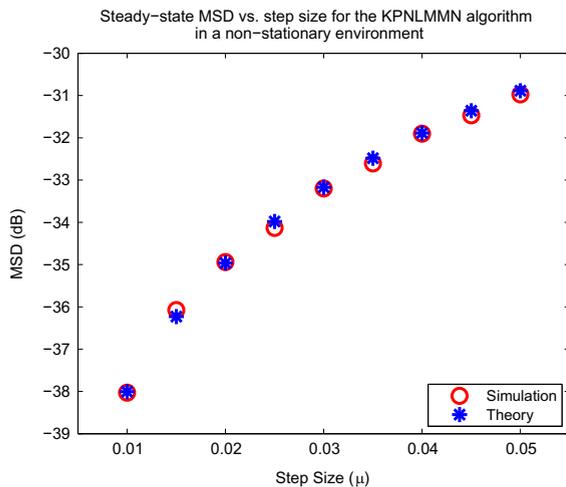


Fig. 12. Dependence of the steady-state MSD on the step size  $\mu$  for the KPNLMF algorithm.



**Fig. 13.** Dependence of the steady-state MSD on the step size  $\mu$  for the KPNLMMN algorithm in a non-stationary environment.

the simulated results for the tracking performance of the KPNLMMN algorithm.

## 8. Conclusion

In this paper, we derive the proportionate update and Krylov-proportionate update based algorithms through the natural gradient descent algorithm, enabling the derivation of new variants of this important family of algorithms. We propose the stable-PNLMS and the stable-KPNLMS algorithms overcoming the well-known stability issues due to the use of the mean fourth error cost function. We propose the KPNLMMN algorithm as a convex mixture combination of the mean-square and mean-fourth error objectives, which achieves superior performance with respect to the both constituent filters. Finally, we provide a comprehensive performance analysis in the steady state, tracking and transient phases for all introduced algorithms, and demonstrate the accuracy of our derivations with several different numerical examples under various configurations.

## References

- [1] T.Y. Ji, Z. Lu, Q.H. Wu, Optimal soft morphological filter for periodic noise removal using a particle swarm optimiser with passive congregation, *Signal Process.* 87 (2007) 2799–2809.
- [2] B.G. Jeong, B.C. Kim, Y.H. Moon, K. Eom, Simplified noise model parameter estimation for signal dependent noise, *Signal Process.* 96 (2014) 266–273.
- [3] S.Q. Yuan, Y.H. Tan, H.L. Sun, Impulse noise removal by the difference-type noise detector and the cost function-type filter, *Signal Process.* 87 (2007) 2417–2430.
- [4] C. Stanciu, J. Benesty, C. Paleologu, T. Gansler, S. Ciochina, A widely linear model for stereophonic acoustic echo cancellation, *Signal Process.* 93 (2013) 511–516.
- [5] C. Paleologu, J. Benesty, S. Ciochina, Widely linear general Kalman filter for stereophonic acoustic echo cancellation, *Signal Process.* 94 (2014) 570–575.
- [6] P.A. Naylor, J. Cui, M. Brookes, Adaptive algorithms for sparse echo cancellation, *Signal Process.* 86 (2006) 1182–1192.
- [7] C. Schuldt, F. Lindstrom, H. Li, I. Claesson, Adaptive filter length selection for acoustic echo cancellation, *Signal Process.* 89 (2009) 1185–1194.

- [8] X. Jiang, T. Kirubarajan, W.J. Zeng, Robust sparse channel estimation and equalization in impulsive noise using linear programming, *Signal Process.* 93 (2013) 1095–1105.
- [9] R. Arablouei, K. Dogancay, Low-complexity adaptive decision-feedback equalization of MIMO channels, *Signal Process.* 92 (2012) 1515–1524.
- [10] E. Walach, B. Widrow, The least mean fourth (LMF) adaptive algorithm and its family, *IEEE Trans. Inf. Theory* 30 (1984) 275–283.
- [11] P.I. Hubsher, J.C.M. Bermudez, An improved statistical analysis of the least mean fourth (LMF) adaptive algorithm, *IEEE Trans. Signal Process.* 51 (2003) 664–671.
- [12] P.I. Hubsher, J.C.M. Bermudez, V.H. Nascimento, A mean square stability analysis of the least mean fourth adaptive algorithm, *IEEE Trans. Signal Process.* 55 (2007) 4018–4028.
- [13] A. Zerguine, Convergence behavior of the normalized least mean fourth algorithm, in: *Proceedings of the 34th Asilomar Conference on Signals, Systems and Computers (ACSSC)*, 2000, pp. 275–278.
- [14] A. Zerguine, Convergence and steady-state analysis of the normalized least mean fourth algorithm, *Digit. Signal Process.* 17 (2007) 17–31.
- [15] N.J. Bershad, J.C.M. Bermudez, Mean-square stability of the normalized least-mean fourth algorithm for white Gaussian inputs, *Digit. Signal Process.* 21 (2011) 694–700.
- [16] D.L. Duttweiler, Proportionate normalized least-mean-squares adaptation in echo cancelers, *IEEE Trans. Speech Audio Process.* 8 (2000) 508–518.
- [17] J. Benesty, S. Gay, An improved PNLMS algorithm, in: *Proceedings of the IEEE International Conference on Acoustic, Speech, and Signal Processing (ICASSP)*, 2002, pp. 1881–1884.
- [18] Y. Yilmaz, S.S. Kozat, An extended version of the NLMS algorithm based on proportionate Krylov subspace projections, in: *Proceedings of the IEEE International Conference on Machine Learning and Applications (ICMLA)*, 2009, pp. 404–408.
- [19] M. Yukawa, On whitening for Krylov-proportionate normalized least-mean-square algorithm, in: *Proceedings of the IEEE Workshop on Machine Learning for Signal Processing (MLSP)*, 2008, pp. 315–320.
- [20] M. Yukawa, Krylov-proportionate adaptive filtering techniques not limited to sparse systems, *IEEE Trans. Signal Process.* 57 (2009) 927–943.
- [21] M. Yukawa, W. Utschick, Proportionate adaptive algorithm for non-sparse systems based on Krylov subspace and constrained optimization, in: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2009, pp. 3121–3124.
- [22] O. Tanrikulu, J.A. Chambers, Convergence and steady-state properties of the least-mean mixed norm (LMMN) adaptive algorithm, *IEEE Proc. Vis. Image Signal Process.* 143 (1996) 137–142.
- [23] E. Eweda, Global stabilization of the least mean fourth algorithm, *IEEE Trans. Signal Process.* 60 (2012) 1473–1477.
- [24] E. Eweda, N.J. Bershad, Stochastic analysis of a stable normalized least mean fourth algorithm for adaptive noise cancelling with a white Gaussian reference, *IEEE Trans. Signal Process.* 60 (2012) 6235–6244.
- [25] M.O. Sayin, N.D. Vanli, S.S. Kozat, A novel family of adaptive filtering algorithms based on the logarithmic cost, *IEEE Trans. Signal Process.* 62 (2014) 4411–4424.
- [26] S. Amari, Natural gradient works efficiently in learning, *Neural Comput.* 10 (1998) 251–276.
- [27] S. Amari, Differential-geometrical methods in statistics, in: *Lecture Notes in Statistics*, vol. 28, Springer-Verlag, NY, 1985.
- [28] D. Comminiello, M. Scarpiniti, L.A. Azpicueta-Ruiz, J. Arenas-Garcia, A. Uncini, Nonlinear acoustic echo cancellation based on sparse functional link representations, *IEEE Trans. Audio Speech Lang. Process.* 22 (2014) 1172–1183.
- [29] S.L. Gay, S.C. Douglas, Normalized natural gradient adaptive filtering for sparse and non-sparse systems, in: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2002, pp. 1405–1408.
- [30] F.T. Castoldi, M.L.R. Campos, Application of a minimum-disturbance description to constrained adaptive filters, *IEEE Signal Process. Lett.* 20 (2013) 1215–1218.
- [31] M.A. Donmez, H.A. Inan, S.S. Kozat, Adaptive mixture methods based on Bregman divergences, *Digit. Signal Process.* 23 (2013) 86–97.
- [32] J. Benesty, C. Paleologu, S. Ciochina, On regularization in adaptive filtering, *IEEE Trans. Audio Speech Lang. Process.* 19 (2011) 1734–1742.

- [33] A.H. Sayed, *Fundamentals of Adaptive Filtering*, Wiley-IEEE Press, Wiley, NJ, 2003.
- [34] W.E. Arnoldi, The principle of minimized iteration in the solution of the matrix eigenvalue problem, *Q. Appl. Math.* 9 (1951) 17–29.
- [35] Y. Saad, Krylov subspace methods for solving large unsymmetric linear systems, *Math. Comput.* 37 (1981) 105–126.
- [36] T.Y. Al-Naffouri, A.H. Sayed, Transient analysis of adaptive filters with error nonlinearities, *IEEE Trans. Signal Process.* 51 (2003) 653–663.
- [37] R. Price, A useful theorem for nonlinear devices having Gaussian inputs, *IEEE Trans. Inf. Theory* 4 (1958) 69–72.
- [38] E. McMahon, An extension of Price's theorem (Correspondence), *IEEE Trans. Inf. Theory* 10 (1964) 168.
- [39] T. Koh, E.J. Powers, Efficient methods of estimate correlation functions of Gaussian processes and their performance analysis, *IEEE Trans. Acoust. Speech Signal Process.* 33 (1985) 1032–1035.
- [40] C.L. Siegel, *Topics in Complex Function Theory, Vol. 2: Automorphic Functions and Abelian Integrals*, Wiley, NY, 1988.
- [41] S.C. Chan, Y. Zhou, Convergence behavior of NLMS algorithm for Gaussian inputs: solutions using generalized Abelian integral functions and step size selection, *J. Signal Process. Syst.* 59 (2010) 255–265.
- [42] T.Y. Al-Naffouri, A.H. Sayed, Adaptive filters with error nonlinearities: mean square analysis and optimum design, *EURASIP J. Appl. Signal Process.* 4 (2001) 192–205.
- [43] H.L.V. Trees, *Detection, Estimation, and Modulation Theory*, Wiley, NY, 2004.
- [44] Recommendation ITU-T P.50, *Artificial Voices*, 1999.
- [45] J. Allen, D. Berkley, Image method for efficiently simulating small-room acoustics, *J. Acoust. Soc. Am.* 65 (1979).
- [46] E. Lehmann, A. Johansen, Prediction of energy decay in room impulse responses simulated with an image-source model, *J. Acoust. Soc. Am.* 124 (2008).
- [47] G. Gui, F. Adachi, Adaptive sparse system identification using normalized least mean fourth algorithm, *Int. J. Commun. Syst.* 10 (2013).
- [48] T. Aboulnasr, K. Mayyas, A robust variable step-size LMS-type algorithm: analysis and simulations, *IEEE Trans. Signal Process.* 45 (1997) 631–639.