Robust Reduced-Rank Adaptive Algorithm Based on Parallel Subgradient Projection and Krylov Subspace

Masahiro Yukawa, *Member, IEEE*, Rodrigo C. de Lamare, *Member, IEEE*, and Isao Yamada, *Senior Member, IEEE*

***Abstract*—In this paper, we propose a novel reduced-rank adaptive filtering algorithm exploiting the Krylov subspace associated with estimates of certain statistics of input and output signals. We point out that, when the estimated statistics are erroneous (e.g., due to sudden changes of environments), the existing Krylov-subspace-based reduced-rank methods compute the point that minimizes a "wrong" mean-square error (MSE) in the subspace. The proposed algorithm exploits the set-theoretic adaptive filtering framework for tracking efficiently the optimal point in the sense of minimizing the "true" MSE in the subspace. Therefore, compared with the existing methods, the proposed algorithm is more suited to adaptive filtering applications. A convergence analysis of the algorithm is performed by extending the adaptive projected subgradient method (APSM). Numerical examples demonstrate that the proposed algorithm enjoys better tracking performance than the existing methods for system identification problems.**

***Index Terms*—Krylov subspace, reduced-rank adaptive filtering, set theory, subgradient methods.**

I. Introduction

REDUCED-RANK adaptive filtering has attracted significant attention over several research communities including signal processing; e.g., [1]–[12]. Whereas early works were motivated by the so-called overmodeling problem, many of the recent works were motivated mainly by computational-constraints and slow-convergence problems due to a large number of parameters. Specifically, a Krylov subspace associated with the input autocorrelation matrix and the crosscorrelation vector between input and output has been used in several methods: Cayley–Hamilton receiver [13], multistage Wiener filter (MSWF) [14]–[16], auxiliary-vector filtering (AVF) [17], [18], Powers of R (POR) receiver [19], and the conjugate gradient reduced-rank filter (CGRRF) [20]–[22] (see [23]–[25] for their connections). All of those previous studies focus on minimizing a mean-square error (MSE) within the Krylov subspace (see [26] for linear estimation and detection in Krylov subspaces). However, in the erroneous case (i.e., in cases where there is a mismatch in estimates of the autocorrelation matrix and the cross-correlation vector), the methods minimize an "erroneous" MSE function in the Krylov subspace. Therefore, the solution obtained at each iteration is no longer "optimal" in the sense of minimizing the "true" MSE within the Krylov subspace.

In this paper, we propose an adaptive technique, named *Krylov reduced-rank adaptive parallel subgradient projection (KRR-APSP) algorithm*, tracking directly the "optimal" solution in the Krylov subspace. The KRR-APSP algorithm firstly performs dimensionality reduction with an orthonormal basis of the Krylov subspace, followed by adjustments of the coefficients of a lower-dimensional filter based on *the set-theoretic adaptive filtering framework*[1] [29]. As a result, in cases where the environment changes dynamically (which makes the estimates of the statistics erroneous), the KRR-APSP algorithm realizes better tracking capability than the existing Krylov-subspace-based methods. (The computational complexity is comparable to the existing methods.)

The rest of the paper is organized as follows. In Section II, the motivation and the problem statement are presented, in which it is shown that, in a low-dimensional Krylov subspace, i) the achievable MSE is close to the minimum MSE (MMSE) and ii) system identification of high accuracy is possible, provided that the condition number of the autocorrelation matrix is close to unity. In Section III, we present the proposed reduced-rank algorithm, and discuss its tracking property and computational complexity. The KRR-APSP algorithm i) designs multiple closed convex sets consistent with the recently arriving data, and ii) moves the filter toward the intersection of the convex sets (to find a feasible solution) by means of parallel subgradient projection at each iteration. Because the noise is taken into account in the set design, KRR-APSP is intrinsically robust. In Section IV, to prove important properties (*monotonicity* and *asymptotic optimality*) of the proposed algorithm, we firstly present an alternative derivation of the algorithm from an extended version of *the adaptive projected subgradient method (APSM)*[2] [38], [39], and then present an analysis of the extended APSM. It is revealed that, in the (original) high dimensional vector space, the proposed algorithm performs parallel subgradient projection in a series of Krylov subspaces. In Section V, numerical examples are presented to verify the advantages of the proposed algorithm over CGRRF, followed by the conclusion in Section VI.

Manuscript received August 18, 2008; accepted June 09, 2009. First published July 14, 2009; current version published November 18, 2009. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Dennis R. Morgan. This work was done in part while M. Yukawa was with the Department of Electronics, University of York, U.K.

M. Yukawa is with the Laboratory for Mathematical Neuroscience, BSI, RIKEN, Saitama, 351-0198, Japan (e-mail: myukawa@riken.jp).

R. C. de Lamare is with the Department of Electronics, University of York, York, YO10 5DD, U.K. (e-mail: rcdl500@ohm.york.ac.uk).

I. Yamada is with the Department of Communications and Integrated Systems, S3-60, Tokyo Institute of Technology, Tokyo 152-8552, Japan (e-mail: isao@comm.ss.titech.ac.jp).

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

Digital Object Identifier 10.1109/TSP.2009.2027397

[1] A related approach called *set-membership adaptive filtering* has independently been developed, e.g., in [27] and [28].

[2] APSM has proven a promising tool to derive efficient algorithms in many applications [30]–[37].

II. Motivation and Problem Statement

Let $\mathbb{R}$, $\mathbb{N}_0$, and $\mathbb{N}$ denote the sets of all real numbers, nonnegative integers, and positive integers, respectively. We consider the following linear model:

$$d_k := \boldsymbol{u}_k^T \boldsymbol{h}^* + n_k, \quad \forall k \in \mathbb{N}_0 \tag{1}$$

where $\boldsymbol{u}_k := [u_k, u_{k-1}, \ldots, u_{k-N+1}]^T \in \mathbb{R}^N$ ($N \in \mathbb{N}$) denotes the input vector, $\boldsymbol{h}^* \in \mathbb{R}^N$ the unknown system, n_k the additive noise, and d_k the output (k: sample index, $(\,\cdot\,)^T$: *transposition*). The MMSE filter in $\mathbb{R}^N$ is well-known to be characterized by the so-called Wiener–Hopf equation $\boldsymbol{R}\boldsymbol{h}_{\mathrm{MMSE}} = \boldsymbol{p}$ (see, e.g., [40]), where $\boldsymbol{R} := \mathrm{E}\{\boldsymbol{u}_k \boldsymbol{u}_k^T\}$ and $\boldsymbol{p} := \mathrm{E}\{\boldsymbol{u}_k d_k\}$ ($\mathrm{E}\{\cdot\}$: *expectation*). For simplicity, we assume that $\boldsymbol{R}$ is invertible and the input and the noise are (statistically) orthogonal; i.e., $E\{n_k \boldsymbol{u}_k\} = \boldsymbol{0}$. In this case, $\boldsymbol{p} = \mathrm{E}\{\boldsymbol{u}_k(\boldsymbol{u}_k^T \boldsymbol{h}^* + n_k)\} = \boldsymbol{R}\boldsymbol{h}^*$, and the MSE function $f : \mathbb{R}^N \to [0, \infty)$ is given as

$$\begin{aligned} f(\boldsymbol{h}) &:= \mathrm{E}\{(d_k - \boldsymbol{h}^T \boldsymbol{u}_k)^2\} \\ &= \boldsymbol{h}^T \boldsymbol{R}\boldsymbol{h} - 2\boldsymbol{h}^T \boldsymbol{p} + \sigma_d^2 \\ &= \|\boldsymbol{h} - \boldsymbol{h}^*\|_{\boldsymbol{R}}^2 - \|\boldsymbol{h}^*\|_{\boldsymbol{R}}^2 + \sigma_d^2. \end{aligned} \tag{2}$$

Here, $\sigma_d^2 := \mathrm{E}\{d_k^2\}$ and $\|\cdot\|_{\boldsymbol{R}}$ is the $\boldsymbol{R}$-norm[3] defined for any vector $\boldsymbol{a} \in \mathbb{R}^N$ as $\|\cdot\|_{\boldsymbol{R}} := \sqrt{\boldsymbol{a}^T \boldsymbol{R}\boldsymbol{a}}$. From (2), it is seen that $\boldsymbol{h}^* = \boldsymbol{h}_{\mathrm{MMSE}} (= \boldsymbol{R}^{-1}\boldsymbol{p})$.

Let us now consider, for $D \in \{1, 2, \ldots, N\}$, the MMSE filter within the following Krylov subspace:

$$\begin{aligned} \mathcal{K}_D(\boldsymbol{R}, \boldsymbol{p}) &:= \mathrm{span}\{\boldsymbol{p}, \boldsymbol{R}\boldsymbol{p}, \ldots, \boldsymbol{R}^{D-1}\boldsymbol{p}\} && (3) \\ &= \mathrm{span}\{\boldsymbol{R}\boldsymbol{h}^*, \boldsymbol{R}^2\boldsymbol{h}^*, \ldots, \boldsymbol{R}^D\boldsymbol{h}^*\} \subset \mathbb{R}^N. && (4) \end{aligned}$$

Referring to (2), the MMSE solution in $\mathcal{K}_D(\boldsymbol{R}, \boldsymbol{p})$ is characterized by

$$P^{(\boldsymbol{R})}_{\mathcal{K}_D(\boldsymbol{R},\boldsymbol{p})}(\boldsymbol{h}^*) \in \arg \min_{\boldsymbol{h} \in \mathcal{K}_D(\boldsymbol{R},\boldsymbol{p})} \|\boldsymbol{h}^* - \boldsymbol{h}\|_{\boldsymbol{R}} \tag{5}$$

where we denote by $P_C^{(\boldsymbol{A})}(\boldsymbol{x})$ the metric projection of a vector $\boldsymbol{x}$ onto a closed convex set C in the $\boldsymbol{A}$-norm sense. In particular, the metric projection in the sense of Euclidean norm is denoted simply by $P_C(\boldsymbol{x})$. In words, the MMSE filter in the subspace is the best approximation, in the $\boldsymbol{R}$-norm sense, of $\boldsymbol{h}^*$ in $\mathcal{K}_D(\boldsymbol{R}, \boldsymbol{p})$. Noting that $P^{(\boldsymbol{R})}_{\mathcal{K}_D(\boldsymbol{R},\boldsymbol{p})}(\boldsymbol{h}^*)$ coincides with the vector obtained through D steps of the conjugate gradient (CG) method with its initial point being the zero vector, the MSE is bounded as follows [41, Theorem 10.2.6]:

$$f\left(P^{(\boldsymbol{R})}_{\mathcal{K}_D(\boldsymbol{R},\boldsymbol{p})}(\boldsymbol{h}^*)\right) \le \left[4\left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}\right)^{2D} - 1\right] \|\boldsymbol{h}^*\|_{\boldsymbol{R}}^2 + \sigma_d^2 \tag{6}$$

where $\kappa := \|\boldsymbol{R}\|_2 \|\boldsymbol{R}^{-1}\|_2 \ge 1$ is the condition number of $\boldsymbol{R}$; $\|\cdot\|_2$ denotes the spectral norm. System identifiability in $\mathcal{K}_D(\boldsymbol{R}, \boldsymbol{p})$ is discussed below.

Remark 1: How accurately can the system $\boldsymbol{h}^*$ be identified in the subspace $\mathcal{K}_D(\boldsymbol{R}, \boldsymbol{p})$? In the system identification problem, we wish to minimize the Euclidean norm $\|\boldsymbol{h}^* - \boldsymbol{h}\|$ rather than the $\boldsymbol{R}$-norm $\|\boldsymbol{h}^* - \boldsymbol{h}\|_{\boldsymbol{R}}$. To clarify the difference between the MSE minimization and the system identification over $\mathcal{K}_D(\boldsymbol{R}, \boldsymbol{p})$, the projections in the different senses are illustrated in Fig. 1. By the Rayleigh–Ritz theorem [42], it is readily verified that $\lambda_{\max}^{-1/2}\|\boldsymbol{x}\|_{\boldsymbol{R}} \le \|\boldsymbol{x}\| \le \lambda_{\min}^{-1/2}\|\boldsymbol{x}\|_{\boldsymbol{R}}$ for any $\boldsymbol{x} \in \mathbb{R}^N$, where $\lambda_{\max} > 0$ and $\lambda_{\min} > 0$ denote the maximum and minimum eigenvalues of $\boldsymbol{R}$, respectively. It is thus verified that $\|P_{\mathcal{K}_D(\boldsymbol{R},\boldsymbol{p})}(\boldsymbol{h}^*) - P^{(\boldsymbol{R})}_{\mathcal{K}_D(\boldsymbol{R},\boldsymbol{p})}(\boldsymbol{h}^*)\| \le \|\boldsymbol{h}^* - P^{(\boldsymbol{R})}_{\mathcal{K}_D(\boldsymbol{R},\boldsymbol{p})}(\boldsymbol{h}^*)\| \le \lambda_{\min}^{-1/2}\|\boldsymbol{h}^* - P^{(\boldsymbol{R})}_{\mathcal{K}_D(\boldsymbol{R},\boldsymbol{p})}(\boldsymbol{h}^*)\|_{\boldsymbol{R}} \le 2\lambda_{\min}^{-1/2}\|\boldsymbol{h}^*\|_{\boldsymbol{R}}\alpha^D(\kappa)$, where $\alpha(\kappa) := (\sqrt{\kappa} - 1)/(\sqrt{\kappa} + 1) \in [0, 1)$. Here, the first inequality is due to the basic property of projection, and the third one is verified by [41, Theorem 10.2.6]. This suggests that system identification of high accuracy would be possible for a small D when $\kappa \approx 1$ (If $\kappa \gg 1$, preconditioning[4] should be performed).

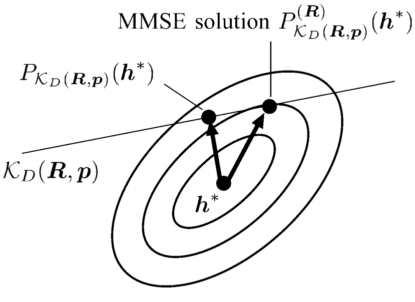

MMSE solution $P^{(\boldsymbol{R})}_{\mathcal{K}_D(\boldsymbol{R},\boldsymbol{p})}(\boldsymbol{h}^*)$
$P_{\mathcal{K}_D(\boldsymbol{R},\boldsymbol{p})}(\boldsymbol{h}^*)$
$\mathcal{K}_D(\boldsymbol{R}, \boldsymbol{p})$
$\boldsymbol{h}^*$

Fig. 1. $P_{\mathcal{K}_D(\boldsymbol{R},\boldsymbol{p})}(\boldsymbol{h}^*)$ and $P^{(\boldsymbol{R})}_{\mathcal{K}_D(\boldsymbol{R},\boldsymbol{p})}(\boldsymbol{h}^*)$ with the equal error contours of the MSE surface.

In reality, $\boldsymbol{R}$ and $\boldsymbol{p}$ are rarely available, thus should be estimated from observed measurements. Let $\widehat{\boldsymbol{R}}$ and $\widehat{\boldsymbol{p}}$ be estimates of $\boldsymbol{R}$ and $\boldsymbol{p}$, respectively, and $\widehat{\boldsymbol{h}}^*$ be characterized by $\widehat{\boldsymbol{R}}\widehat{\boldsymbol{h}}^* = \widehat{\boldsymbol{p}}$. CGRRF [20]–[22] computes, at each iteration, the best approximation of $\widehat{\boldsymbol{h}}^*$ in $\mathcal{K}_D(\widehat{\boldsymbol{R}}, \widehat{\boldsymbol{p}})$ in the $\widehat{\boldsymbol{R}}$-norm sense; i.e., $P^{(\widehat{\boldsymbol{R}})}_{\mathcal{K}_D(\widehat{\boldsymbol{R}},\widehat{\boldsymbol{p}})}(\widehat{\boldsymbol{h}}^*)$. This realizes significantly fast convergence and reasonable steady-state performance as long as good estimates are available; i.e., $\widehat{\boldsymbol{R}} \approx \boldsymbol{R}$ and $\widehat{\boldsymbol{p}} \approx \boldsymbol{p}$. However, once those estimates become unreliable (which happens when the environments change suddenly), $P^{(\widehat{\boldsymbol{R}})}_{\mathcal{K}_D(\widehat{\boldsymbol{R}},\widehat{\boldsymbol{p}})}(\widehat{\boldsymbol{h}}^*)$ makes little sense, and CGRRF (or the other existing Krylov-subspace-based methods) should wait until a certain amount of data arrive to recapture reasonable estimates.

The goal of this paper is to propose an alternative to the existing Krylov-subspace-based methods to address this restriction. To be specific, the main problem in this work is stated as follows. Given that the Krylov subspace is employed for dimensionality reduction, the problem is to design an efficient algorithm that can always track $P^{(\boldsymbol{R})}_{\mathcal{K}_D(\widehat{\boldsymbol{R}},\widehat{\boldsymbol{p}})}(\boldsymbol{h}^*)$, which minimizes

[3] The $\boldsymbol{R}$-norm is also called *the energy norm induced by* $\boldsymbol{R}$. The same norm is used in [23] to derive the CG method.

[4] The importance of preconditioning is well-known in numerical linear algebra; see, e.g., [43], [44] and the references therein. Also the importance is mentioned in [45] for an application of the conjugate gradient method to the adaptive filtering problem. Different types of CG-based adaptive filtering algorithms have also been proposed, e.g., in [46], [47].

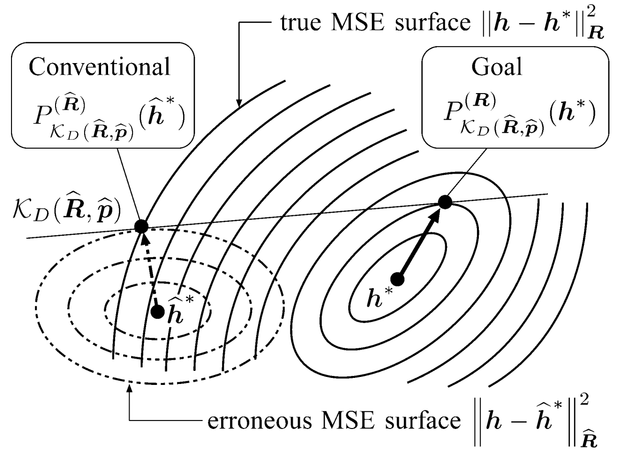

true MSE surface $\|\boldsymbol{h}-\boldsymbol{h}^*\|^2_{\boldsymbol{R}}$
Conventional
$P^{(\widehat{\boldsymbol{R}})}_{\mathcal{K}_D(\widehat{\boldsymbol{R}},\widehat{\boldsymbol{p}})}(\widehat{\boldsymbol{h}}^*)$
Goal
$P^{(\boldsymbol{R})}_{\mathcal{K}_D(\widehat{\boldsymbol{R}},\widehat{\boldsymbol{p}})}(\boldsymbol{h}^*)$
$\mathcal{K}_D(\widehat{\boldsymbol{R}},\widehat{\boldsymbol{p}})$
$\widehat{h}^*$
h^*
erroneous MSE surface $\left\|h-\widehat{h}^*\right\|^2_{\widehat{\boldsymbol{R}}}$

Fig. 2. Illustration of the goal of this paper. "Conventional" stands for the conventional Krylov-subspace-based methods such as CGRRF.

the true MSE $f(\boldsymbol{h})$ over $\mathcal{K}_D(\widehat{\boldsymbol{R}},\widehat{\boldsymbol{p}})$ [see (2)]. Such an algorithm should have better tracking capability than the existing methods after dynamic changes of environments, because $P^{(\widehat{\boldsymbol{R}})}_{\mathcal{K}_D(\widehat{\boldsymbol{R}},\widehat{\boldsymbol{p}})}(\widehat{\boldsymbol{h}}^*)$ does not minimize the true MSE as long as the estimates $\widehat{\boldsymbol{R}}$ and $\widehat{\boldsymbol{p}}$ are erroneous. The concept is illustrated in Fig. 2, in which the estimates are assumed to become erroneous. Note in the figure that the difference between $f(\boldsymbol{h})$ and $\|\boldsymbol{h}-\boldsymbol{h}^*\|^2_{\boldsymbol{R}}$ is a constant in terms of $\boldsymbol{h}$, which makes no difference in the equal error contours. In the following section, we present an adaptive algorithm that achieves this goal.

III. Proposed Reduced-Rank Adaptive Filter

This section consists of the following subsections.

A. Rank-reduction Matrix and Concepts of Set-Theoretic Adaptive Filtering
B. Design of Closed Convex Sets
C. Proposed KRR-APSP Algorithm—Realization of Monotone Approaching
D. On the Parameters Used in KRR-APAP
E. Tracking Property: We show that the proposed algorithm tracks $P^{(\boldsymbol{R})}_{\mathcal{K}_D(\widehat{\boldsymbol{R}},\widehat{\boldsymbol{p}})}(\boldsymbol{h}^*)$
F. Computational Complexity
G. Robustness Issue Against Impulsive Noise

In the following, we let $\langle\cdot,\cdot\rangle$ and $\|\cdot\|$ denote the standard inner product and its induced norm (i.e., the Euclidean norm), respectively, in any dimensional Euclidean space.

A. Rank-Reduction Matrix and Concepts of Set-Theoretic Adaptive Filtering

Let $\widehat{\boldsymbol{R}}_k$ and $\widehat{\boldsymbol{p}}_k$ be estimates of $\boldsymbol{R}$ and $\boldsymbol{p}$ at time $k\in\mathbb{N}_0$, respectively (how to compute $\widehat{\boldsymbol{R}}_k$ and $\widehat{\boldsymbol{p}}_k$ is described in Section III-F). Also let $\boldsymbol{S}_k$ be an $N\times D$ matrix whose column vectors form an orthonormal basis[5] (in the sense of the standard inner product) of the subspace $\mathcal{K}_D(\widehat{\boldsymbol{R}}_k,\widehat{\boldsymbol{p}}_k)$. The orthonormalization can be accomplished through either the well-known *Gram–Schmidt method* or more efficient *Lanczos method* [41]. For dimensionality reduction, we force the adaptive filter $\boldsymbol{h}_k\in\mathbb{R}^N$ to lie in $\mathcal{R}(\boldsymbol{S}_k)=\mathcal{K}_D(\widehat{\boldsymbol{R}}_k,\widehat{\boldsymbol{p}}_k)\subset\mathbb{R}^N$ at each time instant k, where $\mathcal{R}(\cdot)$ stands for the *range space*. Thus, for a lower dimensional vector $\widetilde{\boldsymbol{h}}_k\in\mathbb{R}^D$, the adaptive filter is characterized as $\boldsymbol{h}_k=\boldsymbol{S}_k\widetilde{\boldsymbol{h}}_k$. In the following, a tilde will be used for expressing a D-dimensional vector (or a subset of $\mathbb{R}^D$). The output of the adaptive filter is given by

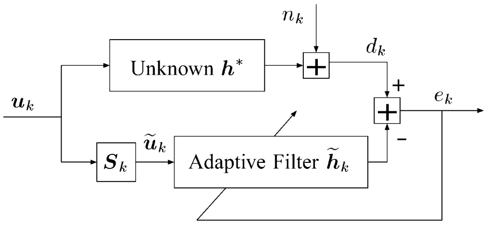

n_k
d_k
Unknown $\boldsymbol{h}^*$
$\boldsymbol{u}_k$
+
e_k
$\boldsymbol{S}_k$
$\widetilde{\boldsymbol{u}}_k$
Adaptive Filter $\widetilde{\boldsymbol{h}}_k$
-

Fig. 3. Reduced-rank adaptive filtering scheme.

$$\boldsymbol{h}_k^T\boldsymbol{u}_k=\widetilde{\boldsymbol{h}}_k^T\boldsymbol{S}_k^T\boldsymbol{u}_k=\widetilde{\boldsymbol{h}}_k^T\widetilde{\boldsymbol{u}}_k\quad\left(\widetilde{\boldsymbol{u}}_k:=\boldsymbol{S}_k^T\boldsymbol{u}_k\in\mathbb{R}^D\right).\tag{7}$$

The reduced-rank adaptive filtering scheme is illustrated in Fig. 3.

We exploit the set-theoretic adaptive filtering framework [29] for tracking $P^{(\boldsymbol{R})}_{\mathcal{K}_D(\widehat{\boldsymbol{R}},\widehat{\boldsymbol{p}})}(\boldsymbol{h}^*)$. The basic idea is the following:

1) construct (possibly multiple) closed convex sets containing the optimal filter, say $\boldsymbol{h}_{\rm opt}$, with high probability;
2) approach the intersection of those sets monotonically at each iteration (more details about the monotone approaching will be discussed in Section IV).

The concepts of set designing and monotone approaching lead to stable convergence behavior of the adaptive algorithm. *How can we design such closed convex sets? How can we realize the monotone approaching in a computationally efficient way?* The answer is given in the following.

B. Design of Closed Convex Sets

Given $r\in\mathbb{N}$, we define

$$\begin{aligned}\boldsymbol{U}_k&:=[\boldsymbol{u}_k,\boldsymbol{u}_{k-1},\ldots,\boldsymbol{u}_{k-r+1}]\in\mathbb{R}^{N\times r}\\ \boldsymbol{d}_k&:=[d_k,d_{k-1},\ldots,d_{k-r+1}]^T\in\mathbb{R}^r\\ \boldsymbol{e}_k(\boldsymbol{h})&:=\boldsymbol{U}_k^T\boldsymbol{h}-\boldsymbol{d}_k\in\mathbb{R}^r,\quad\forall\boldsymbol{h}\in\mathbb{R}^N.\end{aligned}$$

Recall that our goal is to find the $\boldsymbol{h}\in\mathcal{K}_D(\widehat{\boldsymbol{R}},\widehat{\boldsymbol{p}})\subset\mathbb{R}^N$ that minimizes the MSE function (see (2)). This means that the error $\|\boldsymbol{e}_k(\boldsymbol{h})\|^2$ should be minimized in an average sense. Apparently, due to the presence of noise, there is in general no filter $\boldsymbol{h}\in\mathbb{R}^N$ such that $\|\boldsymbol{e}_k(\boldsymbol{h})\|^2=0$, $\forall k\in\mathbb{N}$. Hence, introducing a constant $\rho\geq 0$, we define the following closed convex sets:

$$C_\iota^{(k)}(\rho):=\{\boldsymbol{h}\in\mathcal{R}(\boldsymbol{S}_k):g_\iota(\boldsymbol{h}):=\|\boldsymbol{e}_\iota(\boldsymbol{h})\|^2-\rho\leq 0\},\qquad \iota\leq k\in\mathbb{N}_0.\tag{8}$$

How can we determine the value of ρ? In the full-rank case (i.e., in the case where $\mathcal{R}(\boldsymbol{S}_k)=\mathbb{R}^N$), $\boldsymbol{h}_{\rm opt}:=\boldsymbol{h}^*$, and, by noticing $\boldsymbol{d}_k=\boldsymbol{U}_k^T\boldsymbol{h}^*+\boldsymbol{n}_k$, $k\in\mathbb{N}_0$, where $\boldsymbol{n}_k:=[n_k,n_{k-1},\ldots,n_{k-r+1}]^T\in\mathbb{R}^r$, we have

$$\boldsymbol{h}_{\rm opt}\in C_\iota^{(k)}(\rho)\Leftrightarrow\|\boldsymbol{e}_\iota(\boldsymbol{h}_{\rm opt})\|^2=\|\boldsymbol{n}_\iota\|^2\leq\rho.\tag{9}$$

[5] The orthonormality is essential in the analysis (see Section IV-B).

TABLE I
EFFICIENT IMPLEMENTATION OF THE PROPOSED ALGORITHM

```
Requirements: Initial transformation matrix $\boldsymbol{S}_0$, inputs $(\boldsymbol{U}_k)_{k\in\mathbb{N}_0}$,
outputs $(\boldsymbol{d}_k)_{k\in\mathbb{N}_0}$, control sequence $\mathcal{I}_k$, step size $\lambda_k \in [0,2]$,
weights $w_\iota^{(k)}$, $\forall\iota\in\mathcal{I}_k$, initial vector $\widetilde{\boldsymbol{h}}_0 \in \mathbb{R}^D$, constant $\rho \geq 0$, $m \in \mathbb{N}$
1.   Filter output: $y_k := \widetilde{\boldsymbol{u}}_k^T \widetilde{\boldsymbol{h}}_k (= \boldsymbol{u}_k^T \boldsymbol{S}_k \widetilde{\boldsymbol{h}}_k)$
2.   Filter update:
     (a) For $\iota \in \mathcal{I}_k$, do the following:
         $\boldsymbol{U}_\iota^{(k)} := \boldsymbol{S}_k^T \boldsymbol{U}_\iota \in \mathbb{R}^{D\times r}$
         $\boldsymbol{e}_\iota^{(k)} := (\boldsymbol{U}_\iota^{(k)})^T \widetilde{\boldsymbol{h}}_k - \boldsymbol{d}_\iota \in \mathbb{R}^r$
           If $\left\|\boldsymbol{e}_\iota^{(k)}\right\|^2 \leq \rho$,
             $\widetilde{\boldsymbol{\delta}}_\iota^{(k)} := \boldsymbol{0} \in \mathbb{R}^D$, $\ell_\iota^{(k)} := 0$
           else
             $\boldsymbol{a}_\iota^{(k)} := \boldsymbol{U}_\iota^{(k)} \boldsymbol{e}_\iota^{(k)} \in \mathbb{R}^D$
             $c_\iota^{(k)} := \left\|\boldsymbol{a}_\iota^{(k)}\right\|^2 \in [0,\infty)$
             $d_\iota^{(k)} := \rho - \left\|\boldsymbol{e}_\iota^{(k)}\right\|^2 \in (-\infty,\rho]$
             $\widetilde{\boldsymbol{\delta}}_\iota^{(k)} := w_\iota^{(k)} d_\iota^{(k)} \boldsymbol{a}_\iota^{(k)}/(2c_\iota^{(k)}) \in \mathbb{R}^D$
             $\ell_\iota^{(k)} := \left(\left\|\boldsymbol{\delta}_\iota^{(k)}\right\|^2 / w_\iota^{(k)} =\right) w_\iota^{(k)} (d_\iota^{(k)})^2/(4c_\iota^{(k)}) \in (0,\infty)$
           endif;
         end;
     (b) If $\left\|\boldsymbol{e}_\iota^{(k)}\right\|^2 \leq \rho$ for all $\iota \in \mathcal{I}_k$,
           $\widetilde{\boldsymbol{h}}_{k+1} := \widetilde{\boldsymbol{h}}_k \in \mathbb{R}^D$
         else
           $\widetilde{\boldsymbol{f}}_k := \sum_{\iota\in\mathcal{I}_k} \widetilde{\boldsymbol{\delta}}_\iota^{(k)} \in \mathbb{R}^D$
           $\mathcal{M}_k := \left\|\widetilde{\boldsymbol{f}}_k\right\|^{-2} \sum_{\iota\in\mathcal{I}_k} \ell_\iota^{(k)} \in [1,\infty)$
           $\widetilde{\boldsymbol{h}}_{k+1} := \widetilde{\boldsymbol{h}}_k + \lambda_k \mathcal{M}_k \widetilde{\boldsymbol{f}}_k \in \mathbb{R}^D$
         endif;
3:   if $k \equiv 1 \mod m$
       $\boldsymbol{S}_{k+1}$: an orthonormalized version of
             $\boldsymbol{K}_D(\widehat{\boldsymbol{R}}_k, \widehat{\boldsymbol{p}}_k) := [\widehat{\boldsymbol{p}}_k, \widehat{\boldsymbol{R}}_k\widehat{\boldsymbol{p}}_k, \cdots, \widehat{\boldsymbol{R}}_k^{D-1}\widehat{\boldsymbol{p}}_k] \in \mathbb{R}^{N\times D}$
     else
       $\boldsymbol{S}_{k+1} := \boldsymbol{S}_k$
     endif;
```

This implies that ρ should be determined according to the stochastic property of noise, thus $C_\iota^{(k)}(\rho)$ is referred to as *stochastic property set*. Under the assumption that $(n_k)_{k\in\mathbb{N}_0}$ is a zero-mean i.i.d. Gaussian random process with its variance σ_n^2, the random variable $\upsilon := \|\boldsymbol{n}_\iota\|^2$ obeys the χ^2 distribution of r degrees of freedom. Based on this fact, the following quantities are suggested ([29], Example 1): $\rho_1 := (r + \sqrt{2r})\sigma_n^2$, $\rho_2 := r\sigma_n^2$, or $\rho_3 := \max\{0, (r-2)\sigma_n^2\}$.

In the reduced-rank case, $\boldsymbol{h}_{\text{opt}} := P_{\mathcal{R}(\boldsymbol{S}_k)}^{(\boldsymbol{R})}(\boldsymbol{h}^*)$ for a given $\boldsymbol{S}_k$. In this case,

$$\boldsymbol{h}_{\text{opt}} \in C_\iota^{(k)}(\rho)$$
$$\Leftrightarrow \|\boldsymbol{e}_\iota(\boldsymbol{h}_{\text{opt}})\|^2 = \left\|\boldsymbol{U}_\iota^T\left(P_{\mathcal{R}(\boldsymbol{S}_k)}^{(\boldsymbol{R})}(\boldsymbol{h}^*) - \boldsymbol{h}^*\right) - \boldsymbol{n}_\iota\right\|^2 \leq \rho. \tag{10}$$

From (10), it is seen that, *in theory*, there are three terms to be taken into account in the design of ρ: the quadratic term $\|\boldsymbol{U}_\iota^T(P_{\mathcal{R}(\boldsymbol{S}_k)}^{(\boldsymbol{R})}(\boldsymbol{h}^*) - \boldsymbol{h}^*)\|^2$, the cross term $-2\langle \boldsymbol{U}_\iota^T(P_{\mathcal{R}(\boldsymbol{S}_k)}^{(\boldsymbol{R})}(\boldsymbol{h}^*) - \boldsymbol{h}^*), \boldsymbol{n}_\iota\rangle$, and the other quadratic term $\|\boldsymbol{n}_\iota\|^2$. Fortunately, however, the first two terms involve the approximation inaccuracy $P_{\mathcal{R}(\boldsymbol{S}_k)}^{(\boldsymbol{R})}(\boldsymbol{h}^*) - \boldsymbol{h}^*$, thus nearly diminish as the adaptation progresses. In practically noisy environments, those terms at steady state are negligibly small compared with the third term $\|\boldsymbol{n}_\iota\|^2$; note that $\|\boldsymbol{n}_\iota\|^2$ is *not* affected by the dimensionality reduction. Moreover, the algorithm is fairly robust in tuning the parameter ρ; this is verified by simulations in Section V. Therefore, *in practice*, it is enough to take into account only the statistics of noise (which is invariant under the dimensionality reduction) as in the full-rank case.[6]

Finally, replacing $\boldsymbol{h}$ in (8) by $\boldsymbol{S}_k\widetilde{\boldsymbol{h}}$, the stochastic property set in $\mathbb{R}^D$ is obtained as follows:

$$\widetilde{C}_\iota^{(k)}(\rho) := \left\{\widetilde{\boldsymbol{h}} \in \mathbb{R}^D : g_\iota^{(k)}(\widetilde{\boldsymbol{h}}) := \left\|\boldsymbol{e}_\iota^{(k)}(\widetilde{\boldsymbol{h}})\right\|^2 - \rho \leq 0\right\}, \quad \iota \leq k \in \mathbb{N}_0 \tag{11}$$

where $\boldsymbol{e}_\iota^{(k)}(\widetilde{\boldsymbol{h}}) := \boldsymbol{U}_\iota^T\boldsymbol{S}_k\widetilde{\boldsymbol{h}} - \boldsymbol{d}_\iota \in \mathbb{R}^r$, $\forall\widetilde{\boldsymbol{h}} \in \mathbb{R}^D$.

C. Proposed KRR-APSP Algorithm—Realization of Monotone Approaching

Now, we discuss how to realize the monotone approaching, in $\mathbb{R}^D$, to $\widetilde{\boldsymbol{h}}_{\text{opt}} := \boldsymbol{S}_k^T\boldsymbol{h}_{\text{opt}}$ (a counterpart of $\boldsymbol{h}_{\text{opt}}$ in $\mathbb{R}^D$) in a computationally efficient way; the monotone approaching to $\boldsymbol{h}_{\text{opt}}$ in $\mathbb{R}^N$ will be discussed in Section IV. Note that $\boldsymbol{h}_{\text{opt}} = \boldsymbol{S}_k\widetilde{\boldsymbol{h}}_{\text{opt}}$ because $\boldsymbol{h}_{\text{opt}} \in \mathcal{R}(\boldsymbol{S}_k)$. What we can do (in $\mathbb{R}^D$) is to approach every element in $\widetilde{C}_\iota^{(k)}(\rho)$ monotonically, and this is accomplished in an efficient way by means of *parallel subgradient projection*. Since $\boldsymbol{h}_{\text{opt}} \in C_\iota^{(k)}(\rho)$ ($\Leftrightarrow \widetilde{\boldsymbol{h}}_{\text{opt}} \in \widetilde{C}_\iota^{(k)}(\rho)$) is guaranteed with high probability, it is highly expected that monotone approaching to $\widetilde{\boldsymbol{h}}_{\text{opt}}$ is realized.

For the sake of fast convergence, we use multiple closed convex sets simultaneously at each iteration. Each convex set employed at kth iteration is indicated by each element of the control sequence [7] $\mathcal{I}_k \subset \{0, 1, \ldots, k\} \subset \mathbb{N}_0$. Namely, the collection of sets $\{C_\iota^{(k)}(\rho)\}_{\iota\in\mathcal{I}_k}$ is employed at kth iteration. A typical example is $\mathcal{I}_k := \{k, k-1, \ldots, k-q+1\}$ for $q \in \mathbb{N}$.

Since the projection onto $\widetilde{C}_\iota^{(k)}(\rho)$ is computationally expensive, we approximate it by the projection onto the simple closed half-space $\widetilde{H}_{\iota,k}^-(\widetilde{\boldsymbol{h}}_k) \supset \widetilde{C}_\iota^{(k)}(\rho)$ defined as

$$\widetilde{H}_{\iota,k}^-(\widetilde{\boldsymbol{h}}_k) := \left\{\widetilde{\boldsymbol{h}} \in \mathbb{R}^D : \left\langle \widetilde{\boldsymbol{h}} - \widetilde{\boldsymbol{h}}_k, \widetilde{\boldsymbol{s}}_\iota^{(k)}\right\rangle + g_\iota^{(k)}(\widetilde{\boldsymbol{h}}_k) \leq 0\right\}, \quad \iota \in \mathcal{I}_k,\ k \in \mathbb{N}_0. \tag{12}$$

where $\widetilde{\boldsymbol{s}}_\iota^{(k)} := \nabla g_\iota^{(k)}(\widetilde{\boldsymbol{h}}_k) := 2\boldsymbol{S}_k^T\boldsymbol{U}_\iota\boldsymbol{e}_\iota^{(k)}(\widetilde{\boldsymbol{h}}_k) \in \mathbb{R}^D$. An important property is $\widetilde{\boldsymbol{h}}_k \notin \widetilde{C}_\iota^{(k)}(\rho) \Rightarrow \widetilde{\boldsymbol{h}}_k \notin \widetilde{H}_{\iota,k}^-(\widetilde{\boldsymbol{h}}_k)$ ([29], Lemma 2), thus the boundary of $\widetilde{H}_{\iota,k}^-(\widetilde{\boldsymbol{h}}_k)$ is a separating hyperplane between $\widetilde{\boldsymbol{h}}_k$ and $\widetilde{C}_\iota^{(k)}(\rho)$. The projection of $\widetilde{\boldsymbol{h}}_k$ onto $\widetilde{H}_{\iota,k}^-(\widetilde{\boldsymbol{h}}_k)$ is given as

$$P_{\widetilde{H}_{\iota,k}^-(\widetilde{\boldsymbol{h}}_k)}(\widetilde{\boldsymbol{h}}_k) = \begin{cases} \widetilde{\boldsymbol{h}}_k, & \text{if } g_\iota^{(k)}(\widetilde{\boldsymbol{h}}_k) \leq 0 \\ \widetilde{\boldsymbol{h}}_k - \dfrac{g_\iota^{(k)}(\widetilde{\boldsymbol{h}}_k)}{\left\|\widetilde{\boldsymbol{s}}_\iota^{(k)}\right\|^2}\widetilde{\boldsymbol{s}}_\iota^{(k)}, & \text{otherwise} \end{cases} \tag{13}$$

[6] Another practical reason is as follows. Generally speaking, a too small value of ρ yields fast initial convergence with a possibly large steady-state error. The ρ designed based solely on the noise would be too small in the initial phase of adaptation, but becomes a reasonable value along with the progress of adaptation of $\widehat{\boldsymbol{R}}_k$ and $\widehat{\boldsymbol{p}}_k$. Therefore, such ρ will not cause a large error at steady state.

[7] One should not confuse *control sequence* with *training sequence*.

which is referred to as the *subgradient projection*[8] *relative to* $g_\iota^{(k)}$ (see Appendix A). We take a convex combination of the subgradient projections $P_{\widetilde{H}_{\iota,k}^{-}(\widetilde{\boldsymbol{h}}_k)}(\widetilde{\boldsymbol{h}}_k)$, $\iota \in \mathcal{I}_k$, with coefficients $w_\iota^{(k)} \in (0,1]$, $\iota \in \mathcal{I}_k$, $k \in \mathbb{N}_0$, satisfying $\sum_{\iota\in\mathcal{I}_k} w_\iota^{(k)} = 1$. The proposed *Krylov Reduced-Rank Adaptive Parallel Subgradient Projection (KRR-APSP)* algorithm is presented in what follows.

Given an arbitrary initial vector $\widetilde{\boldsymbol{h}}_0 \in \mathbb{R}^D$, the sequence $(\widetilde{\boldsymbol{h}}_k)_{k\in\mathbb{N}_0} \subset \mathbb{R}^D$ is inductively generated as follows. Given $\boldsymbol{h}_k$ and $\mathcal{I}_k$ at each time $k \in \mathbb{N}_0$, $\boldsymbol{h}_{k+1}$ is defined as

$$\widetilde{\boldsymbol{h}}_{k+1} = \widetilde{\boldsymbol{h}}_k + \lambda_k \mathcal{M}_k \left(\sum_{\iota\in\mathcal{I}_k} w_\iota^{(k)} P_{\widetilde{H}_{\iota,k}^{-}(\widetilde{\boldsymbol{h}}_k)}(\widetilde{\boldsymbol{h}}_k) - \widetilde{\boldsymbol{h}}_k \right) \quad (14)$$

where $\lambda_k \in [0,2]$, $\widetilde{H}_{\iota,k}^{-}(\widetilde{\boldsymbol{h}}_k)$ is defined as in (12), and the equation, shown at the bottom of the page.

The convexity of $\|\cdot\|^2$ implies $\mathcal{M}_k \geq 1$, and the use of $\mathcal{M}_k$ allows $\widetilde{\boldsymbol{h}}_k$ to step further than just taking a convex combination. In the literature, $\mu_k := \lambda_k \mathcal{M}_k$ is referred to as *extrapolation coefficient*. By [29, Prop. 1], we can immediately show the following monotone approaching property in $\mathbb{R}^D$:

$$\begin{aligned} &\|\widetilde{\boldsymbol{h}}_{k+1} - \widetilde{\boldsymbol{h}}_k^*\| \leq \|\widetilde{\boldsymbol{h}}_k - \widetilde{\boldsymbol{h}}_k^*\|, \\ &\qquad \forall \widetilde{\boldsymbol{h}}_k^* \in \bigcap_{\iota\in\mathcal{I}_k} \widetilde{H}_{\iota,k}^{-}(\widetilde{\boldsymbol{h}}_k) \supset \bigcap_{\iota\in\mathcal{I}_k} \widetilde{C}_\iota^{(k)}(\rho), \quad \forall k \in \mathbb{N}_0. \end{aligned} \quad (15)$$

Efficient implementation of the proposed algorithm is given in Table I. For computational efficiency, we introduce the parameter m to control the frequency of updating $\boldsymbol{S}_k$. We mention that, although the condition for updating $\widetilde{\boldsymbol{\delta}}_\iota^{(k)}$ is similar to the one used in *the set-membership affine projection algorithm* [48], the major differences are that i) the update is based on the subgradient projection, ii) multiple closed convex sets are employed at each iteration (each set is indicated by an element of $\mathcal{I}_k$), and iii) no matrix inversion is required.

D. On the Parameters Used in KRR-APAP

We summarize below the parameters used in the proposed algorithm:

- $\lambda_k (k \in \mathbb{N}_0)$: step size;
- $w_\iota^{(k)} (\iota \in \mathcal{I}_k, k \in \mathbb{N}_0)$: weight assigned to $C_\iota^{(k)}(\rho)$ at kth iteration;
- D: Krylov subspace dimension;
- m: the frequency of updating $\boldsymbol{S}_k$;
- q: the number of projections computed at each iteration;
- r: the dimension of the orthogonal complement of the underlying subspace of $C_\iota^{(k)}(0)$ [see the definition of $\boldsymbol{U}_k$, and $\boldsymbol{d}_k$ before (8)];
- ρ: the error bound (controlling the "volume" of $C_\iota^{(k)}(\rho)$).

The algorithm is very robust against the choice of these parameters, although the optimal choice depends on problems. Nevertheless, a general remark is given below on the parameter choice.

Remark 2 (On the Choice of Parameters): Similarly to the normalized least mean square (NLMS) algorithm, λ_k has a function to balance the speed of convergence and the steady-state performance. However, any choice of $\lambda_k \in [0,2]$ never causes filter-divergence and no delicate tuning is necessary.

A simple and acceptable design of $w_\iota^{(k)}$ is the uniform ones, and more strategic design has been presented in [32], [36]. The choice of D affects the convergence speed and the approximation accuracy (thus the achievable MSE and system mismatch); too small D results in fast convergence but large MSE and system mismatch, and vice versa. Fortunately, however, $D = 4, 5$ yields fairly small MSE and system mismatch (and fast convergence) in a variety of situations. As for m, the performance is insensitive to its choice; though, in highly dynamic environments, m should not be too large in order for the filter to be able to track the unknown system on a reasonably updated subspace.

The values of r and q determine the degree of data reusing, which has a function to increase the rate of convergence. In our experiments, fixing $r := 1$ and increasing q up to D lead to significant acceleration of convergence speed with little degradation of steady-state performance (see Section V).

Regarding the choice of ρ, a discussion was given already in Section III-B. For $r := 1$, in particular, we have $\rho_3 := \max\{0, (r-2)\sigma_n^2\} = 0$ and an arbitrary choice of $\rho \geq 0$ never causes instability (although $\rho \gg \rho_1$ makes the set $\widetilde{C}_\iota^{(k)}(\rho)$ too large and results in slow convergence).

The tracking property and the computational complexity of the proposed algorithm are discussed in the following subsection.

E. Tracking Property

As explained in Section II, an algorithm that tracks $P^{(\boldsymbol{R})}_{\mathcal{K}_D(\widehat{\boldsymbol{R}}_k, \widehat{\boldsymbol{p}}_k)}(\boldsymbol{h}^*)$ is expected to enjoy better tracking capability than the existing Krylov-subspace-based reduced-rank methods. In this subsection, we first show that the proposed algorithm (or the vector $\boldsymbol{h}_k (= \boldsymbol{S}_k \widetilde{\boldsymbol{h}}_k)$, $k \in \mathbb{N}_0$, generated by the proposed algorithm) has such a property for its simplest

[8] Although the function $g_\iota^{(k)}$ is differentiable, the subgradient projection can be defined also for non-differentiable functions. Note that $\operatorname{lev}_{\leq 0} g_\iota^{(k)} := \{\widetilde{\boldsymbol{h}} \in \mathbb{R}^D : g_\iota^{(k)}(\widetilde{\boldsymbol{h}}) \leq 0\} \neq \emptyset$.

$$\mathcal{M}_k := \begin{cases} 1 & \text{if } g_\iota^{(k)}(\widetilde{\boldsymbol{h}}_k) \leq 0, \quad \forall \iota \in \mathcal{I}_k, \\ \dfrac{\sum_{\iota\in\mathcal{I}_k} w_\iota^{(k)} \left\| P_{\widetilde{H}_{\iota,k}^{-}(\widetilde{\boldsymbol{h}}_k)}(\widetilde{\boldsymbol{h}}_k) - \widetilde{\boldsymbol{h}}_k \right\|^2}{\left\| \sum_{\iota\in\mathcal{I}_k} w_\iota^{(k)} P_{\widetilde{H}_{\iota,k}^{-}(\widetilde{\boldsymbol{h}}_k)}(\widetilde{\boldsymbol{h}}_k) - \widetilde{\boldsymbol{h}}_k \right\|^2} & \text{otherwise.} \end{cases}$$

case: $r = 1$, $\rho = 0$, $\mathcal{I}_k = \{k\}$ (i.e., $q = 1$). In this case, the proposed algorithm is reduced to

$$\widetilde{\boldsymbol{h}}_{k+1} = \widetilde{\boldsymbol{h}}_k + \bar{\lambda}_k \frac{d_k - \widetilde{\boldsymbol{h}}_k^T \widetilde{\boldsymbol{u}}_k}{\|\widetilde{\boldsymbol{u}}_k\|^2} \widetilde{\boldsymbol{u}}_k \tag{16}$$

where $\bar{\lambda}_k := \lambda_k/2 \in [0,1]$. The update equation in (16) is nothing but the NLMS algorithm. (It should be mentioned that the step-size range of $\bar{\lambda}_k$ is a half of that of NLMS.) Thus, (16) is a stochastic gradient algorithm for the following problem:

$$\min_{\widetilde{\boldsymbol{h}} \in \mathbb{R}^D} \mathrm{E}\{(d_k - \widetilde{\boldsymbol{h}}^T \widetilde{\boldsymbol{u}}_k)^2\}. \tag{17}$$

This implies that $\widetilde{\boldsymbol{h}}_k$ generated by (16) tracks the minimizer of (17); for details about the tracking performance of NLMS, see [49] and the references therein. Hence, noting that $\widetilde{\boldsymbol{u}}_k = \boldsymbol{S}_k^T \boldsymbol{u}_k$, it is seen that $\boldsymbol{h}_k (:= \boldsymbol{S}_k \widetilde{\boldsymbol{h}}_k)$ tracks the solution to the following problem [which is equivalent to (17)]:

$$\min_{\boldsymbol{h} \in \mathcal{R}(\boldsymbol{S}_k)} \mathrm{E}\{(d_k - \boldsymbol{h}^T \boldsymbol{u}_k)^2\}. \tag{18}$$

Referring to (2) and (5), the minimizer of (18) is $P^{(\boldsymbol{R})}_{\mathcal{K}_D(\widehat{\boldsymbol{R}}_k, \widehat{\boldsymbol{p}}_k)}(\boldsymbol{h}^*)$. This verifies that $\boldsymbol{h}_k (= \boldsymbol{S}_k \widetilde{\boldsymbol{h}}_k)$ generated by (16) tracks $P^{(\boldsymbol{R})}_{\mathcal{K}_D(\widehat{\boldsymbol{R}}_k, \widehat{\boldsymbol{p}}_k)}(\boldsymbol{h}^*)$.

F. Computational Complexity

Now, let us move to the discussion about the computational complexity (i.e., the number of multiplications per iteration) of the proposed algorithm. For simplicity, we let $\mathcal{I}_k := \{k, k-1, \ldots, k-q+1\}$, which is used in Section V. We assume that, given $\widehat{\boldsymbol{R}}_k$ and $\widehat{\boldsymbol{p}}_k$, the complexity to construct the matrix $\boldsymbol{S}_k$ is the same as that of CGRRF[9]. As $\boldsymbol{S}_k$ is computed every m iterations (see Table I), the average complexity for computing $\boldsymbol{S}_k$ is $(D-1)N^2/m + (5D-4)N/m + 2(D-1)/m$.

What about the complexity to update $\widehat{\boldsymbol{R}}_k$ and $\widehat{\boldsymbol{p}}_k$? In general, $\widehat{\boldsymbol{R}}_k$ and $\widehat{\boldsymbol{p}}_k$ can be computed recursively as $\widehat{\boldsymbol{R}}_{k+1} := \gamma \widehat{\boldsymbol{R}}_k + \boldsymbol{u}_k \boldsymbol{u}_k^T$ and $\widehat{\boldsymbol{p}}_{k+1} := \gamma \widehat{\boldsymbol{p}}_k + d_k \boldsymbol{u}_k$, $k \in \mathbb{N}_0$, where $\gamma \in (0,1)$ is the forgetting factor. Fortunately, because of the symmetry of $\boldsymbol{R}$, only the upper (or lower) triangular portion of $\widehat{\boldsymbol{R}}_k$ needs to be computed, resulting in the complexity $\xi := N^2 + 3N$. Moreover, further computational reduction is possible when $\boldsymbol{R}$ has *a Toeplitz structure*; for the system model considered in this study, it is known that $\boldsymbol{R}$ is Toeplitz, provided that the input process is stationary (Note: It does not matter whether $\boldsymbol{h}^*$ changes dynamically). In this case, it is sufficient to estimate $E\{u_k \boldsymbol{u}_k\} \in \mathbb{R}^N$ instead of $\boldsymbol{R}$, which can be done by $\widehat{\boldsymbol{r}}_{k+1} := \gamma \widehat{\boldsymbol{r}}_k + u_k \boldsymbol{u}_k$, $k \in \mathbb{N}_0$, leading to the complexity reduced to $\xi := 4N$. For instance, speech signals are generally nonstationary, but it is well-known that the signals can be assumed to be stationary during a short period. Therefore, within a short period, we can assume that $\boldsymbol{R}$ is a Toeplitz matrix so that $\widehat{\boldsymbol{R}}_k$ and $\widehat{\boldsymbol{p}}_k$ can be updated with the complexity $4N$. Regarding the choice of γ, its value should be close to one (a reasonable choice is $\gamma \approx 0.999$) for two reasons. One is that a small γ (such as $\gamma \approx 0.99$) yields inaccurate estimates of $\boldsymbol{R}$ and $\boldsymbol{p}$ at steady state, resulting in deterioration of steady-state performance. The other is that a smaller γ makes adaptation of $\widehat{\boldsymbol{R}}_k$ and $\widehat{\boldsymbol{p}}_k$ faster, meaning that the estimation becomes more sensitive to disturbance such as impulsive noise (see Section V).

The rest is the complexity for the filter update. One of the distinguished advantages of the APSP algorithm is its *inherently parallel structure* [29], [32], [50]–[53]. We start by considering the case where only a single processor is available. Because the matrices $(\boldsymbol{U}_\iota)_{\iota \in \mathcal{I}_k}$, used at time k, have only $q + r - 1$ distinct column vectors $(\boldsymbol{u}_k, \boldsymbol{u}_{k-1}, \ldots, \boldsymbol{u}_{k-q-r+2})$, the complexity to compute $\boldsymbol{U}_\iota^{(k)}$ for all $\iota \in \mathcal{I}_k$ is $(q + r - 1)DN$. Fortunately, however, this is only required when $\boldsymbol{S}_k$ is updated (every m iterations), and, when $\boldsymbol{S}_k$ is *not* updated, only the first column of $\boldsymbol{U}_k^{(k)}$ (i.e., $\boldsymbol{S}_k^T \boldsymbol{u}_k$) should be computed. This is because, when $\boldsymbol{S}_k$ is *not* updated, it holds that $\boldsymbol{U}_\iota^{(k)} = \boldsymbol{U}_\iota^{(k-1)}$ for $\iota = \mathcal{I}_k \setminus \{k\}$ and $[\boldsymbol{U}_k^{(k)}]_{2:r} = [\boldsymbol{U}_{k-1}^{(k-1)}]_{1:r-1}$, where $[\boldsymbol{A}]_{a:b}$ designates the submatrix of $\boldsymbol{A}$ consisting of the ath to bth column vectors. Thus, the average complexity for $\boldsymbol{U}_\iota^{(k)}$ is $[(q + r - 1)DN + (m-1)DN]/m$. For the same reason as $(\boldsymbol{U}_\iota)_{\iota \in \mathcal{I}_k}$, the matrices $(\boldsymbol{U}_\iota^{(k)})_{\iota \in \mathcal{I}_k}$ also have only $q + r - 1$ distinct column vectors, hence the complexity to compute $\boldsymbol{e}_\iota^{(k)}$ and $\boldsymbol{a}_\iota^{(k)}$ is no more than $2(q + r - 1)D$. Overall, the total complexity for the filter update is $\alpha(q,r,m)DN + (4q + 2r)D + (r + 7)q + 2$, where $\alpha(q,r,m) := (q+r+m-2)/m$. If we set, for instance, $D = 5$, $m = 10$, $r = 1$, and $q = 5$ (which are used in Section V-C), the complexity for the filter update is $7N + 152$.

Finally, we consider the case where q parallel processors are available. In this case, the computation of the variables corresponding to each $\iota \in \mathcal{I}_k$ is naturally assigned to each processor. We consider the complexity imposed on each processor at each iteration. The complexity to compute $\boldsymbol{U}_\iota^{(k)}$ is rDN, when $\boldsymbol{S}_k$ is updated, and DN, when $\boldsymbol{S}_k$ is *not* updated. The average complexity is thus $\beta(r,m)DN$, where $\beta(r,m) := (r + m - 1)/m$. Overall, the per-processor complexity for the filter update is $\beta(r,m)DN + (2r+4)D + r + 9$. For $D = 5$, $m = 10$, $r = 1$, and an arbitrary q, the complexity for the filter update is $5N + 40$.

In Table II, the overall complexity of the proposed algorithm is summarized with those of the NLMS algorithm, the recursive least squares (RLS) algorithm [40, Table 9.1], and CGRRF [20]; we assume for fairness that CGRRF updates the filter every m iterations. Fig. 4 plots the number of multiplications against the filter length N for $D = 5$, $m = 10$, $r = 1$, and $q = 5$ (which are used in Section V-C). We can see that the complexity of the proposed algorithm is much lower than that of RLS (due to the factor m), and marginally higher than that of CGRRF; in particular, for a large value of N, the difference between the proposed and CGRRF methods is negligible. Moreover, compared with NLMS, the proposed algorithm requires higher complexity for realizing better performance. However, the difference can be significantly reduced by increasing m; in our experiments, the use of $m = 100$ gives almost the same performance as the use of $m = 10$. It should be mentioned that the difference (in computational complexity) between CGRRF and KRR-APSP can be further reduced by taking into account the update rate of the

[9] The Lanczos method, which is essentially equivalent to the CG method [41], can also be used for constructing $\boldsymbol{S}_k$.

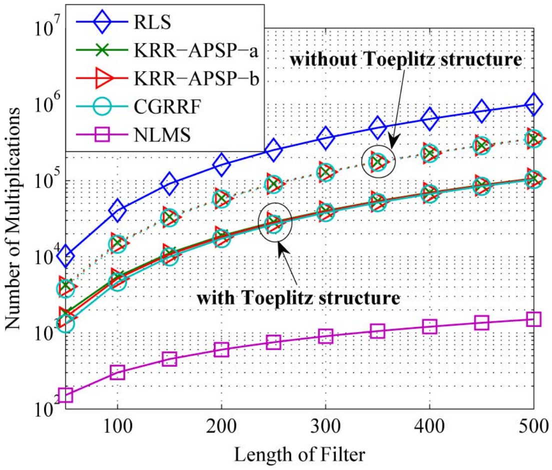


Fig. 4. Complexities of the conventional methods and the proposed algorithm with (a) single processor and (b) q processors.

TABLE II
COMPUTATIONAL COMPLEXITIES OF ALGORITHMS. $\alpha(q,r,m) := (q+r+m-2)/m$, $\beta(r,m) := (r+m-1)/m$, AND $\xi := N^2+3N$ (IN GENERAL) OR $\xi := 4N$ (WHEN THE TOEPLITZ STRUCTURE OF $\boldsymbol{R}$ IS EXPLOITED)

Algorithm	Number of multiplications per iteration
NLMS	$3N+2$
RLS	$4N^2+4N+1$
CGRRF	$(D-1)N^2/m+[(5D-4)/m]N+\xi$ $+2(D-1)$
KRR-APSP (single processor)	$(D-1)N^2/m$ $+[(5D-4)/m]N+\xi+\alpha(q,r,m)DN$ $+2(D-1)+(4q+2r)D+(r+7)q+2$
KRR-APSP (q processors)	$(D-1)N^2/m$ $+[(5D-4)/m]N+\xi+\beta(r,m)DN$ $+2(D-1)+(2r+4)D+r+9$

vector $\widetilde{\boldsymbol{h}}_k$ (i.e., the rate in which it happens that $\|\boldsymbol{e}_t^{(k)}\|^2 \leq \rho$). For a ρ chosen appropriately, the update rate is typically less than 10%.

In conclusion, the proposed algorithm is highly expected to realize, with comparable computational complexity, superior tracking performance to the existing Krylov-subspace-based reduced-rank methods, as will be verified by simulations in Section V. Moreover, the algorithm has a fault tolerance nature thanks to its inherently parallel structure; i.e., even if some of the engaged concurrent processors are crashed, the lack of information from the crashed processors would *not* cause any serious degradation in performance. This is because the direction of update is determined by taking into account all the directions suggested by each input data vector little by little. We finally mention that the Krylov-proportionate adaptive filtering, an alternative approach based on the Krylov subspace, has been proposed in [54]; it utilizes the Krylov subspace for *sparsifying* the optimal filter and enjoys fast convergence, optimal (full-rank) steady-state performance, and $O(N)$ complexity per iteration, whereas its initial convergence-speed is not as fast as the reduced-rank approaches. Which is preferred between the proposed approach and the one in [54] would be application-dependent.

G. *Robustness Issue Against Impulsive Noise*

Impulsive noise generally causes performance deterioration of adaptive algorithms. Several techniques have already been proposed in the literature; e.g., an approach based on *robust statistics* [55] (see [56] for the robust statistics itself), an approach to constrain the energy of the filter update [57], etc. Our approach to the robustness issue is based on the idea in [57].

Given a $\delta_{\boldsymbol{h}} > 0$, we impose the following constraint on the filter:

$$\|\widetilde{\boldsymbol{h}}_{k+1} - \widetilde{\boldsymbol{h}}_k\|^2 \leq \delta_{\boldsymbol{h}}. \tag{19}$$

This constraint simply changes the filter update equation into the following:

$$\widetilde{\boldsymbol{h}}_{k+1} = \widetilde{\boldsymbol{h}}_k + \bar{\lambda}_k \mathcal{M}_k \widetilde{\boldsymbol{f}}_k \tag{20}$$

where $\bar{\lambda}_k := \min\{\lambda_k, (\sqrt{\delta_{\boldsymbol{h}}})/(\mathcal{M}_k \|\widetilde{\boldsymbol{f}}_k\|)\} \in (0, \lambda_k]$. One can see that this is a sort of variable step size method. Although it is possible to adapt $\delta_{\boldsymbol{h}}$ in the same way as in [57], we use a constant $\delta_{\boldsymbol{h}}$ for simplicity. We finally mention that the extra computational cost for $\bar{\lambda}_k$ is negligibly low because the cost for $\|\widetilde{\boldsymbol{f}}_k\|$ is already counted in the computation of $\mathcal{M}_k$ (see Table I).

In the following section, we present an analysis of the proposed algorithm; the analysis is valid for any $\lambda_k \in [0, 2]$ (thus no matter if we use the above approach).

IV. ANALYSIS OF THE PROPOSED ALGORITHM

In adaptive filtering or learning, the observed measurements are mostly corrupted by noise and the environments are nonstationary in many scenarios. Under such uncertain situations, it is difficult (or nearly impossible) to guarantee that the adaptive filter approaches the optimal one monotonically at every iteration. Thus, a meaningful and realistic property desired for an adaptive algorithm would be to approach every point in an appropriately designed set of filtering vectors monotonically at each iteration. How can such a set, say $\Omega_k \subset \mathbb{R}^N$, be designed?

In our analysis, we let $\Theta_k : \mathbb{R}^N \to [0, \infty)$ be a (continuous and convex) objective function, and Ω_k is defined as a set of all the vectors that achieve the infimum of Θ_k over a certain constraint set. (The constraint is associated with the requirements that the filter should lie in the Krylov subspace.) Then, the desired *monotone approximation* property is expressed as follows[10]:

$$\left\|\boldsymbol{h}_{k+1} - \boldsymbol{h}^*_{(k)}\right\| \leq \left\|\boldsymbol{h}_k - \boldsymbol{h}^*_{(k)}\right\|, \quad \forall \boldsymbol{h}^*_{(k)} \in \Omega_k,\ k \in \mathbb{N}_0. \tag{21}$$

We stress that (21) insists that the monotonicity holds *for all the elements of* Ω_k.

What about "optimality" in terms of the objective function Θ_k? Is it possible to prove "optimality" in any sense? As you might notice, the objective function Θ_k depends on k. Namely, what we should "minimize" is *not* a fixed objective function but is a sequence of objective functions $(\Theta_k)_{k\in\mathbb{N}_0}$. This is the major difference from the normal optimization problems, and this formulation naturally fits the adaptive signal processing because the objective function should be changing in conjunction with changing environments. Thus, a meaningful "optimality"

[10] To ensure (21), *closedness and convexity* of Ω_k are essential.

to show would be that $(\boldsymbol{h}_k)_{k\in\mathbb{N}_0}$ minimizes $(\Theta_k)_{k\in\mathbb{N}_0}$ asymptotically; i.e.,

$$\lim_{k\to\infty}\Theta_k(\boldsymbol{h}_k)=0 \tag{22}$$

which is called *asymptotic optimality* [38], [39].

The goal of this section is to prove that the proposed algorithm enjoys the two desired properties (21) and (22). To this end, we firstly build, with the objective function Θ_k, a unified framework named *reduced-rank adaptive projected subgradient method (R-APSM)*, and derive the proposed algorithm from R-APSM with a specific design of Θ_k. We then prove that R-APSM, including the proposed algorithm as its special case, has the desired properties under some mild conditions.

A. Alternative Derivation of the Proposed Algorithm

Recall here that $\boldsymbol{h}_k$ is forced to lie in $\mathcal{R}(\boldsymbol{S}_k)$ at each iteration $k\in\mathbb{N}_0$. For an analysis of the proposed algorithm, we define

$$\boldsymbol{\Phi}_k := \boldsymbol{S}_{k+1}\boldsymbol{S}_k^T \in \mathbb{R}^{N\times N}. \tag{23}$$

Given an arbitrary $\boldsymbol{h}_0\in\mathbb{R}^N$ and a sequence of continuous convex objective functions $\Theta_k:\mathbb{R}^N\to[0,\infty)$, $k\in\mathbb{N}_0$, R-APSM[11] generates a sequence $(\boldsymbol{h}_k)_{k\in\mathbb{N}_0}\subset\mathbb{R}^N$ by

$$\boldsymbol{h}_{k+1} := \begin{cases} \boldsymbol{\Phi}_k\left[\boldsymbol{h}_k - \lambda_k\frac{\Theta_k(\boldsymbol{h}_k)}{\|\Theta_k'(\boldsymbol{h}_k)\|^2}\Theta_k'(\boldsymbol{h}_k)\right] & \text{if } \Theta_k'(\boldsymbol{h}_k)\neq\boldsymbol{0}\\ \boldsymbol{\Phi}_k\boldsymbol{h}_k & \text{otherwise}\end{cases} \tag{24}$$

where $\lambda_k\in[0,2]$, $k\in\mathbb{N}_0$, and $\Theta_k'(\boldsymbol{h}_k)\in\partial\Theta_k(\boldsymbol{h}_k)$ is a *subgradient* of Θ_k at $\boldsymbol{h}_k$ (see Appendix A).

Suppose that $\mathrm{lev}_{\leq 0}\Theta_k := \{\boldsymbol{h}\in\mathbb{R}^N : \Theta_k(\boldsymbol{h})\leq 0\}\neq\emptyset$ ($\Leftrightarrow \min_{\boldsymbol{h}\in\mathbb{R}^N}\Theta_k(\boldsymbol{h})=0$). Then, removing $\boldsymbol{\Phi}_k$, (24) for $\lambda_k=1$ is the subgradient projection relative to Θ_k [cf. (13)], which is denoted by $T_{\mathrm{sp}(\Theta_k)}(\boldsymbol{h}_k)$ (see Fig. 5). The update equation in (24) can be expressed as

$$\boldsymbol{h}_{k+1} := \boldsymbol{\Phi}_k\left[\boldsymbol{h}_k + \lambda_k\left(T_{\mathrm{sp}(\Theta_k)}(\boldsymbol{h}_k)-\boldsymbol{h}_k\right)\right]. \tag{25}$$

Noticing that the thick arrow in Fig. 5 expresses $T_{\mathrm{sp}(\Theta_k)}(\boldsymbol{h}_k)-\boldsymbol{h}_k$, the figure with (25) provides a geometric interpretation of R-APSM (except for $\boldsymbol{\Phi}_k$).

Let us now derive the proposed algorithm from R-APSM. Let $\mathcal{I}_k$ be the control sequence, and $w_\iota^{(k)}\in(0,1]$, $\iota\in\mathcal{I}_k$, $k\in\mathbb{N}_0$, the weight, both of which are defined in the same way as in Section III-C. An outer approximating closed half-space $H_\iota^-(\boldsymbol{h}_k)\supset C_\iota^{(k)}(\rho)$ is defined as [see (8)]

$$H_\iota^-(\boldsymbol{h}_k) := \left\{\boldsymbol{h}\in\mathbb{R}^N : \left\langle\boldsymbol{h}-\boldsymbol{h}_k,\boldsymbol{s}_\iota^{(k)}\right\rangle + g_\iota(\boldsymbol{h}_k)\leq 0\right\},\quad \iota\in\mathcal{I}_k,\ k\in\mathbb{N}_0$$

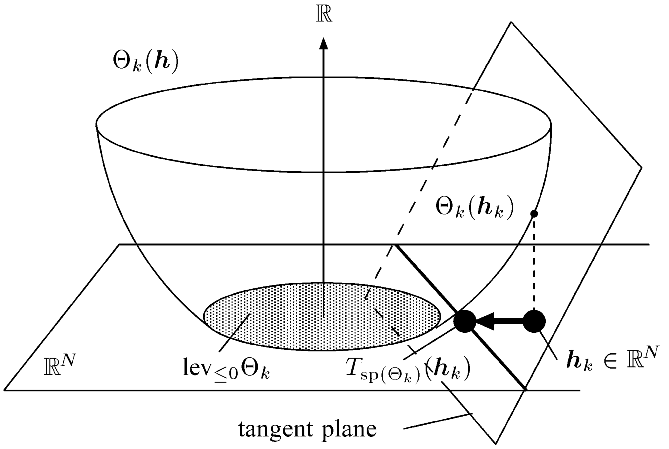

ℝ
$\Theta_k(\boldsymbol{h})$
$\Theta_k(\boldsymbol{h}_k)$
$\mathbb{R}^N$
$\mathrm{lev}_{\leq 0}\Theta_k$
$T_{\mathrm{sp}(\Theta_k)}(\boldsymbol{h}_k)$
$\boldsymbol{h}_k\in\mathbb{R}^N$
tangent plane

Fig. 5. Geometric interpretation of the subgradient projection $T_{\mathrm{sp}(\Theta_k)}(\boldsymbol{h}_k)$ when $\mathrm{lev}_{\leq 0}\Theta_k (:= \{\boldsymbol{h}\in\mathbb{R}^N : \Theta_k(\boldsymbol{h})\leq 0\})\neq\emptyset$.

where $\boldsymbol{s}_\iota^{(k)} := \boldsymbol{\nabla} g_\iota(\boldsymbol{h}_k) := 2\boldsymbol{U}_\iota\boldsymbol{e}_\iota(\boldsymbol{h}_k)\subset\mathbb{R}^N$. Because

1) $H_\iota^-(\boldsymbol{h}_k)$, $\iota\in\mathcal{I}_k$, contains favorable vectors because of the definition of $C_\iota^{(k)}(\rho)$, and
2) $\boldsymbol{h}_k$ should lie in $\mathcal{R}(\boldsymbol{S}_k)=\mathcal{K}_D(\widehat{\boldsymbol{R}}_k,\widehat{\boldsymbol{p}}_k)$,

the distance to $H_\iota^-(\boldsymbol{h}_k)\cap\mathcal{R}(\boldsymbol{S}_k)$ is a natural candidate of objective function. Moreover, for assigning a larger weight to a farther set, the weight $d(\boldsymbol{h}_k, H_\iota^-(\boldsymbol{h}_k)\cap\mathcal{R}(\boldsymbol{S}_k))$ is given to the distance function $d(\boldsymbol{h}, H_\iota^-(\boldsymbol{h}_k)\cap\mathcal{R}(\boldsymbol{S}_k))$. With a normalization factor $L_k := \sum_{\iota\in\mathcal{I}_k} w_\iota^{(k)} d(\boldsymbol{h}_k, H_\iota^-(\boldsymbol{h}_k)\cap\mathcal{R}(\boldsymbol{S}_k))$, the resulting objective function is given as follows:

$$\Theta_k(\boldsymbol{h}) := \begin{cases}\frac{1}{L_k}\sum_{\iota\in\mathcal{I}_k} w_\iota^{(k)} d(\boldsymbol{h}_k, H_\iota^-(\boldsymbol{h}_k)\cap\mathcal{R}(\boldsymbol{S}_k))\\ d(\boldsymbol{h}, H_\iota^-(\boldsymbol{h}_k)\cap\mathcal{R}(\boldsymbol{S}_k)) \text{ if } L_k\neq 0,\\ 0 \text{ otherwise.}\end{cases} \tag{26}$$

An application of R-APSM to $\Theta_k(\boldsymbol{h})$ in (26) yields (cf. [39])

$$\boldsymbol{h}_{k+1} = \boldsymbol{\Phi}_k\left[\boldsymbol{h}_k + \lambda_k\mathcal{M}_k\left(\sum_{\iota\in\mathcal{I}_k} w_\iota^{(k)} P_{H_\iota^-(\boldsymbol{h}_k)\cap\mathcal{R}(\boldsymbol{S}_k)}(\boldsymbol{h}_k) - \boldsymbol{h}_k\right)\right] \tag{27}$$

where $\lambda_k\in[0,2]$, $k\in\mathbb{N}_0$, and the equation, shown at the bottom of the page. Noticing $\boldsymbol{h}_k\in\mathcal{R}(\boldsymbol{S}_k)$ and defining $\boldsymbol{Q}_k := \boldsymbol{S}_k\boldsymbol{S}_k^T$, the projection of $\boldsymbol{h}_k$ onto $H_\iota^-(\boldsymbol{h}_k)\cap\mathcal{R}(\boldsymbol{S}_k)$ is given as follows:

$$P_{H_\iota^-(\boldsymbol{h}_k)\cap\mathcal{R}(\boldsymbol{S}_k)}(\boldsymbol{h}_k) = \begin{cases}\boldsymbol{h}_k & \text{if } g_\iota(\boldsymbol{h}_k)\leq 0\\ \boldsymbol{h}_k - \frac{g_\iota(\boldsymbol{h}_k)}{\left\|\boldsymbol{Q}_k\boldsymbol{s}_\iota^{(k)}\right\|^2}\boldsymbol{Q}_k\boldsymbol{s}_\iota^{(k)} & \text{otherwise.}\end{cases} \tag{28}$$

$$\mathcal{M}_k := \begin{cases} 1 & \text{if } g_\iota(\boldsymbol{h}_k)\leq 0,\quad \forall\iota\in\mathcal{I}_k,\\ \frac{\sum_{\iota\in\mathcal{I}_k} w_\iota^{(k)}\left\|P_{H_\iota^-(\boldsymbol{h}_k)\cap\mathcal{R}(\boldsymbol{S}_k)}(\boldsymbol{h}_k)-\boldsymbol{h}_k\right\|^2}{\left\|\sum_{\iota\in\mathcal{I}_k} w_\iota^{(k)} P_{H_\iota^-(\boldsymbol{h}_k)\cap\mathcal{R}(\boldsymbol{S}_k)}(\boldsymbol{h}_k)-\boldsymbol{h}_k\right\|^2} & \text{otherwise.}\end{cases}$$

[11] The original APSM [38], [39] is obtained by replacing $\boldsymbol{\Phi}_k$ in (24) by a projection operator onto a closed convex set of an absolute constraint.

Letting $\boldsymbol{h}_k = \boldsymbol{S}_k\widetilde{\boldsymbol{h}}_k$, we obtain $\boldsymbol{e}_\iota(\boldsymbol{h}_k) = \boldsymbol{e}_\iota^{(k)}(\widetilde{\boldsymbol{h}}_k)$, $g_\iota(\boldsymbol{h}_k) = g_\iota^{(k)}(\widetilde{\boldsymbol{h}}_k)$, and $\boldsymbol{S}_k^T\boldsymbol{s}_\iota^{(k)} = \widetilde{\boldsymbol{s}}_\iota^{(k)}$, from which and $P_{H_\iota^-(\boldsymbol{h}_k)\cap\mathcal{R}(\boldsymbol{S}_k)}(\boldsymbol{h}_k) \in \mathcal{R}(\boldsymbol{S}_k)$ we can verify

$$P_{H_\iota^-(\boldsymbol{h}_k)\cap\mathcal{R}(\boldsymbol{S}_k)}(\boldsymbol{h}_k) = \boldsymbol{S}_k P_{\widetilde{H}_{\iota,k}^-(\widetilde{\boldsymbol{h}}_k)}(\widetilde{\boldsymbol{h}}_k). \tag{29}$$

Substituting (29) and $\boldsymbol{h}_k = \boldsymbol{S}_k\widetilde{\boldsymbol{h}}_k$ into (27), and left-multiplying both sides of (27) by $\boldsymbol{S}_k^T$, we obtain the proposed algorithm. Taking a look at the update equation in (27), it is seen that it has the same form as the *linearly constrained adaptive filtering algorithm* [31] except for the mapping $\boldsymbol{\Phi}_k$ from $\mathcal{R}(\boldsymbol{S}_k)$ to $\mathcal{R}(\boldsymbol{S}_{k+1})$. Hence, viewing the behavior of the proposed algorithm in $\mathbb{R}^N$, *it performs parallel subgradient projection in a series of (constraint) Krylov subspaces* $(\mathcal{R}(\boldsymbol{S}_k))_{k\in\mathbb{N}_0}$.

B. Analysis of R-APSM

We prove that the sequence $(\boldsymbol{h}_k)_{k\in\mathbb{N}_0}$ generated by R-APSM satisfies the desired properties (21) and (22). In the analysis, *the fixed point set* of the 'mapping' $\boldsymbol{\Phi}_k(:= \boldsymbol{S}_{k+1}\boldsymbol{S}_k^T) : \mathbb{R}^N \to \mathcal{R}(\boldsymbol{S}_{k+1}), \boldsymbol{a} \mapsto \boldsymbol{\Phi}_k\boldsymbol{a}$, plays an important role. *What is the fixed point set?* Given a mapping $T : \mathbb{R}^N \to \mathbb{R}^N$, a point $\boldsymbol{x} \in \mathbb{R}^N$ satisfying $T(\boldsymbol{x}) = \boldsymbol{x}$ is called a *fixed point* of T. Moreover, the set of all such points, i.e., the set $\mathrm{Fix}(T) := \{\boldsymbol{x} \in \mathbb{R}^N : T(\boldsymbol{x}) = \boldsymbol{x}\}$, is called the *fixed point set* of T. The set $\mathrm{Fix}(\boldsymbol{\Phi}_k)$ is characterized as below.

Proposition 1 (Characterizations of $\mathrm{Fix}(\boldsymbol{\Phi}_k)$*):*

a) $\boldsymbol{0} \in \mathrm{Fix}(\boldsymbol{\Phi}_k)$.

b) $\mathrm{Fix}(\boldsymbol{\Phi}_k) \subset \mathcal{R}(\boldsymbol{S}_k) \cap \mathcal{R}(\boldsymbol{S}_{k+1})$.

c)

$$\mathrm{Fix}(\boldsymbol{\Phi}_k) = \left\{\boldsymbol{S}_k\widetilde{\boldsymbol{z}} = \boldsymbol{S}_{k+1}\widetilde{\boldsymbol{z}} : \widetilde{\boldsymbol{z}} \in \mathrm{Fix}\left(\boldsymbol{S}_k^T\boldsymbol{S}_{k+1}\right) \subset \mathbb{R}^D\right\} \tag{30}$$

and

$$\mathrm{Fix}\left(\boldsymbol{S}_k^T\boldsymbol{S}_{k+1}\right) = \{\widetilde{\boldsymbol{z}} \in \mathbb{R}^D : \boldsymbol{S}_{k+1}\widetilde{\boldsymbol{z}} = \boldsymbol{S}_k\widetilde{\boldsymbol{z}}\}. \tag{31}$$

d) If $\boldsymbol{S}_{k+1} = \boldsymbol{S}_k$, then $\boldsymbol{\Phi}_k = P_{\mathcal{R}(\boldsymbol{S}_k)}$ and $\mathrm{Fix}(\boldsymbol{\Phi}_k) = \mathcal{R}(\boldsymbol{S}_k)$.

Proof: See Appendix B. □

Define

$$\Theta_k^* := \inf_{\boldsymbol{x}\in\mathrm{Fix}(\boldsymbol{\Phi}_k)} \Theta_k(\boldsymbol{x}),\ k \in \mathbb{N}_0, \tag{32}$$

$$\Omega_k := \{\boldsymbol{h} \in \mathrm{Fix}(\boldsymbol{\Phi}_k) : \Theta_k(\boldsymbol{h}) = \Theta_k^*\},\ k \in \mathbb{N}_0. \tag{33}$$

(As mentioned before (21), the constraint set $\mathrm{Fix}(\boldsymbol{\Phi}_k)$ is associated with the requirements $\boldsymbol{h}_k \in \mathcal{R}(\boldsymbol{S}_k)$ for any $k \in \mathbb{N}_0$.) Then, the following theorem holds.

Theorem 1: The sequence $(\boldsymbol{h}_k)_{k\in\mathbb{N}_0}$ generated by R-APSM satisfies the following.

a) (Monotone Approximation)

I) Assume $\Omega_k \neq \emptyset$. Then, for any $\lambda_k \in [0, 2(1 - \Theta_k^*/\Theta_k(\boldsymbol{h}_k))]$, (21) holds.

II) Assume in addition $\Theta_k(\boldsymbol{h}_k) > \inf_{\boldsymbol{x}\in\mathbb{R}^N}\Theta_k(\boldsymbol{x}) \geq 0$. Then, for any $\lambda_k \in (0, 2(1 - \Theta_k^*/\Theta_k(\boldsymbol{h}_k)))$,

$$\left\|\boldsymbol{h}_{k+1} - \boldsymbol{h}_{(k)}^*\right\| < \left\|\boldsymbol{h}_k - \boldsymbol{h}_{(k)}^*\right\|, \quad \forall \boldsymbol{h}_{(k)}^* \in \Omega_k. \tag{34}$$

b) (Boundedness, Asymptotic Optimality) Assume

$$\exists K_0 \in \mathbb{N}_0 \text{ s.t. } \begin{cases} \text{(i) } \Theta_k^* = 0, \quad \forall k \geq K_0, \text{ and} \\ \text{(ii) } \Omega := \bigcap_{k\geq K_0}\Omega_k \neq \emptyset. \end{cases} \tag{35}$$

Then $(\boldsymbol{h}_k)_{k\in\mathbb{N}_0}$ is bounded. In particular, if there exist $\varepsilon_1, \varepsilon_2 > 0$ such that $\lambda_k \in [\varepsilon_1, 2 - \varepsilon_2] \subset (0, 2)$, then (22) holds, provided that $(\Theta_k'(\boldsymbol{h}_k))_{k\in\mathbb{N}_0}$ is bounded.

Proof: See Appendix C. □

Finally, for the Θ_k specified by (26), we discuss the assumptions made in Theorem 1. First, it is worth mentioning that $\boldsymbol{S}_k$ tends to stop moving when the estimates of $\boldsymbol{R}$ and $\boldsymbol{p}$ become reliable, and, in such a case, Proposition 1 implies $\mathrm{Fix}(\boldsymbol{\Phi}_k) = \mathcal{R}(\boldsymbol{S}_k)$. Hence, we assume $\mathrm{Fix}(\boldsymbol{\Phi}_k) = \mathcal{R}(\boldsymbol{S}_k)$ for simplicity here. Moreover, it mostly holds that $\bigcap_{\iota\in\mathcal{I}_k} H_\iota^-(\boldsymbol{h}_k) \cap \mathcal{R}(\boldsymbol{S}_k) \neq \emptyset$ at each $k \in \mathbb{N}_0$, unless the observed data are highly inconsistent. In this case, ($\Theta_k^* = 0$ and) $\Omega_k = \bigcap_{\iota\in\mathcal{I}_k} H_\iota^-(\boldsymbol{h}_k) \cap \mathcal{R}(\boldsymbol{S}_k)(\neq \emptyset)$, thus (21) holds. We remark that, under $\mathrm{Fix}(\boldsymbol{\Phi}_k) = \mathcal{R}(\boldsymbol{S}_k)$, the condition $\bigcap_{\iota\in\mathcal{I}_k} H_\iota^-(\boldsymbol{h}_k) \cap \mathcal{R}(\boldsymbol{S}_k) \neq \emptyset$ is sufficient but not necessary for (21) to hold. (In fact, Ω_k can be nonempty even if $\bigcap_{\iota\in\mathcal{I}_k} H_\iota^-(\boldsymbol{h}_k) = \emptyset$).

Under $\mathrm{Fix}(\boldsymbol{\Phi}_k) = \mathcal{R}(\boldsymbol{S}_k)$, the conditions in (35) are satisfied when $\bigcap_{k\geq K_0}[\bigcap_{\iota\in\mathcal{I}_k} H_\iota^-(\boldsymbol{h}_k) \cap \mathcal{R}(\boldsymbol{S}_k)] \neq \emptyset$, which mostly holds if the observed data are consistent for $k \geq K_0$. We mention that $(\Theta_k'(\boldsymbol{h}_k))_{k\in\mathbb{N}_0}$ for the Θ_k in (26) is automatically bounded [58].

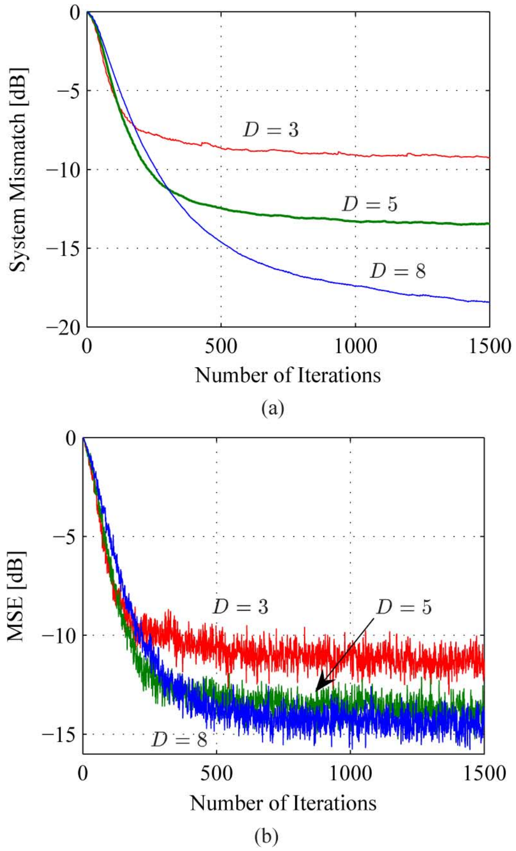
In dynamic environments, it is hardly possible to ensure $\mathrm{Fix}(\boldsymbol{\Phi}_k) = \mathcal{R}(\boldsymbol{S}_k)$ for all $k \geq K_0$, since $\boldsymbol{S}_k$ will move when the environment changes. In this case, the asymptotic optimality is difficult to be guaranteed. However, it is possible that the monotone approximation is guaranteed, because the environment would be nearly static in some (short) periods and, within such periods, $\boldsymbol{S}_k$ may stop moving.

V. Numerical Examples

This section provides numerical examples to verify the advantages of the proposed algorithm over the CGRRF method [20] for simple system identification problems. We omit a comparison with the RLS algorithm, because it is known that CGRRF provides convergence comparable to RLS with lower computational complexity and it does not suffer from any numerical instability problems [46], [47]. For the sake of conciseness, weakly correlated input signals are employed in order to avoid preconditioning. First, we examine the performance of the proposed algorithm for different values of D, q, and ρ, and also the performance of CGRRF for different values of the forgetting factor γ. Then, we compare the proposed and CGRRF methods in terms of i) robustness against impulsive noise and ii) tracking capability after a drastic change of $\boldsymbol{h}^*$. In all the simulations, we set $\widetilde{\boldsymbol{h}}_0 = \boldsymbol{0}$, $\mathcal{I}_k := \{k, k-1, \ldots, k-q+1\}$, and the matrix $\boldsymbol{S}_k$ is updated every m = ten iterations with $\widehat{\boldsymbol{R}}_0 := \boldsymbol{O}$ and $\widehat{\boldsymbol{p}}_0 := \boldsymbol{0}$.

A. Performance of the Proposed and CGRRF Algorithms

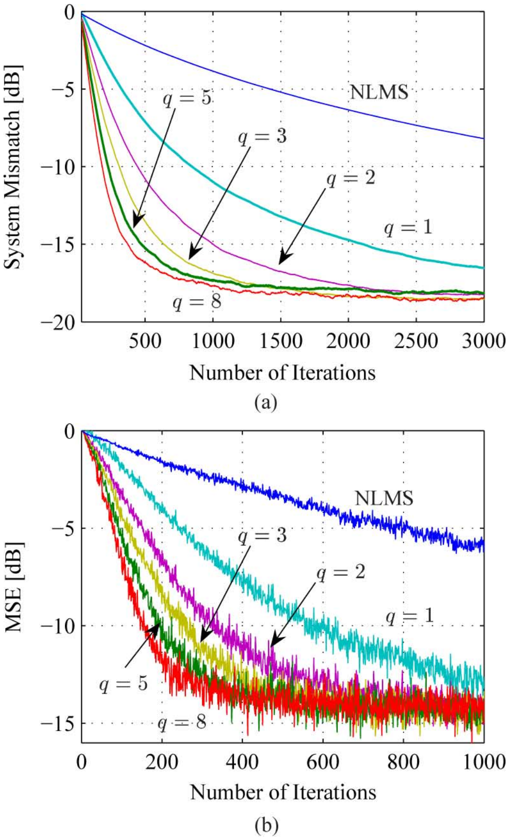
To compute arithmetic averages of MSE and system mismatch, i.e., $\|\boldsymbol{h}^* - \boldsymbol{h}_k\|^2/\|\boldsymbol{h}^*\|^2$, 300 independent experiments are performed. In each experiment, $\boldsymbol{h}^* \in \mathbb{R}^N$ is generated randomly for $N = 50$, and weakly correlated input signals $(u_k)_{k\in\mathbb{N}_0}$ are generated by passing white Gaussian signals $(s_k)_{k\in\mathbb{N}_0}$ through a length-30 finite-impulse-response (FIR) filter $\boldsymbol{f}_{\mathrm{FIR}} \in \mathbb{R}^{30}$ whose coefficients are chosen randomly; i.e., $u_k := \boldsymbol{s}_k^T\boldsymbol{f}_{\mathrm{FIR}}$, $k \in \mathbb{N}_0$, where



(a)

(b)

Fig. 6. Performance of the proposed algorithm for $D = 3, 5, 8$, $q = 4$, and $r = 1$ under SNR = 15 dB in (a) system mismatch and (b) MSE.



(a)

(b)

Fig. 7. Performance of the proposed algorithm for $D = 8$, $\rho = 0.15$, and $r = 1$ under $\mathrm{SNR} = 15$ dB in (a) system mismatch and (b) MSE.

$\boldsymbol{s}_k := [s_k, s_{k-1}, \ldots, s_{k-29}]^T$. The signal-to-noise ratio (SNR) is set to SNR $:= 10\log_{10}(\sigma_z^2/\sigma_n^2) = 15$ dB, where $\sigma_z^2 := E\{z_k^2\}$ with $z_k := \langle \boldsymbol{u}_k, \boldsymbol{h}^* \rangle$.

The parameters are set to[12] $\lambda_k = 0.03$, $\rho = 0.15$, $q = 4$, $r = 1$, $\gamma = 0.999$, and $D = 3, 5, 8$. The results are depicted in Fig. 6. It is seen that, from $D = 3$ to $D = 5$, an increase of D leads to better steady-state performance both in system mismatch and MSE. However, from $D = 5$ to $D = 8$, the gain in MSE is slight, although a significant gain is obtained in system mismatch. This is because the value of $\|\boldsymbol{h}_k - \boldsymbol{h}^*\|$ at the steady state is still not small enough in the case of $D = 5$, but the value of $\|\boldsymbol{h}_k - \boldsymbol{h}^*\|_{\boldsymbol{R}}$ is already small enough (see Section II).

Next we fix the value of $D = 8$, and change the value of q as $q = 1, 2, 3, 5, 8$. The rest of the parameters are the same as in Fig. 6. The results are depicted in Fig. 7. As a benchmark, the performance curves of NLMS for step size $\lambda_k = 0.03$ are also drawn. It is seen that an increase of q (the number of parallel projections computed at each iteration) raises the speed of convergence significantly.

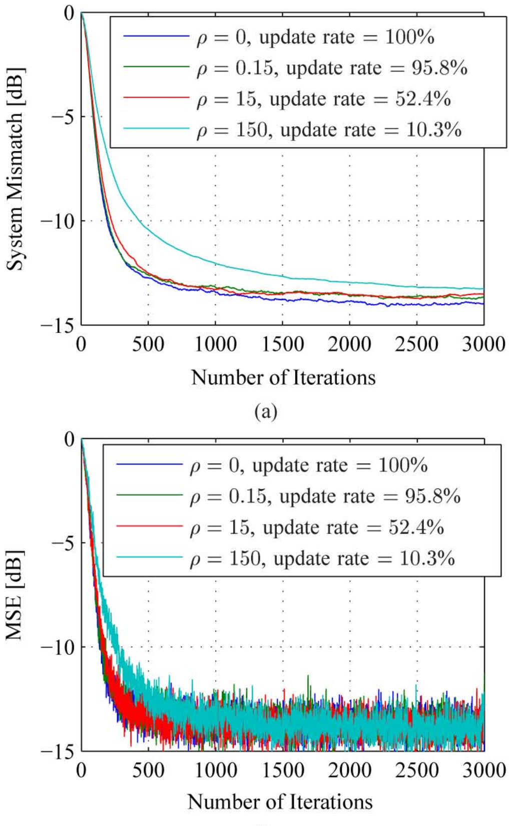
We now show the robustness of the proposed algorithm against the choice of ρ. We set $D = q = 5$, and change ρ as $\rho = 0, 0.15, 15, 150$. The rest of the parameters are the same as in Fig. 6. The results are depicted in Fig. 8, which shows that the performance is nearly constant for a wide range of ρ, while the update rate decreases as ρ increases. We stress that the purpose of this simulation is to show how robust the proposed algorithm is against the choice of ρ, but *not to show its performance for appropriately chosen parameters*. Indeed, although the use of $\rho = 150$ results in relatively slow convergence in the figure, a slight increase of the step size makes its performance comparable to the other choices of ρ with low update rate. Typically, the update rate for an appropriately chosen ρ is less than 10%, as mentioned in Section III-F.

Finally, we examine how the value of γ affects the performance of CGRRF with $D = 5$ fixed. Fig. 9 plots γ versus (a) the iteration number required to converge in MSE and (b) steady-state performance in system mismatch and MSE; as expected naturally, there is a tradeoff between (a) and (b). Nevertheless, it is seen that $\gamma = 0.999$ would be a good compromise (see also Section V-B).

B. Robustness Against Impulsive Noise—Proposed Versus CGRRF

We consider the situation where we have impulsive noise at the one thousandth iteration. Impulsive noise is assumed to decay exponentially and generated as follows: $v_k := \sigma_v(-1)^k e^{-(k-1000)}$, $k = 1000, 1001, \ldots, 1099$, where $\sigma_v^2 := 20\sigma_z^2$. The $\boldsymbol{h}^*$ and the input signals are generated in the same way as in Section V-A, and the SNR is set to $\mathrm{SNR} = 20$ dB. For the proposed and NLMS algorithms, the step size is set to $\lambda_k = 0.05$. For the proposed algorithm, moreover, we set

[12] In the current study, we only focus on the case of $r = 1$ to make the parameter settings simple. In fact, it has been reported in [31]–[33], [35], and [36] that fast convergence and good steady-state performance are attained when we use $r = 1$ and a large value of q (e.g., $q = 8, 16, 32$) for the N within the range of 64 to 2000 in the (full-rank) APSP algorithm [29].



(a)

(b)

Fig. 8. Robustness of the proposed algorithm against the choice of ρ for $D = q = 5$, and $r = 1$ under $\mathrm{SNR} = 15$ dB in (a) system mismatch and (b) MSE.

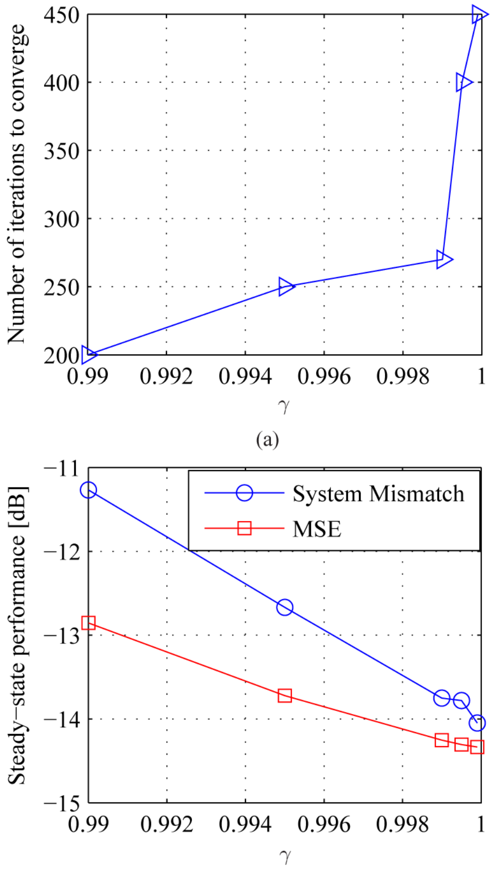
$\rho = 0.1$, $q = 1, 5$, $r = 1$, $\gamma = 0.999$, and $D = 5$. For robustness against the impulsive noise, we use the method described in Section III-G for $\delta_{\boldsymbol{h}} = 0.1$. For CGRRF, $\gamma = 0.99, 0.999$, $D = 5$, and the initial vector at each time instant is set to the zero vector.

The results are plotted in Fig. 10. It is seen that the methods for $\gamma = 0.999$ are robust against the impulsive noise, while CGRRF for $\gamma = 0.99$ exhibits instability (Although it might be possible to devise a more robust scheme against impulsive noise with respect to the choice of γ, it is beyond the scope of our current study.) This and Fig. 9 suggest that, for the sake of good performance and robustness, $\gamma = 0.999$ would be a reasonable choice.

C. Tracking Capability After Drastic Change of $\boldsymbol{h}^*$—Proposed Versus CGRRF

We consider the situation where $\boldsymbol{h}^*$ changes dynamically at one thousandth iteration; the input statistics are *unchanged*, which means that all that is changed is the cross-correlation vector $\boldsymbol{p}$. The other conditions are the same as in Section V-B.

Fig. 11 plots the results. As expected from the discussion in Section II, the tracking speed of CGRRF (for $\gamma = 0.999$) after the sudden change of $\boldsymbol{h}^*$ is slow, although its convergence speed at the initial phase is fast. On the other hand, the proposed algorithm for $q = 5$ achieves fast initial convergence and good tracking performance simultaneously. As expected from the results in Fig. 9, the use of $\gamma = 0.99$ in CGRRF causes significant degradation of the steady-state performance (as well as better tracking capability).



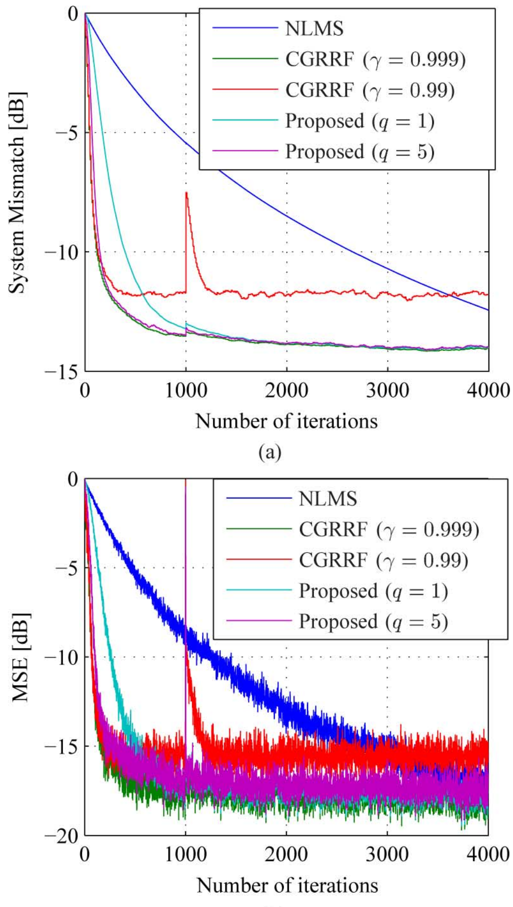
(a)

(b)

Fig. 9. Performance of CGRRF for different values of γ under $\mathrm{SNR} = 15$ dB and $D = 5$. Performance measures are (a) the number of iterations to converge in MSE and (b) steady-state performance in system mismatch and MSE, respectively.

VI. Conclusion

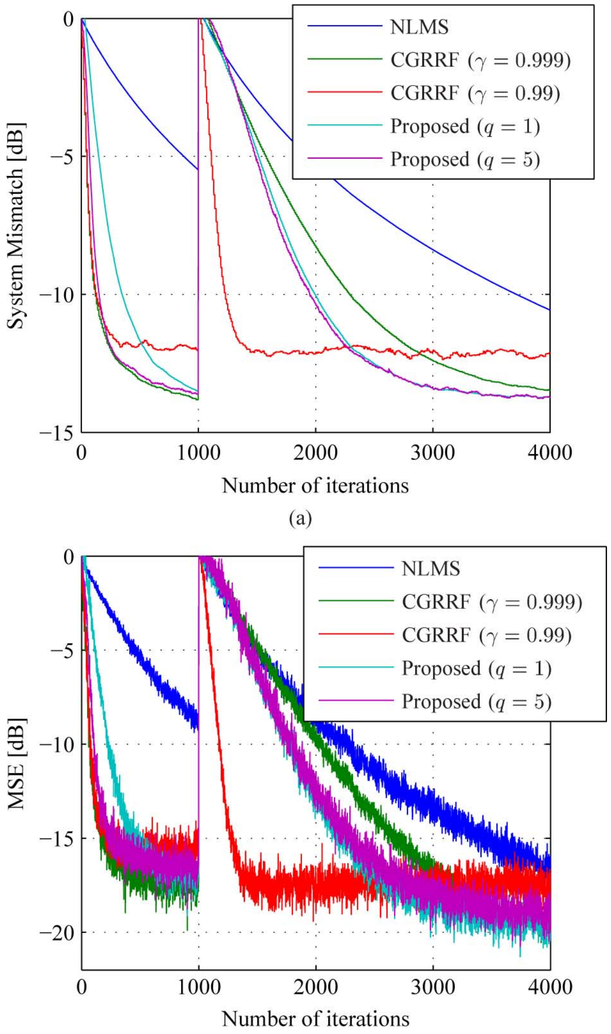
This paper has presented a robust reduced-rank adaptive filtering algorithm based on the Krylov subspace and the set-theoretic adaptive filtering method. The proposed algorithm provides an excellent tradeoff between performance (in particular, tracking capability) and computational complexity. The valuable properties (monotone approximation and asymptotic optimality) of the proposed algorithm have been proven within the framework of the R-APSM. The presented design and analysis of the proposed algorithm reveal better understanding of Krylov-subspace-based filtering methods. It would be worth repeating that the algorithm has a fault tolerance nature due to its inherently parallel structure. The numerical examples have demonstrated that the proposed algorithm exhibits much better tracking performance than CGRRF (with comparable computational complexity) as well as robustness against impulsive noise. This suggests that the proposed algorithm performs better than the existing Krylov-subspace-based reduced-rank methods in nonstationary environments. We finally mention that the proposed algorithm has no numerical problems, since



(a)

(b)

Fig. 10. Performance against impulsive noise under SNR = 20 dB in (a) system mismatch and (b) MSE. For the proposed algorithm, $\lambda_k = 0.05$, $k \in \mathbb{N}_0$, $D = 5$, $\rho = 0.1$, $r = 1$, and $\gamma = 0.999$. For CGRRF, $D = 5$. For NLMS, $\lambda_k = 0.05$, $k \in \mathbb{N}_0$.



(a)

(b)

Fig. 11. Performance against a drastic change of $\boldsymbol{h}^*$ under the same conditions as in Fig. 10.

it requires no matrix inversion, implying that the algorithm is easy to implement.

Appendix A
Mathematical Definitions

Let $\mathcal{H}$ denote a real Hilbert space equipped with an inner product $\langle \cdot, \cdot \rangle$ and its induced norm $\|\cdot\|$. We introduce some mathematical definitions used in this paper.

a) A set $C \subset \mathcal{H}$ is said to be *convex* if $\nu \boldsymbol{x} + (1-\nu)\boldsymbol{y} \in C$, $\forall \boldsymbol{x}, \boldsymbol{y} \in C, \forall \nu \in (0,1)$. A function $\Theta : \mathcal{H} \to \mathbb{R}$ is said to be *convex* if $\Theta(\nu \boldsymbol{x} + (1-\nu)\boldsymbol{y}) \leq \nu\Theta(\boldsymbol{x}) + (1-\nu)\Theta(\boldsymbol{y})$, $\forall \boldsymbol{x}, \boldsymbol{y} \in \mathcal{H}$, $\forall \nu \in (0,1)$; the inequality is sometimes called *Jensen's inequality* [59].

b) A mapping T is said to be i) *nonexpansive* if $\|T(\boldsymbol{x}) - T(\boldsymbol{y})\| \leq \|\boldsymbol{x} - \boldsymbol{y}\|$, $\forall \boldsymbol{x}, \boldsymbol{y} \in \mathcal{H}$; ii) *attracting nonexpansive* if T is nonexpansive with $\mathrm{Fix}(T) \neq \emptyset$ and $\|T(\boldsymbol{x}) - \boldsymbol{f}\|^2 < \|\boldsymbol{x} - \boldsymbol{f}\|^2$, $\forall (\boldsymbol{x}, \boldsymbol{f}) \in \mathcal{H} \setminus \mathrm{Fix}(T) \times \mathrm{Fix}(T)$; and iii) *strongly* or *η-attracting nonexpansive* if T is nonexpansive with $\mathrm{Fix}(T) \neq \emptyset$ and there exists $\eta > 0$ s.t. $\eta\|\boldsymbol{x} - T(\boldsymbol{x})\|^2 \leq \|\boldsymbol{x} - \boldsymbol{f}\|^2 - \|T(\boldsymbol{x}) - \boldsymbol{f}\|^2$, $\forall \boldsymbol{x} \in \mathcal{H}$, $\forall \boldsymbol{f} \in \mathrm{Fix}(T)$.

c) Given a continuous convex function $\Theta : \mathcal{H} \to \mathbb{R}$, the *subdifferential* of Θ at any $\boldsymbol{y} \in \mathcal{H}$, defined as $\partial\Theta(\boldsymbol{y}) := \{\boldsymbol{a} \in \mathcal{H} : \langle \boldsymbol{x} - \boldsymbol{y}, \boldsymbol{a} \rangle + \Theta(\boldsymbol{y}) \leq \Theta(\boldsymbol{x}), \forall \boldsymbol{x} \in \mathcal{H}\}$, is nonempty. An element of the subdifferential $\partial\Theta(\boldsymbol{y})$ is called a subgradient of Θ at $\boldsymbol{y}$.

d) Suppose that a continuous convex function $\Theta : \mathcal{H} \to \mathbb{R}$ satisfies $\mathrm{lev}_{\leq 0}\Theta := \{\boldsymbol{x} \in \mathcal{H} : \Theta(\boldsymbol{x}) \leq 0\} \neq \emptyset$. Then, for a subgradient $\Theta'(\boldsymbol{x}) \in \partial\Theta(\boldsymbol{x})$, a mapping $T_{\mathrm{sp}(\Theta)} : \mathcal{H} \to \mathcal{H}$ defined by

$$T_{\mathrm{sp}(\Theta)}(\boldsymbol{x}) := \begin{cases} \boldsymbol{x} - \frac{\Theta(\boldsymbol{x})}{\|\Theta'(\boldsymbol{x})\|^2}\Theta'(\boldsymbol{x}) & \text{if } \Theta(\boldsymbol{x}) > 0 \\ \boldsymbol{x} & \text{if } \Theta(\boldsymbol{x}) \leq 0 \end{cases}$$

is called *a subgradient projection relative to* Θ (see, e.g., [39]).

Appendix B
Properties of $\boldsymbol{\Phi}_k$ and Proof of Proposition 1

This Appendix presents basic properties of $\boldsymbol{\Phi}_k$, the Proof of Proposition 1, and some results regarding the attracting nonexpansivity of $\boldsymbol{\Phi}_k$ (see Appendix A).

Lemma B.1: (*Basic properties of* $\boldsymbol{\Phi}_k$)

a) $\boldsymbol{\Phi}_k \boldsymbol{x} = \boldsymbol{S}_{k+1}\widetilde{\boldsymbol{x}}$ for all $\widetilde{\boldsymbol{x}} \in \mathbb{R}^D$ and $\boldsymbol{x} = \boldsymbol{S}_k\widetilde{\boldsymbol{x}}$.

b) For any $\boldsymbol{x} \in \mathbb{R}^N$, $\|\boldsymbol{\Phi}_k \boldsymbol{x}\| \leq \|\boldsymbol{x}\|$; the equality holds if and only if $\boldsymbol{x} \in \mathcal{R}(\boldsymbol{S}_k)$. Moreover, the mapping $\boldsymbol{\Phi}_k$ is *nonexpansive* (cf. Appendix A). □

Proof of Lemma B.1.a: For all $\widetilde{\boldsymbol{x}} \in \mathbb{R}^D$, we have $\boldsymbol{\Phi}_k \boldsymbol{x} = \boldsymbol{S}_{k+1}\boldsymbol{S}_k^T \boldsymbol{S}_k \widetilde{\boldsymbol{x}} = \boldsymbol{S}_{k+1}\widetilde{\boldsymbol{x}}$.

Proof of Lemma B.1.b: $\boldsymbol{S}_{k+1}^T \boldsymbol{S}_{k+1} = \boldsymbol{S}_k^T \boldsymbol{S}_k = \boldsymbol{I}$, we have, for any $\boldsymbol{x} \in \mathbb{R}^N$,

$$\begin{aligned}\|\boldsymbol{\Phi}_k \boldsymbol{x}\| &= \left\|\boldsymbol{S}_{k+1}\boldsymbol{S}_k^T \boldsymbol{x}\right\| \\ &= \left\|\boldsymbol{S}_k \boldsymbol{S}_k^T \boldsymbol{x}\right\| = \left\|P_{\mathcal{R}(\boldsymbol{S}_k)}(\boldsymbol{x})\right\| \leq \|\boldsymbol{x}\|. \qquad \text{(B.1)}\end{aligned}$$

The inequality is verified by the nonexpansivity of the projection operator; the equality holds if and only if $\boldsymbol{x} \in \mathcal{R}(\boldsymbol{S}_k)$. (B.1) and the linearity of $\boldsymbol{\Phi}_k$ suggest the nonexpansivity of $\boldsymbol{\Phi}_k$. □

Proof of Proposition 1:

Proof of Proposition 1.a: $\boldsymbol{\Phi}_k \boldsymbol{0} = \boldsymbol{0}$ implies $\boldsymbol{0} \in \mathrm{Fix}(\boldsymbol{\Phi}_k)$.

Proof of Proposition 1.b: Suppose $\boldsymbol{h} \in \mathrm{Fix}(\boldsymbol{\Phi}_k)$. Then, $\boldsymbol{h} = \boldsymbol{\Phi}_k \boldsymbol{h} \in \mathcal{R}(\boldsymbol{S}_{k+1})$. Moreover, by Lemma B.1.b, $\boldsymbol{\Phi}_k \boldsymbol{h} = \boldsymbol{h} \Rightarrow \boldsymbol{h} \in \mathcal{R}(\boldsymbol{S}_k)$. Hence, $\boldsymbol{h} \in \mathcal{R}(\boldsymbol{S}_k) \cap \mathcal{R}(\boldsymbol{S}_{k+1})$, implying that $\mathrm{Fix}(\boldsymbol{\Phi}_k) \subset \mathcal{R}(\boldsymbol{S}_k) \cap \mathcal{R}(\boldsymbol{S}_{k+1})$.

Proof of Proposition 1.c: To prove (31), it is sufficient to show

$$\boldsymbol{S}_k^T \boldsymbol{S}_{k+1}\widetilde{\boldsymbol{z}} = \widetilde{\boldsymbol{z}} \Leftrightarrow \boldsymbol{S}_{k+1}\widetilde{\boldsymbol{z}} = \boldsymbol{S}_k \widetilde{\boldsymbol{z}}. \qquad \text{(B.2)}$$

Assume $\boldsymbol{S}_k^T \boldsymbol{S}_{k+1}\widetilde{\boldsymbol{z}} = \widetilde{\boldsymbol{z}}$. Then, we have

$$\begin{aligned}&\boldsymbol{S}_k \boldsymbol{S}_k^T \boldsymbol{S}_{k+1}\widetilde{\boldsymbol{z}} = \boldsymbol{S}_k \widetilde{\boldsymbol{z}} \\ &\Leftrightarrow P_{\mathcal{R}(\boldsymbol{S}_k)}(\boldsymbol{S}_{k+1}\widetilde{\boldsymbol{z}}) = \boldsymbol{S}_k \widetilde{\boldsymbol{z}} && \text{(B.3)} \\ &\Rightarrow \left\|P_{\mathcal{R}(\boldsymbol{S}_k)}(\boldsymbol{S}_{k+1}\widetilde{\boldsymbol{z}})\right\| = \|\boldsymbol{S}_k \widetilde{\boldsymbol{z}}\| \\ &= \|\widetilde{\boldsymbol{z}}\| = \|\boldsymbol{S}_{k+1}\widetilde{\boldsymbol{z}}\| && \text{(B.4)} \\ &\Leftrightarrow \left\|P_{\mathcal{R}(\boldsymbol{S}_k)}(\boldsymbol{S}_{k+1}\widetilde{\boldsymbol{z}}) - \boldsymbol{S}_{k+1}\widetilde{\boldsymbol{z}}\right\| = 0 && \text{(B.5)} \\ &\Leftrightarrow P_{\mathcal{R}(\boldsymbol{S}_k)}(\boldsymbol{S}_{k+1}\widetilde{\boldsymbol{z}}) = \boldsymbol{S}_{k+1}\widetilde{\boldsymbol{z}}. && \text{(B.6)}\end{aligned}$$

Here, the equivalence between (B.4) and (B.5) is verified by the well-known Pythagorean theorem. From (B.3) and (B.6), we obtain $\boldsymbol{S}_{k+1}\widetilde{\boldsymbol{z}} = \boldsymbol{S}_k \widetilde{\boldsymbol{z}}$. The converse is obvious, which verifies (B.2).

By Proposition 1.b, any element $\boldsymbol{z} \in \mathrm{Fix}(\boldsymbol{\Phi}_k)$ can be expressed as $\boldsymbol{z} = \boldsymbol{S}_{k+1}\widetilde{\boldsymbol{z}}$, $\exists \widetilde{\boldsymbol{z}} \in \mathbb{R}^D$. Then, we have

$$\begin{aligned}\boldsymbol{S}_{k+1}\widetilde{\boldsymbol{z}} \in \mathrm{Fix}(\boldsymbol{\Phi}_k) &\Leftrightarrow \boldsymbol{S}_{k+1}\boldsymbol{S}_k^T \boldsymbol{S}_{k+1}\widetilde{\boldsymbol{z}} = \boldsymbol{S}_{k+1}\widetilde{\boldsymbol{z}} \\ &\Leftrightarrow \boldsymbol{S}_k^T \boldsymbol{S}_{k+1}\widetilde{\boldsymbol{z}} = \widetilde{\boldsymbol{z}} \\ &\Leftrightarrow \widetilde{\boldsymbol{z}} \in \mathrm{Fix}\left(\boldsymbol{S}_k^T \boldsymbol{S}_{k+1}\right) \qquad \text{(B.7)}\end{aligned}$$

which with (31) verifies (30).

Proof of Proposition 1.d: The orthonormality of $\boldsymbol{S}_k$ and $\boldsymbol{S}_k = \boldsymbol{S}_{k+1}$ imply that $\boldsymbol{\Phi}_k = P_{\mathcal{R}(\boldsymbol{S}_k)}$ [60]. Moreover, due to the basic property of projection, we obtain $\mathrm{Fix}(\boldsymbol{\Phi}_k) = \mathrm{Fix}(P_{\mathcal{R}(\boldsymbol{S}_k)}) = \mathcal{R}(\boldsymbol{S}_k)$. □

Finally, thanks to Proposition 1, we can show that $\boldsymbol{\Phi}_k$ is attracting nonexpansive if and only if $\boldsymbol{S}_k = \boldsymbol{S}_{k+1}$, as described below.

Lemma B.2 (On Attracting Nonexpansivity of $\boldsymbol{\Phi}_k$):

a) If $\boldsymbol{S}_k = \boldsymbol{S}_{k+1}$, then $\boldsymbol{\Phi}_k$ is the projection matrix thus *1-attracting nonexpansive*.

b) If $\boldsymbol{S}_k \neq \boldsymbol{S}_{k+1}$, then $\boldsymbol{\Phi}_k$ is nonexpansive but *not* attracting nonexpansive.

Proof of Lemma B.2.a: By Proposition 1.d, $\boldsymbol{S}_k = \boldsymbol{S}_{k+1} \Rightarrow \boldsymbol{\Phi}_k = P_{\mathcal{R}(\boldsymbol{S}_k)}$, $\mathcal{R}(\boldsymbol{S}_k) = \mathrm{Fix}(\boldsymbol{\Phi}_k)$. Hence, by the Pythagorean theorem, we have

$$\begin{aligned}&\|\boldsymbol{x} - \boldsymbol{\Phi}_k \boldsymbol{x}\|^2 = \|\boldsymbol{x} - \boldsymbol{f}\|^2 - \|\boldsymbol{\Phi}_k \boldsymbol{x} - \boldsymbol{f}\|^2, \\ &\qquad\qquad \forall \boldsymbol{x} \in \mathbb{R}^N,\ \forall \boldsymbol{f} \in \mathrm{Fix}(\boldsymbol{\Phi}_k). \qquad \text{(B.8)}\end{aligned}$$

This means that the mapping $\boldsymbol{\Phi}_k$ is 1-attracting nonexpansive.

Proof of Lemma B.2.b: By $\boldsymbol{S}_k \neq \boldsymbol{S}_{k+1}$, there exists $\widetilde{\boldsymbol{z}}^* \in \mathbb{R}^D$ s.t. $\boldsymbol{S}_{k+1}\widetilde{\boldsymbol{z}}^* \neq \boldsymbol{S}_k \widetilde{\boldsymbol{z}}^*$. For such a $\widetilde{\boldsymbol{z}}^*$, it holds that $\boldsymbol{\Phi}_k \boldsymbol{S}_k \widetilde{\boldsymbol{z}}^* = \boldsymbol{S}_{k+1}\boldsymbol{S}_k^T \boldsymbol{S}_k \widetilde{\boldsymbol{z}}^* = \boldsymbol{S}_{k+1}\widetilde{\boldsymbol{z}}^* \neq \boldsymbol{S}_k \widetilde{\boldsymbol{z}}^*$, implying $\boldsymbol{S}_k \widetilde{\boldsymbol{z}}^* \notin \mathrm{Fix}(\boldsymbol{\Phi}_k)$. Hence, we obtain

$$\|\boldsymbol{\Phi}_k \boldsymbol{z}^* - \boldsymbol{0}\| = \|\boldsymbol{S}_{k+1}\widetilde{\boldsymbol{z}}^*\| = \|\boldsymbol{S}_k \widetilde{\boldsymbol{z}}^*\| = \|\boldsymbol{z}^* - \boldsymbol{0}\|, \qquad \text{(B.9)}$$

where $\boldsymbol{z}^* := \boldsymbol{S}_k \widetilde{\boldsymbol{z}}^* \in \mathbb{R}^N \setminus \mathrm{Fix}(\boldsymbol{\Phi}_k)$ and $\boldsymbol{0} \in \mathrm{Fix}(\boldsymbol{\Phi}_k)$. This verifies that $\boldsymbol{\Phi}_k$ is not attracting nonexpansive. □

Appendix C
Proof of Theorem 1

Proof of (a)-(I): If $\Theta_k'(\boldsymbol{h}_k) = \boldsymbol{0}$, then, $\forall \boldsymbol{h}_{(k)}^* \in \Omega_k$,

$$\begin{aligned}\left\|\boldsymbol{h}_{k+1} - \boldsymbol{h}_{(k)}^*\right\|^2 &= \left\|\boldsymbol{\Phi}_k \boldsymbol{h}_k - \boldsymbol{\Phi}_k \boldsymbol{h}_{(k)}^*\right\|^2 \\ &\leq \left\|\boldsymbol{h}_k - \boldsymbol{h}_{(k)}^*\right\|^2. \qquad \text{(C.1)}\end{aligned}$$

Assume now $\Theta_k'(\boldsymbol{h}_k) \neq \boldsymbol{0}$. In this case, we have

$$\begin{aligned}&\left\|\boldsymbol{h}_{k+1} - \boldsymbol{h}_{(k)}^*\right\|^2 \\ &= \left\|\boldsymbol{\Phi}_k \left[\boldsymbol{h}_k - \lambda_k \frac{\Theta_k(\boldsymbol{h}_k)}{\|\Theta_k'(\boldsymbol{h}_k)\|^2}\Theta_k'(\boldsymbol{h}_k)\right] - \boldsymbol{\Phi}_k \boldsymbol{h}_{(k)}^*\right\|^2 \\ &\leq \left\|\boldsymbol{h}_k - \boldsymbol{h}_{(k)}^* - \lambda_k \frac{\Theta_k(\boldsymbol{h}_k)}{\|\Theta_k'(\boldsymbol{h}_k)\|^2}\Theta_k'(\boldsymbol{h}_k)\right\|^2 \\ &= \left\|\boldsymbol{h}_k - \boldsymbol{h}_{(k)}^*\right\|^2 + \lambda_k^2 \frac{\Theta_k^2(\boldsymbol{h}_k)}{\|\Theta_k'(\boldsymbol{h}_k)\|^2} \\ &\qquad - 2\lambda_k \frac{\Theta_k(\boldsymbol{h}_k)}{\|\Theta_k'(\boldsymbol{h}_k)\|^2}\left\langle \Theta_k'(\boldsymbol{h}_k), \boldsymbol{h}_k - \boldsymbol{h}_{(k)}^*\right\rangle \\ &\leq \left\|\boldsymbol{h}_k - \boldsymbol{h}_{(k)}^*\right\|^2 - \lambda_k \\ &\qquad \times \left[2\left(1 - \frac{\Theta_k^*}{\Theta_k(\boldsymbol{h}_k)}\right) - \lambda_k\right] \frac{\Theta_k^2(\boldsymbol{h}_k)}{\|\Theta_k'(\boldsymbol{h}_k)\|^2} \qquad \text{(C.2)}\end{aligned}$$

which verifies (21). Here, the first and second inequalities are verified by the nonexpansivity of $\boldsymbol{\Phi}_k$ and the definition of subgradient (see Lemma B.1 and Appendix A), respectively.

Proof of (a)-(II): Noting that $\Theta_k(\boldsymbol{h}_k) > \inf_{\boldsymbol{x} \in \mathbb{R}^N} \Theta_k(\boldsymbol{x})$ implies $\Theta_k'(\boldsymbol{h}_k) \neq \boldsymbol{0}$, we can readily verify (34) by (C.2).

Proof of (b): From Theorem 1.a.I, we see that the nonnegative sequence $(\|\boldsymbol{h}_k - \boldsymbol{\omega}\|)_{k \geq K_0}$ for any $\boldsymbol{\omega} \in \Omega$ is convergent, hence $(\boldsymbol{h}_k)_{k \in \mathbb{N}_0}$ is bounded. Moreover, since $\boldsymbol{0} \in \partial\Theta_k(\boldsymbol{h}_k)$ implies $\Theta_k(\boldsymbol{h}_k) = 0$, it is sufficient to check the case $\Theta_k'(\boldsymbol{h}_k) \neq \boldsymbol{0}$. In this case, by (C.2), we have

$$\|\boldsymbol{h}_k - \boldsymbol{\omega}\|^2 - \|\boldsymbol{h}_{k+1} - \boldsymbol{\omega}\|^2 \geq \varepsilon_1 \varepsilon_2 \frac{\Theta_k^2(\boldsymbol{h}_k)}{\|\Theta_k'(\boldsymbol{h}_k)\|^2} \geq 0. \qquad \text{(C.3)}$$

Therefore, the convergence of $(\|\boldsymbol{h}_k - \boldsymbol{\omega}\|)_{k \geq K_0}$ implies

$$\lim_{k \to \infty} \frac{\Theta_k^2(\boldsymbol{h}_k)}{\|\Theta_k'(\boldsymbol{h}_k)\|^2} = 0, \qquad \text{(C.4)}$$

hence the boundedness of $(\Theta'_k(\boldsymbol{h}_k))_{k\geq\mathbb{N}_0}$ ensures $\lim_{k\to\infty,\Theta'_k(\boldsymbol{h}_k)\neq\mathbf{0}}\Theta_k(\boldsymbol{h}_k)=0$. □

References

[1] D. W. Tufts, R. Kumaresan, and I. Kirsteins, "Data adaptive signal estimation by singular value decomposition of a data matrix," *Proc. IEEE*, vol. 70, pp. 684–685, Jun. 1982.

[2] W. F. Gabriel, "Using spectral estimation techniques in adaptive processing antenna systems," *IEEE Trans. Antennas Propag.*, vol. AP-34, pp. 291–300, Mar. 1986.

[3] L. L. Scharf and D. W. Tufts, "Rank reduction for modeling stationary signals," *IEEE Trans. Acoust., Speech,Signal Process.*, vol. ASSP-35, no. 3, pp. 350–355, Mar. 1987.

[4] B. D. Van Veen and R. A. Roberts, "Partially adaptive beamformer design via output power minimization," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-35, pp. 1524–1532, Nov. 1987.

[5] L. L. Scharf, "The SVD and reduced rank signal processing," *Signal Process.*, vol. 25, no. 2, pp. 113–133, 1991.

[6] A. M. Haimovich and Y. Bar-Ness, "An eigenanalysis interference canceler," *IEEE Trans. Signal Process.*, vol. 39, no. 1, pp. 76–84, Jan. 1991.

[7] J. S. Goldstein and I. S. Reed, "Reduced-rank adaptive filtering," *IEEE Trans. Signal Process.*, vol. 45, no. 2, pp. 492–496, Feb. 1997.

[8] X. Wang and H. V. Poor, "Blind multiuser detection: A subspace approach," *IEEE Trans. Inf. Theory*, vol. 44, no. 2, pp. 677–690, Mar. 1998.

[9] E. G. Ström and S. L. Miller, "Properties of the single-bit single-user MMSE receiver for DS-CDMA system," *IEEE Trans. Commun.*, vol. 47, pp. 416–425, Mar. 1999.

[10] Y. Song and S. Roy, "Blind adaptive reduced-rank detection for DSCDMA signals in multipath channels," *IEEE J. Sel. Areas Commun.*, vol. 17, no. 11, pp. 1960–1970, Nov. 1999.

[11] R. C. de Lamare and R. Sampaio-Neto, "Adaptive reduced-rank MMSE filtering with interpolated FIR filters and adaptive interpolators," *IEEE Signal Process. Lett.*, vol. 12, no. 3, pp. 177–180, Mar. 2005.

[12] M. Yukawa, R. C. de Lamare, and R. Sampaio-Neto, "Efficient acoustic echo cancellation with reduced-rank adaptive filtering based on selective decimation and adaptive interpolation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 56, no. 4, pp. 696–710, May 2008.

[13] S. Moshavi, E. G. Kanterakis, and D. L. Schilling, "Multistage linear receivers for DS-CDMA systems," *Int. J. Wireless Inf. Netw.*, vol. 3, no. 1, pp. 1–17, 1996.

[14] J. S. Goldstein, I. S. Reed, and L. L. Scharf, "A multistage representation of the Wiener filter based on orthogonal projections," *IEEE Trans. Inf. Theory*, vol. 44, no. 7, pp. 2943–2959, Nov. 1998.

[15] M. L. Honig and W. Xiao, "Performance of reduced-rank linear interference suppression," *IEEE Trans. Inf. Theory*, vol. 47, no. 5, pp. 1928–1946, Jul. 2001.

[16] M. L. Honig and J. S. Goldstein, "Adaptive reduced-rank interference suppression based on multistage Wiener filter," *IEEE Trans. Commun.*, vol. 50, no. 6, pp. 986–994, Jun. 2002.

[17] A. Kansal, S. N. Batalama, and D. A. Pados, "Adaptive maximum SINR RAKE filtering for DS-CDMA multipath fading channels," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 9, pp. 1765–1773, Dec. 1998.

[18] D. A. Pados and S. N. Batalama, "Joint space-time auxiliary-vector filtering for DS/CDMA systems with antenna arrays," *IEEE Trans. Commun.*, vol. 47, no. 9, pp. 1406–1415, Sep. 1999.

[19] M. L. Honig and W. Xiao, "Adaptive reduced-rank interference suppression with adaptive rank selection," in *Proc. MILCOM*, 2000, vol. 2, pp. 747–751.

[20] S. Chowdhury and M. D. Zoltowski, "Application of conjugate gradient methods in MMSE equalization for the forward link of DS-CDMA," in *Proc. IEEE Vehicular Technology Conf. 2001—Fall*, Oct. 2001, pp. 2434–2438.

[21] G. Dietl, M. D. Zoltowski, and M. Joham, "Reduced-rank equalization for EDGE via conjugate gradient implementation of multi-stage nested Wiener filter," in *Proc. IEEE Vehicular Technology Conf. 2001—Fall*, 2001, vol. 3, pp. 1912–1916.

[22] S. Burykh and K. Abed-Meraim, "Multi-stage reduced-rank adaptive filter with flexible structure," in *Proc. Eur. Signal Processing Conf.*, Sep. 2002.

[23] G. Dietl, M. D. Zoltowski, and M. Joham, "Recursive reduced-rank adaptive equalization for wireless communications," in *Proc. SPIE*, Toulouse, France, Apr. 2001, vol. 4395.

[24] W. Chen, U. Mitra, and P. Schniter, "On the equivalence of three reduced rank linear estimators with applications to DS-CDMA," *IEEE Trans. Inf. Theory*, vol. 48, no. 9, pp. 2609–2614, Sep. 2002.

[25] S. Burykh and K. Abed-Meraim, "Reduced-rank adaptive filtering using Krylov subspace," *EURASIP J. Appl. Signal Process.*, no. 12, pp. 1387–1400, Dec. 2002.

[26] G. K. E. Dietl, *Linear Estimation and Detection in Krylov Subspaces—Foundations in Signal Processing, Communications and Networking*. New York: Springer, 2007.

[27] S. Gollamudi, S. Nagaraj, S. Kapoor, and Y. H. Huang, "Set-membership filtering and a set-membership normalized LMS algorithm with an adaptive step size," *IEEE Signal Process. Lett.*, vol. 5, no. 5, pp. 111–114, May 1998.

[28] L. Guo, A. Ekpenyong, and Y. H. Huang, "Frequency-domain adaptive filtering—A set-membership approach," in *Proc. Asilomar Conf. Signals, Syst., Comput.*, 2003, pp. 2073–2077.

[29] I. Yamada, K. Slavakis, and K. Yamada, "An efficient robust adaptive filtering algorithm based on parallel subgradient projection techniques," *IEEE Trans. Signal Process.*, vol. 50, no. 5, pp. 1091–1101, May 2002.

[30] R. L. G. Cavalcante, I. Yamada, and K. Sakaniwa, "A fast blind MAI reduction based on adaptive projected subgradient method," *IEICE Trans. Fundam.*, vol. E87-A, no. 8, pp. 1973–1980, Aug. 2004.

[31] M. Yukawa, R. L. G. Cavalcante, and I. Yamada, "Efficient blind MAI suppression in DS/CDMA systems by embedded constraint parallel projection techniques," *IEICE Trans. Fundam.*, vol. E88-A, no. 8, pp. 2062–2071, Aug. 2005.

[32] M. Yukawa and I. Yamada, "Pairwise optimal weight realization—Acceleration technique for set-theoretic adaptive parallel subgradient projection algorithm," *IEEE Trans. Signal Process.*, vol. 54, no. 12, pp. 4557–4571, Dec. 2006.

[33] M. Yukawa, N. Murakoshi, and I. Yamada, "Efficient fast stereo acoustic echo cancellation based on pairwise optimal weight realization technique," *EURASIP J. Appl. Signal Process.*, vol. 2006, pp. 1–15, 2006, Article ID 84797.

[34] K. Slavakis, M. Yukawa, and I. Yamada, "Robust Capon beamforming by the adaptive projected subgradient method," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, May 2006, pp. 1005–1008.

[35] M. Yukawa, K. Slavakis, and I. Yamada, "Adaptive parallel quadraticmetric projection algorithms," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 5, pp. 1665–1680, Jul. 2007.

[36] R. G. Cavalcante and I. Yamada, "Multiaccess interference suppression in OSTBC-MIMO systems by adaptive projected subgradient method," *IEEE Trans. Signal Process.*, vol. 56, no. 3, pp. 1028–1042, Mar. 2008.

[37] K. Slavakis, S. Theodoridis, and I. Yamada, "Online kernel-based classification using adaptive projection algorithms," *IEEE Trans. Signal Process.*, vol. 56, no. 7, pt. 1, pp. 2781–2796, Jul. 2008.

[38] I. Yamada, "Adaptive projected subgradient method: A unified view for projection based adaptive algorithms," *J. IEICE*, vol. 86, no. 8, pp. 654–658, Aug. 2003, in Japanese.

[39] I. Yamada and N. Ogura, "Adaptive projected subgradient method for asymptotic minimization of sequence of nonnegative convex functions," *Numer. Funct. Anal. Optim.*, vol. 25, no. 7&8, pp. 593–617, 2004.

[40] S. Haykin, *Adaptive Filter Theory*, 4th ed. Englewood Cliffs, NJ: Prentice-Hall, 2002.

[41] G. H. Golub and C. F. V. Loan, *Matrix Computations*, 3rd ed. Baltimore, MD: The Johns Hopkins Univ. Press, 1996.

[42] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge: Cambridge University Press, 1985.

[43] O. Axelsson, "A survey of preconditioned iterative methods for linear systems of algebraic equations," *BIT*, vol. 25, pp. 166–187, 1985.

[44] Y. Saad, *Iterative Methods for Sparse Linear Systems*, 2nd ed. Philadelphia, PA: SIAM, 2003.

[45] A. W. Hull and W. K. Jenkins, "Preconditioned conjugate gradient methods for adaptive filtering," in *Proc. IEEE Int. Symp. Circuits Syst.*, Jun. 1991, pp. 540–543.

[46] G. K. Boray and M. D. Srinath, "Conjugate gradient techniques for adaptive filtering," *IEEE Trans. Circuits Syst. I*, vol. 39, no. 1, pp. 1–10, Jan. 1992.

[47] P. S. Chang and A. N. Willson, Jr., "Analysis of conjugate gradient algorithms for adaptive filtering," *IEEE Trans. Signal Process.*, vol. 48, no. 2, pp. 409–418, Feb. 2000.

[48] S. Werner and P. S. R. Diniz, "Set-membership affine projection algorithm," *IEEE Signal Process. Lett.*, vol. 8, no. 8, pp. 231–235, Aug. 2001.

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

[50] P. L. Combettes, "The foundations of set theoretic estimation," *Proc. IEEE*, vol. 81, no. 2, pp. 182–208, Feb. 1993.

[51] H. H. Bauschke and J. M. Borwein, "On projection algorithms for solving convex feasibility problems," *SIAM Rev.*, vol. 38, no. 3, pp. 367–426, 1996.

[52] Y. Censor and S. A. Zenios, *Parallel Optimization: Theory, Algorithm, and Optimization*. London, U.K.: Oxford Univ. Press, 1997.

[53] , D. Butnariu, Y. Censor, and S. Reich, Eds., *Inherently Parallel Algorithms in Feasibility and Optimization and Their Applications*. New York: Elsevier, 2001.

[54] M. Yukawa, "Krylov-proportionate adaptive filtering techniques not limited to sparse systems," *IEEE Trans. Signal Process.*, vol. 57, no. 3, pp. 927–943, Mar. 2009.

[55] T. Gänsler, S. L. Gay, M. M. Sondhi, and J. Benesty, "Double-talk robust fast converging algorithms for network echo cancellation," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 6, pp. 656–663, Nov. 2000.

[56] P. J. Huber, *Robust Statistics*. New York: Wiley, 1981.

[57] L. R. Vega, H. Rey, J. Benesty, and S. Tressens, "A robust adaptive filtering algorithm against impulsive noise," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, 2007, pp. 1437–1440.

[58] I. Yamada and N. Ogura, "Adaptive projected subgradient method and its applications to set theoretic adaptive filtering," in *Proc. 37th Asilomar Conf. Signals, Syst., Comput.*, Nov. 2003, pp. 600–606.

[59] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[60] G. Strang, *Linear Algebra and Its Applications*, 3rd ed. Philadelphia, PA: Saunders College Publishing, 1988.

Masahiro Yukawa (S'05–M'07) received the B.E., M.E., and Ph.D. degrees from the Tokyo Institute of Technology, Japan, in 2002, 2004, and 2006, respectively.

He is currently a Postdoctoral Researcher in the Laboratory for Mathematical Neuroscience, Brain Science Institute, RIKEN, Saitama, Japan. He was a Visiting Researcher at the Department of Electronics, the University of York, U.K. (October 2006 to March 2007) and a Guest Researcher at the Associate Institute for Signal Processing, the Technical University of Munich, Germany (August to November 2008). His research interests include mathematical adaptive signal processing, constrained /sparse optimization, and their applications to acoustic/communication systems.

Dr. Yukawa is a member of the Institute of Electrical, Information and Communication Engineers (IEICE) of Japan, and he currently serves as an Associate Editor for the *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*. From April 2005 to March 2007, he was a recipient of the Research Fellowship of the Japan Society for the Promotion of Science (JSPS). He received the Excellent Paper Award from the IEICE in 2006, the Yasujiro Niwa Outstanding Paper Award from Tokyo Denki University in 2007, and the Ericsson Young Scientist Award from Ericsson Japan in 2009.

Rodrigo C. de Lamare (S'99–M'04) received the Diploma degree in electronic engineering from the Federal University of Rio de Janeiro (UFRJ) in 1998 and the M.Sc. and Ph.D. degrees, both in electrical engineering, from the Pontifical Catholic University of Rio de Janeiro (PUC-Rio) in 2001 and 2004, respectively.

From January 2004 to June 2005, he was a Postdoctoral Fellow at the Center for Studies in Telecommunications (CETUC) of PUC-Rio and from July 2005 to January 2006, he worked as a Postdoctoral Fellow at the Signal Processing Laboratory, UFRJ. Since January 2006 he has been with the Communications Research Group, Department of Electronics, University of York, U.K., where he is currently Lecturer in Communications Engineering. His research interests lie in communications and signal processing.

Isao Yamada (M'96–SM'06) received the B.E. degree in computer science from the University of Tsukuba, Ibaraki, Japan, in 1985 and the M.E. and Ph.D. degrees in electrical and electronic engineering from the Tokyo Institute of Technology, Tokyo, Japan, in 1987 and 1990, respectively.

Currently, he is a Professor in the Department of Communications and Integrated Systems, Tokyo Institute of Technology. From August 1996 to July 1997, he was a Visiting Associate Professor with Pennsylvania State University, State College. His current research interests are in mathematical signal processing, optimization theory, and nonlinear inverse problem.

Dr. Yamada is a member of the American Mathematical Society (AMS), the Society for Industrial and Applied Mathematics (SIAM), the Japan Society for Industrial and Applied Mathematics (JSIAM), the Institute of Electronics, Information and Communication Engineers (IEICE) of Japan, and the Society of Information Theory and its Applications (SITA). He has been an Associate Editor for several journals, including the *International Journal on Multidimensional Systems and Signal Processing* since 1997, the *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences* from 2001 to 2005, and the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—PART I: FUNDAMENTAL THEORY AND APPLICATIONS from 2006 to 2007. Currently, he is a member of the IEEE Signal Processing Theory and Methods Technical Committee as well as an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING. He received Excellent Paper Awards in 1991, 1995, 2006, and 2009, the Young Researcher Award in 1992, and the Achievement Award in 2009 from the IEICE; the ICF Research Award from the International Communications Foundation in 2004; the DoCoMo Mobile Science Award (Fundamental Science Division) from the Mobile Communication Fund in 2005; and the Fujino Prize from Tejima Foundation in 2008.