Robust Hierarchical-Optimization RLS Against Sparse Outliers

Konstantinos Slavakis, *Senior Member, IEEE*, and Sinjini Banerjee

***Abstract*—This letter fortifies the recently introduced hierarchical-optimization recursive least squares (HO-RLS) against outliers which contaminate infrequently linear-regression models. Outliers are modeled as nuisance variables and are estimated together with the linear filter/system variables via a sparsity-inducing (non-)convexly regularized least-squares task. The proposed outlier-robust HO-RLS builds on steepest-descent directions with a constant step size (learning rate), needs no matrix inversion (lemma), accommodates colored nominal noise of known correlation matrix, exhibits small computational footprint, and offers theoretical guarantees, in a probabilistic sense, for the convergence of the system estimates to the solutions of a hierarchical-optimization problem: Minimize a convex loss, which models a-priori knowledge about the unknown system, over the minimizers of the classical ensemble LS loss. Extensive numerical tests on synthetically generated data in both stationary and non-stationary scenarios showcase notable improvements of the proposed scheme over state-of-the-art techniques.**

***Index Terms*—RLS, robust, outliers, sparsity.**

I. Introduction

THE recursive least squares (RLS) has been a pivotal method in solving LS problems in adaptive filtering and system identification [1], with a reach that extends also into contemporary learning tasks, such as solving large-scale LS problems in online learning, *e.g.*, [2]. Nevertheless, the performance of RLS (LS estimators in general) deteriorates in the presence of outliers, *i.e.*, data or noise not adhering to a nominal data-generation model [3]. This work focuses on outliers that contaminate infrequently data models, *e.g.*, impulse noise [4]–[6].

Methods that strengthen RLS against outliers have been reported in [7]–[13]. Propelled by robust-regression arguments [3], studies [7]–[9] utilize M-estimate losses instead of typical LS ones to penalize system-output errors. The *non*-recursive algorithm of [7] employs Huber's loss, while the recursive schemes of [8] and [9] build on Hampel's three-part redescending objective [3] and a modified Huber's loss, respectively. The numerical tests of [7] show that [7] outperforms median filters [14], a classical solution to mitigate impulse noise, while those of [8], [9] show that [8], [9] are more effective than order-statistics techniques [15]. Notwithstanding, noise and error-filtering statistics need to be known a-priori to set the parameters of Hampel's objective in [8], while the solutions of [8], [9] to the M-estimate normal equations are built on the assumption that filter's estimates do not change significantly for certain amounts of time [10]. Studies [10], [11] update filter's estimates by minimizing weighted LS error costs subject to "ball" constraints to prevent from large perturbations which may be inflicted by outliers: A Euclidean-ball constraint onto filter estimates is advocated by [10], while an ℓ_1-ball constraint onto the RLS gain vectors is utilized in [11]. The numerical tests of [10], [11] demonstrate the improved performance of [10], [11] over [8], [9].

Outliers are modeled as nuisance variables and are jointly estimated with the filter's coefficients in [12], [13]. To address identifiability issues in the estimation task due to the ever-growing number of the outlier unknowns with recursions (time instances), filter and outlier vectors are updated per recursion via the minimization of an LS data-fit loss plus a surrogate of the ℓ_0-norm of the outlier vector to exploit sparsity: The ℓ_1-norm is utilized in [12], while non-convex surrogates in [13]. The composite minimization task is solved in an alternating way: First, the outlier vector is updated, and then the classical RLS is used to update the filter's estimates. Methods [12], [13] accommodate also colored nominal noise of known correlation matrix to avoid any prewhitening that would spread the outliers in the nominal data, adding further complication to the challenge of outlier removal. Numerical tests in [12], [13] show the improved performance of [12], [13] over [10], [11].

This short letter follows the path of [12], [13] to model outliers as nuisance variables, but employs the recently introduced hierarchical(-optimization) recursive least squares (HO-RLS) [16] to update filter coefficients instead of the classical RLS. Unlike RLS, which propels all of [7]–[13], HO-RLS provides a way to quantify side information about the system since it solves a hierarchical-optimization problem: Minimize a convex loss, which models the available side information, over the minimizers of the classical ensemble LS data-fit loss. The proposed outlier-robust HO-RLS builds on steepest-descent directions with a constant step size (learning rate), needs *no* matrix inversion (lemma), exhibits similar computational complexity with the implementations in [12], [13], accommodates colored noise of known covariance matrix without any prewhitening, and offers theoretical guarantees, in a probabilistic sense, for the convergence of the filter/system estimates to the solution of the aforementioned hierarchical-optimization task. Extensive numerical tests on synthetically generated data, in both stationary and non-stationary scenarios, showcase notable improvements of the proposed scheme over the state-of-the-art [12], [13].

Manuscript received October 11, 2019; revised December 19, 2019; accepted December 23, 2019. Date of publication December 31, 2019; date of current version January 30, 2020. This work was supported by the NSF under Grant CIF 1718796. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Woon-Seng Gan. *(Corresponding author: Konstantinos Slavakis.)*

The authors are with the Department of Electrical Engineering, University at Buffalo, The State University of New York, Buffalo, NY 14260 USA (e-mail: kslavaki@buffalo.edu; sinjinib@buffalo.edu).

Digital Object Identifier 10.1109/LSP.2019.2963188

II. The Problem and State-of-The-Art Solutions

With the positive integer n denoting both discrete time and recursion index, the following data model is considered:

$$\mathbf{y}_n = \mathbf{F}_*\mathbf{x}_n + \mathbf{o}_n + \mathbf{v}_n \,, \tag{1}$$

where the $L \times P$ matrix $\mathbf{F}_*$ is the wanted filter/system, the $L \times 1$ vector $\mathbf{y}_n$ collects the output data, the $P \times 1$ vector $\mathbf{x}_n$ gathers the input ones, vector $\mathbf{o}_n$ models the outlier data, $\mathbf{v}_n$ stands for zero-mean (colored) noise with correlation matrix $\mathbf{R}_{vv} := \mathbb{E}\{\mathbf{v}_n\mathbf{v}_n^\intercal\}$ that is assumed to stay constant $\forall n$, $\mathbb{E}\{\cdot\}$ is the expectation operator with respect to (w.r.t.) a probability space [17], and $\intercal$ denotes vector/matrix transposition. The multiple-input-multiple-output model (1) is chosen here to offer a general model that is able to capture data-generation mechanisms in numerous modern application domains, *e.g.*, [18], [19] and [1, p. 647]. To save space, $\mathbf{y}_n$, $\mathbf{x}_n$, $\mathbf{o}_n$ and $\mathbf{v}_n$ represent both random variables (RVs) and their realizations. Furthermore, equality in (1) is assumed to hold almost surely (a.s.) w.r.t. the underlying probability space. Since outliers $(\mathbf{o}_n)_{n\geq 1}$ are assumed to appear infrequently in (1), vector $\mathbf{o}_n$ can be considered to be sparse, *i.e.*, most of its entries are zero $\forall n$. The input-output data become available to the user sequentially: The pair $(\mathbf{x}_n, \mathbf{y}_n)$ is revealed to the user at time n. The problem is to devise an iterative algorithm to estimate $\mathbf{F}_*$ from the available $(\mathbf{x}_\nu, \mathbf{y}_\nu)_{\nu=1}^n$, $\forall n$, and $\mathbf{R}_{vv}$.

For every n, the typical LS estimator

$$(\hat{\mathbf{F}}_{n+1}, \{\hat{\mathbf{o}}_{\nu,n}\}_{\nu=1}^n) \in \arg\min_{(\boldsymbol{F},\{\boldsymbol{o}_\nu\}_{\nu=1}^n)} \sum_{\nu=1}^n \|\mathbf{y}_\nu - \boldsymbol{o}_\nu - \boldsymbol{F}\mathbf{x}_\nu\|^2_{\mathbf{R}_{vv}^{-1}} , \tag{2}$$

does not offer significant help, since $\hat{\mathbf{F}}_{n+1} := \mathbf{0}$ and $\{\hat{\mathbf{o}}_{\nu,n} := \mathbf{y}_\nu\}_{\nu=1}^n$ qualify as solutions to (2). Motivated by classical arguments, *e.g.*, [1, §29.6], the weighted norm $\|\mathbf{a}\|^2_{\mathbf{R}_{vv}^{-1}} := \mathbf{a}^\intercal\mathbf{R}_{vv}^{-1}\mathbf{a}$ is introduced in (2) to handle entries of the error vector unequally via the available $\mathbf{R}_{vv}^{-1}$. To avoid trivial solutions, a popular way is to form a regularized LS estimation task

$$(\hat{\mathbf{F}}_{n+1}, \{\hat{\mathbf{o}}_{\nu,n}\}_{\nu=1}^n) \in \arg\min_{(\boldsymbol{F},\{\boldsymbol{o}_\nu\}_{\nu=1}^n)} \sum_{\nu=1}^n \times \left[\tfrac{1}{2}\|\mathbf{y}_\nu - \boldsymbol{F}\mathbf{x}_\nu - \boldsymbol{o}_\nu\|^2_{\mathbf{R}_{vv}^{-1}} + \lambda_\nu\rho(\boldsymbol{o}_\nu)\right] , \tag{3}$$

where $\rho(\cdot)$ is a sparsity-inducing loss with user-defined weights $\lambda_\nu > 0$. However, this path seems to be impractical since the $LP + nL$ number of unknown variables at time n raises insurmountable computational obstacles when solving (3) at large time instances n.

To address this "curse of dimensionality," studies [12], [13] adopt the following recursions for all $n \geq n_0 + 1$,

$$\hat{\mathbf{o}}_n \in \arg\min_{\boldsymbol{o}} \tfrac{1}{2}\|\mathbf{y}_n - \hat{\mathbf{F}}_n\mathbf{x}_n - \boldsymbol{o}\|^2_{\mathbf{R}_{vv}^{-1}} + \lambda_n\rho(\boldsymbol{o}) , \tag{4a}$$

$$\hat{\mathbf{F}}_{n+1} := \text{RLS}\left(\hat{\mathbf{F}}_n, (\mathbf{x}_n, \mathbf{y}_n - \hat{\mathbf{o}}_n)\right) , \tag{4b}$$

where n_0 is a user-defined time instance, $\text{RLS}(\hat{\mathbf{F}}_n, (\mathbf{x}_n, \mathbf{y}_n - \hat{\mathbf{o}}_n))$ denotes the classical RLS updates, with the newly introduced data pair at time n being $(\mathbf{x}_n, \mathbf{y}_n - \hat{\mathbf{o}}_n)$, and with $\mathbf{R}_{vv}^{-1}$ incorporated in the RLS formulae. A warm start is achieved by solving the offline task (3) for $(\hat{\mathbf{F}}_{n_0}, \{\hat{\mathbf{o}}_\nu\}_{\nu=1}^{n_0-1})$, where $n_0 - 1$ is used in the place of n. Loss $\rho(\cdot)$ takes the form of the ℓ_1-norm in [12], rendering (3) and (4a) typical LASSO tasks [20], while non-convex $\rho(\cdot)$ are promoted in [13] and non-convex optimization solvers, *e.g.*, [21], are required to solve (4a).

III. The Proposed Algorithm

This work follows [12], [13], but instead of the classical RLS in (4b), the recently introduced hierarchical-optimization recursive least squares (HO-RLS) [16] is used. In the current context, HO-RLS solves the following HO task: Given the convex function $g : \mathbb{R}^L \to \mathbb{R} \cup \{+\infty\}$, which is generally non-differentiable, find

$$\arg\min\nolimits_{\boldsymbol{F}} \; g(\boldsymbol{F}) \quad \text{s.to } \boldsymbol{F} \in \arg\min_{\boldsymbol{F}'} \mathbb{E}\left\{\tfrac{1}{2}\|\mathbf{y}_n - \mathbf{o}_n - \boldsymbol{F}'\mathbf{x}_n\|^2_{\mathbf{R}_{vv}^{-1}}\right\} . \tag{5}$$

Unlike the RLS in (4b), $g(\cdot)$ is able to quantify any a-priori knowledge (side information) about the system. For example, if $\mathbf{F}_*$ is known to be sparse, $g(\cdot) := \|\cdot\|_1$ can be used to promote sparse solutions in (5). To approximate the expectation in (5), the following sample-average loss is adopted: $\forall n \geq n_0$, $l_n(\mathbf{F}) := [1/(2\Gamma_n)] \sum_{\nu=n_0}^n \gamma^{n-\nu}\|\mathbf{y}_\nu - \hat{\mathbf{o}}_\nu - \mathbf{F}\mathbf{x}_\nu\|^2_{\mathbf{R}_{vv}^{-1}}$, with a "forgetting factor" $\gamma \in (0, 1]$ to mimic the classical exponentially-weighted RLS scheme [1, §30.6], and $\Gamma_n := \sum_{\nu=n_0}^n \gamma^{n-\nu}$. The outlier vector $\mathbf{o}_n$ is replaced by its estimate $\hat{\mathbf{o}}_n$ in $l_n(\cdot)$. Notice that in the case of $g(\cdot) := 0$ and $\hat{\mathbf{o}}_n := \mathbf{0}$, the loss $l_n(\cdot)$ is along the lines of (2) with $\boldsymbol{o}_\nu := \mathbf{0}$.

Being an offspring of the stochastic Fejér-monotone hybrid steepest-descent method [16], HO-RLS is based on the gradient of $l_n(\cdot)$. To this end, given two stochastic processes $(\mathbf{a}_n)_n$ and $(\mathbf{b}_n)_n$, define $\forall n \geq n_0$, $\mathbf{R}_{ab,n} := (1/\Gamma_n) \sum_{\nu=n_0}^n \gamma^{n-\nu}\mathbf{a}_\nu\mathbf{b}_\nu^\intercal$, so that for the processes $(\mathbf{x}_n)_n$, $(\mathbf{y}_n)_n$, and $(\hat{\mathbf{o}}_n)_n$ under study, define $\mathbf{R}_{xx,n}$, $\mathbf{R}_{yx,n}$, and $\mathbf{R}_{\hat{o}x,n}$. Then, the gradient of l_n can be expressed as $\nabla l_n(\mathbf{F}) = \mathbf{R}_{vv}^{-1}(\mathbf{F}\mathbf{R}_{xx,n} - \mathbf{R}_{yx,n} + \mathbf{R}_{\hat{o}x,n})$, $\forall \mathbf{F}$. Following the arguments of [16, Algorithm 1 and (5a)], the previous gradient information is incorporated into HO-RLS via the mapping $T_n(\mathbf{F}) := \mathbf{F} - (1/\varpi_n)\nabla l_n(\mathbf{F})$, $\forall \mathbf{F}$, with $\varpi_n \geq \|\mathbf{R}_{vv}^{-1}\| \, \|\mathbf{R}_{xx,n}\|$, to produce Algorithm 1.

Any off-the-shelf solver can be employed to solve the sub-task in Line 8. Several solvers are explored here: **i**) The alternating direction method of multipliers (ADMM) [22], [23] and **ii**) the recently developed Fejér-monotone hybrid steepest descent method (FMHSDM) [24], which is the deterministic precursor of [16], in the case of $\rho(\cdot) := \|\cdot\|_1$; as well as **iii**) the general iterative shrinkage and thresholding (GIST) algorithm [21] in the case where $\rho(\cdot)$ takes the form of any non-convex surrogate of the ℓ_0-norm, such as the minimax concave penalty (MCP) [13]. As it will be seen in Section V, only a small number of iterations of the previous solvers suffice to provide the estimates $\hat{\mathbf{o}}_n$ in Line 8.

In Lines 6 and 12, the proximal operator is defined as $\text{Prox}_{\lambda g}(\mathbf{F}) := \arg\min_{\boldsymbol{F}}(1/2)\|\mathbf{F} - \boldsymbol{F}\|^2_{\text{Fr}} + \lambda g(\boldsymbol{F}), \forall \mathbf{F}$, where $\|\cdot\|_{\text{Fr}}$ denotes the Frobenius norm. Clearly, if $g(\cdot) = 0$, then $\text{Prox}_{\lambda g}(\mathbf{F}) = \mathbf{F}$. The computational complexity of Line 12 depends on the loss $g(\cdot)$. For example, if $g(\cdot) = \|\cdot\|_1$, then $\text{Prox}_{\lambda g}(\cdot)$ operates on each entry of its matrix argument separately, and boils down to the classical soft-thresholding mapping [25, Example 4.9]. Lines 9 and 10 realize the power method [26] to provide running estimates of the spectral norm $\|\mathbf{R}_{xx,n}\|$, and the user-defined $\epsilon_\varpi > 0$ helps to provide an overestimate ϖ_n of that spectral norm.

Algorithm 1: Outlier-robust (OR-)HO-RLS

Data : $(\mathbf{x}_n, \mathbf{y}_n)_{n\geq 0}$, $\mathbf{R}_{vv}^{-1}$.
User's input: n_0, $\alpha \in (0.5, 1]$, $\lambda \in \mathbb{R}_{>0}$, $\epsilon_\varpi \in \mathbb{R}_{>0}$.
Output : Sequence $(\hat{\mathbf{F}}_n)_{n\geq n_0}$.

1 **Initialization**
2 $(\hat{\mathbf{F}}_{n_0}, \{\hat{\mathbf{o}}_{\nu,n_0}\}_{\nu=1}^{n_0}) \in \arg\min_{(\boldsymbol{F}, \{\boldsymbol{o}_\nu\}_{\nu=1}^{n_0})} \sum_{\nu=1}^{n_0} \Big[\frac{1}{2n_0} \|\mathbf{y}_\nu - \boldsymbol{F}\mathbf{x}_\nu - \boldsymbol{o}_\nu\|^2_{\mathbf{R}_{vv}^{-1}} + \lambda\rho(\boldsymbol{o}_\nu)\Big]$.
3 $\hat{\mathbf{o}}_{n_0} := \hat{\mathbf{o}}_{n_0,n_0}$.
4 $\varpi_{n_0} := \|\mathbf{R}_{vv}^{-1}\| \, \|\mathbf{R}_{xx,n_0}\| + \epsilon_\varpi$.
5 $\hat{\mathbf{F}}_{n_0+1/2} := \hat{\mathbf{F}}_{n_0} - \frac{\alpha}{\varpi_{n_0}} \mathbf{R}_{vv}^{-1} (\hat{\mathbf{F}}_{n_0}\mathbf{R}_{x,n_0} - \mathbf{R}_{yx,n_0} + \mathbf{R}_{\hat{o}x,n_0})$.
6 $\hat{\mathbf{F}}_{n_0+1} := \mathrm{Prox}_{\lambda g}(\hat{\mathbf{F}}_{n_0+1/2})$.
7 **for** $n = n_0 + 1$ **to** $+\infty$ **do**
8 $\hat{\mathbf{o}}_n \in \arg\min_{\boldsymbol{o}} \frac{1}{2}\|\mathbf{y}_n - \hat{\mathbf{F}}_n\mathbf{x}_n - \boldsymbol{o}\|^2_{\mathbf{R}_{vv}^{-1}} + \lambda\rho(\boldsymbol{o})$.
9 $\mathbf{q}_n := \mathbf{R}_{xx,n}\mathbf{p}_{n-1}$; $\mathbf{p}_n := \mathbf{q}_n / \|\mathbf{q}_n\|$.
10 $\varpi_n := \|\mathbf{R}_{vv}^{-1}\| \cdot (\mathbf{p}_n^\intercal \mathbf{R}_{xx,n}\mathbf{p}_n) + \epsilon_\varpi$.
11 $\hat{\mathbf{F}}_{n+1/2} := \hat{\mathbf{F}}_n + \hat{\mathbf{F}}_{n-1/2} - \hat{\mathbf{F}}_{n-1} + \frac{\alpha}{\varpi_{n-1}} \mathbf{R}_{vv}^{-1} (\hat{\mathbf{F}}_{n-1}\mathbf{R}_{xx,n-1} - \mathbf{R}_{yx,n-1} + \mathbf{R}_{\hat{o}x,n-1}) - \frac{1}{\varpi_n} \mathbf{R}_{vv}^{-1} (\hat{\mathbf{F}}_n\mathbf{R}_{xx,n} - \mathbf{R}_{yx,n} + \mathbf{R}_{\hat{o}x,n})$.
12 $\hat{\mathbf{F}}_{n+1} := \mathrm{Prox}_{\lambda g}(\hat{\mathbf{F}}_{n+1/2})$.

Unlike the classical RLS (Newton's method in general) [1, §9.8], HO-RLS uses $\nabla l_n(\cdot)$ in a way that avoids any matrix inversion (lemma). The system updates which take place within Lines 9 and 12 of Algorithm 1 show a small computational footprint, with the main burden being the matrix multiplications in Line 11. Multiplications in Line 11 amount only to $\mathbf{R}_{vv}^{-1}(\hat{\mathbf{F}}_n\mathbf{R}_{xx,n} - \mathbf{R}_{yx,n} + \mathbf{R}_{\hat{o}x,n})$, since $\mathbf{R}_{vv}^{-1}(\hat{\mathbf{F}}_{n-1}\mathbf{R}_{xx,n-1} - \mathbf{R}_{yx,n-1} + \mathbf{R}_{\hat{o}x,n-1})$ is already available from the previous recursion.

IV. Convergence Analysis

The convergence analysis of Algorithm 1 is based on the following set of assumptions.

Assumption 1:

i) $\mathbf{R}_{xx,n} \xrightarrow{\text{a.s.}}_n \mathbf{R}_{xx} := \mathbb{E}\{\mathbf{x}_{n'}\mathbf{x}_{n'}^\intercal\}$ and $\mathbf{R}_{yx,n} \xrightarrow{\text{a.s.}}_n \mathbf{R}_{yx} := \mathbb{E}\{\mathbf{y}_{n'}\mathbf{x}_{n'}^\intercal\}$, $\forall n' \geq 1$, where $\xrightarrow{\text{a.s.}}_n$ denotes a.s. convergence [17].

ii) Both $(\mathbf{x}_n)_n$ and $(\mathbf{v}_n)_n$ are zero-mean processes, and $(\mathbf{x}_n)_n$ is independent of $(\mathbf{o}_n)_n$ and $(\mathbf{v}_n)_n$.

iii) There exists $C \in \mathbb{R}_{>0}$ s.t. $\mathbb{E}\{\|\mathbf{R}_{yx,n}\|^2\} \leq C, \forall n$.

iv) With $\mathcal{F}_n := \sigma(\{\hat{\mathbf{F}}_\nu\}_{\nu=n_0}^n)$ denoting the filtration (σ-algebra) generated by $\{\hat{\mathbf{F}}_\nu\}_{\nu=n_0}^n$, and $\mathbb{E}_{|\mathcal{F}_n}\{\cdot\}$ being the conditional expectation given $\mathcal{F}_n$ [17], assume that $\mathbb{E}_{|\mathcal{F}_n}\{\mathbf{R}_{xx,\nu}\} = \mathbf{R}_{xx}$ and $\mathbb{E}_{|\mathcal{F}_n}\{\mathbf{R}_{yx,\nu}\} = \mathbf{R}_{yx}$, $\forall \nu \in \{n_0, \ldots, n\}$.

v) Consider a sequence $(\mathbf{Z}_n, \boldsymbol{\Xi}_n)_n$ s.t. $(\mathbf{Z}_n, \boldsymbol{\Xi}_n) \in \mathrm{Dom}\, g \times \partial g(\mathbf{Z}_n)$, $\forall n$, where $\mathrm{Dom}\, g := \{\mathbf{F} \in \mathbb{R}^{L\times P} \mid g(\mathbf{F}) < +\infty\}$, $\partial g(\cdot)$ stands for the subdifferential mapping of g, *i.e.*, $\partial g(\mathbf{Z}) := \{\boldsymbol{\Xi} \in \mathbb{R}^{L\times P} \mid g(\mathbf{Z}) + \mathrm{Trace}[(\mathbf{F} - \mathbf{Z})^\intercal \boldsymbol{\Xi}] \leq g(\mathbf{F}), \forall \mathbf{F} \in \mathbb{R}^{L\times P}\}$, and $\mathrm{Trace}(\cdot)$ denotes the trace of a square matrix. If $(\mathbf{Z}_n)_n$ is bounded a.s., then $(\boldsymbol{\Xi}_n)_n$ is also bounded a.s. Moreover, if $(\mathbb{E}\{\|\mathbf{Z}_n\|^2_{\mathrm{Fr}}\})_n$ is bounded, then $(\mathbb{E}\{\|\boldsymbol{\Xi}_n\|^2_{\mathrm{Fr}}\})_n$ is also bounded. ■

Assumption 1(i) is motivated by ergodicity arguments [27], and conditions which suffice to guarantee such an assumption can be based on laws of large numbers through statistical independency or mixing conditions [28]. Bounded-moment assumptions, such as Assumption 1(iii), appear frequently in stochastic approximation *e.g.*, [29, p. 126, (A2.1)]. Assumption 1(iv) is a sufficient condition for the more relaxed and technical one in [16, Assumption 6]. Due to space limitations, the stronger Assumption 1(iv) is used here. Moreover, many candidate losses for $g(\cdot)$ satisfy Assumption 1(v). Examples are: **i)** the zero loss; **ii)** the indicator function $\iota_C(\cdot)$, used to enforce closed convex constraints $C \subset \mathbb{R}^{L\times P}$ onto the desired solutions, with definition $\iota_C(\mathbf{F}) = 0$, if $\mathbf{F} \in C$, while $\iota_C(\mathbf{F}) = +\infty$, if $\mathbf{F} \notin C$; **iii)** $\|\cdot\|_1$; and **iv)** $\|\cdot\|^2$.

The following theorem is a consequence of Corollary 1 in [16]. The subsequent proof translates the arguments of [16] into the current context.

Theorem 1: Consider a large integer $n_\sharp$ and set $\hat{\mathbf{o}}_n$ equal to zero in Line 8 of Algorithm 1, $\forall n \geq n_\sharp$. Moreover, set $\varpi_n := \varpi \geq \max\{\|\mathbf{R}_{vv}^{-1}\| \, \|\mathbf{R}_{xx}\|, \|\mathbf{R}_{vv}^{-1}\| \, \|\mathbf{R}_{xx,n}\|\}$, $\forall n > n_\sharp$.

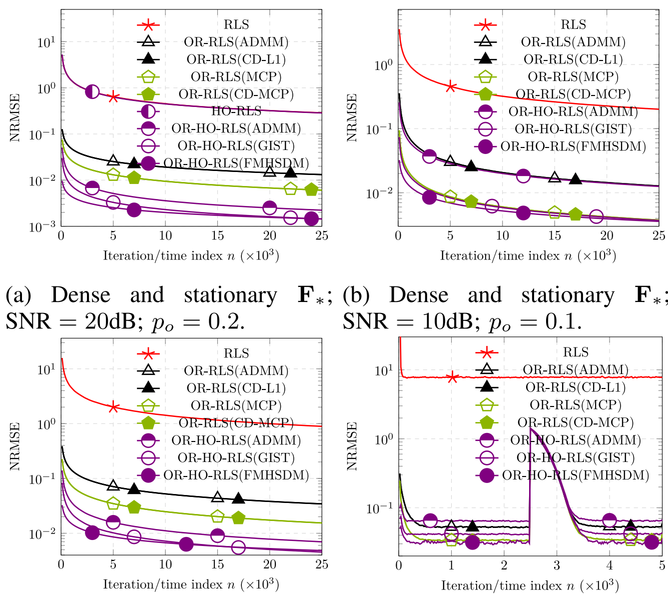
i) Under Assumptions 1(i) to 1(v), the set of cluster points of the sequence $(\hat{\mathbf{F}}_n)_{n\geq n_0}$ is non-empty, and any of those cluster points solves (5) a.s.

ii) If $g(\cdot) = 0$ and $\mathbf{R}_{xx}$ is positive definite, then under Assumptions 1(i) to 1(iv), the sequence $(\hat{\mathbf{F}}_n)_{n\geq n_0}$ generated by Algorithm 1 converges a.s. to the unknown $\mathbf{F}_*$. ■

Proof: **i)** First of all, notice that under the hypotheses of the theorem, $\mathbf{R}_{\hat{o}x,n} = (1/\Gamma_n) \sum_{\nu=n_0}^{n_\sharp - 1} \gamma^{n-\nu} \hat{\mathbf{o}}_\nu \mathbf{x}_\nu^\intercal$. Since $\lim_{n\to\infty} \Gamma_n = +\infty$, if $\gamma = 1$, and $\lim_{n\to\infty} \Gamma_n = 1/(1-\gamma)$ as well as $\lim_{n\to\infty} \gamma^{n-\nu} = 0$, if $\gamma < 1$, it can be verified that $\lim_{n\to\infty} \mathbf{R}_{\hat{o}x,n} = \mathbf{0}$ a.s. According to [16, Thm. 1 and Cor. 1], only Assumptions 2, 6, 7(ii) and 8 of [16] need to be verified to establish Theorem 1. Assumptions 3 and 4 of [16] are trivially satisfied due to the construction of problem (5). The ergodicity Assumption 2 of [16] is satisfied via Assumption 1(i), Assumption 7(ii) of [16] is guaranteed by Assumption 1(iii), and Assumption 8 by Assumption 1(v). The proof of [16] adapts to the present context via the following mappings: By the definition of T_n in Section III, let $Q_n(\mathbf{F}) := T_n(\mathbf{F}) - (1/\varpi_n)\mathbf{R}_{vv}^{-1}\mathbf{R}_{yx,n}$, while $T(\mathbf{F}) := Q(\mathbf{F}) + (1/\varpi)\mathbf{R}_{vv}^{-1}\mathbf{R}_{yx}$, with $Q(\mathbf{F}) := \mathbf{F} - (1/\varpi)\mathbf{R}_{vv}^{-1}\mathbf{F}\mathbf{R}_{xx}$ and $\varpi \geq \|\mathbf{R}_{vv}^{-1}\| \, \|\mathbf{R}_{xx}\|$. The application now of the conditional expectation $\mathbb{E}_{|\mathcal{F}_n}\{\cdot\}$ to the terms that appear between (19b) and (19c) of [16] and define ϑ_n, yields $\mathbb{E}_{|\mathcal{F}_n}\{\vartheta_n\} = 0$, a.s., and thus Assumption 6 of [16] is satisfied by setting $\psi = 0$, a.s. Therefore, [16, Thm. 1 and Cor. 1] establish Theorem 1.

ii) Multiplying both sides of (1) from the right side by $\mathbf{x}_n^\intercal$ and by applying $\mathbb{E}\{\cdot\}$, it can be verified via Assumption 1(ii) that $\mathbf{F}_* = \mathbf{R}_{xy}\mathbf{R}_{xx}^{-1}$. The loss in the constraint of (5) becomes $\mathrm{Trace}\{\mathbf{F}^\intercal\mathbf{R}_{vv}^{-1}\mathbf{F}\mathbf{R}_{xx} - 2\mathbf{F}^\intercal\mathbf{R}_{vv}^{-1}\mathbf{R}_{xy} + \mathbf{R}_{vv}^{-1}\mathbf{R}_{yy}\}$. Hence, the minimizer $\mathbf{F}_{\text{opt}}$ of this loss satisfies the normal equations $\mathbf{R}_{vv}^{-1}\mathbf{F}_{\text{opt}}\mathbf{R}_{xx} = \mathbf{R}_{vv}^{-1}\mathbf{R}_{xy} \Rightarrow \mathbf{F}_{\text{opt}} = \mathbf{R}_{xy}\mathbf{R}_{xx}^{-1} = \mathbf{F}_*$. Since the minimizer is unique, Theorem 1(i) suggests that all cluster points of $(\hat{\mathbf{F}}_n)_n$ coincide with $\mathbf{F}_{\text{opt}} = \mathbf{F}_*$. ■

Setting $\hat{\mathbf{o}}_n$ equal to zero, for all sufficiently large n, has been also used in the convergence analysis of [12]. Moreover, the assumption on the positive definiteness of $\mathbf{R}_{xx}$ in Theorem 1(ii) holds true for any regular stochastic process, *e.g.*, a process with non-zero innovation [30, Prob. 2.2].



(a) Dense and stationary $\mathbf{F}_*$; SNR $= 20$dB; $p_o = 0.2$.
(b) Dense and stationary $\mathbf{F}_*$; SNR $= 10$dB; $p_o = 0.1$.
(c) Sparse and stationary $\mathbf{F}_*$. SNR $= 20$dB; $p_o = 0.2$.
(d) Dense and non-stationary $\mathbf{F}_*$; SNR $= 20$dB; $p_o = 0.2$.

Fig. 1. NRMSE values vs. iteration / time indices.

V. Numerical Tests

To validate the contributions of this letter, extensive numerical tests on synthetically generated data were conducted vs. the classical RLS [1] and the state-of-the-art [12], [13]. Two variants of [12] were employed to tackle (4a) with $\rho(\cdot) := \|\cdot\|_1$: **i)** The OR-RLS(ADMM) that employs ADMM [22], [23], and **ii)** a coordinate-descent approach OR-RLS(CD-L1) [12]. For [13], two variants were used to solve (4a) with $\rho(\cdot)$ being the MCP: **i)** The OR-RLS(MCP) that employs the GIST method [21], and **ii)** its coordinate-descent flavor OR-RLS(CD-MCP) [13]. OR-HO-RLS appears in three flavors, namely OR-HO-RLS(ADMM), OR-HO-RLS(GIST) and OR-HO-RLS(FMHSDM), depending on the solver of Line 8 in Algorithm 1. The outlier-agnostic HO-RLS is also considered in Figure 1, with $g(\cdot) := 0$ in (5) and $\hat{\mathbf{o}}_n := \mathbf{0}, \forall n$. The performance metric is the normalized root mean squared error NRMSE $:= \|\hat{\mathbf{F}}_n - \mathbf{F}_*\|_{\text{Fr}} / \|\mathbf{F}_*\|_{\text{Fr}}$, where $\hat{\mathbf{F}}_n$ stands for the estimate of any of the employed methods at time n. The software code was written in Julia (ver. 1.0.3) [31]. The Julia package JuMP [32] was utilized, along with the Gurobi solver [33], to solve (3) for [12], [13] ($\rho(\cdot) = \|\cdot\|_1$, $n = 500$), and Line 2 of Algorithm 1 for OR-HO-RLS ($\rho(\cdot) = \|\cdot\|_1$, $n_0 = 500$) as a warm start. Warm-start iterations and times are not included in any of the subsequent numbers and figures. One hundred independent tests were run and averaged values are reported. The parameters of all methods, including those of any off-the-shelf solver, were carefully tuned s.t. optimal performance is achieved in all scenarios. In all cases, $\alpha := 0.5$ and $\epsilon_{\varpi} := 5e - 2$ which help Algorithm 1 reach a "sweet spot" with regards to convergence speed and estimation accuracy.

In all tests, $P = 20$ and $L = 10$, and noise $(\mathbf{v}_n)_n$ is modeled as zero-mean and colored Gaussian, generated by an autoregressive (AR) model, where the state matrix is randomly generated with maximum singular value 0.95. To generate sparse outliers, the entries of $\mathbf{o}_n$ (also across time) are modeled as independent and identically distributed (IID) Bernoulli RVs with parameter p_o. Following [12], the nonzero entries of $\mathbf{o}_n$ are drawn from a uniform distribution with zero mean and variance 1e4. Although OR-HO-RLS behaves well also in cases with smaller outliers variance, 1e4 was chosen to highlight OR-HO-RLS's performance in extreme scenarios. The entries of $(\mathbf{x}_n)_n$ are modeled (also across time) as IID Gaussian RVs with zero mean and unit variance.

TABLE I
Average Values $\pm$ Standard Deviations of the Run Times (in Secs) Per Iteration for the Scenario of Figure 1a

Algorithm	Time (secs) / iteration
RLS	$4.12e{-3} \pm 7.86e{-7}$
OR-RLS(ADMM)	$3.57e{-3} \pm 3.08e{-7}$
OR-RLS(CD-L1)	$7.47e{-3} \pm 1.23e{-6}$
OR-RLS(MCP)	$1.51e{-2} \pm 5.79e{-6}$
OR-RLS(CD-MCP)	$8.54e{-3} \pm 1.68e{-6}$
OR-HO-RLS(ADMM)	$\mathbf{1.11e{-3} \pm 3.95e{-8}}$
OR-HO-RLS(GIST)	$1.24e{-2} \pm 4.06e{-6}$
OR-HO-RLS(FMHSDM)	$2.79e{-3} \pm 2.52e{-7}$

Figure 1a and 1b refer to the "stationary" case where $\mathbf{F}_*$ stays fixed $\forall n$. Matrix $\mathbf{F}_*$ is "dense," *i.e.*, with no zero entries. The entries are IID Gaussian RVs with zero mean and unit variance. Figure 1a considers the case of (SNR $= 20$ dB, $p_o = 0.2$), while Figure 1b the case of (SNR $= 10$ dB, $p_o = 0.1$). In these scenarios, $g(\cdot) := 0$ in Algorithm 1. Moreover, in all stationary scenarios $\gamma = 1$. All robust techniques outperform the classical RLS and the outlier-agnostic HO-RLS. OR-RLS(ADMM) and OR-RLS(MCP) perform identically to their coordinate-descent counterparts, while all flavors of OR-HO-RLS outperform all other outlier-robust schemes. This behavior is also observed in Figure 1b, but OR-RLS(MCP) and OR-RLS(CD-MCP) seem to reach the levels of OR-HO-RLS(GIST) and OR-HO-RLS(FMHSDM). It is worth noticing here that OR-HO-RLS(FMHSDM) outperforms OR-HO-RLS(ADMM) even though FMHSDM and ADMM solve the same ℓ_1-norm penalized LS task in Line 8 of Algorithm 1.

A sparse stationary $\mathbf{F}_*$ is examined in Figure 1c. Non-zero values are placed randomly at 10% of the entries of $\mathbf{F}_*$, while the rest of the entries are zero. Here, $g(\cdot) := \|\cdot\|_1$ in Algorithm 1. All curves exhibit similar behavior to that in Figure 1a and 1b. In Figure 1d, a sudden system change is introduced at time 2,500, by randomly re-initializing $\mathbf{F}_*$, to examine how fast the employed algorithms adapt to the change. Here, $g(\cdot) := 0$ in Algorithm 1. In this scenario, $\gamma = 0.97$ for all methods. Figure 1d shows that OR-HO-RLS(FMSHDM) exhibits fast adaptation to system changes while scoring the lowest levels of NRMSE among all methods.

Table I lists the computation times (in secs) per "iteration" on an Intel(R) Xeon(R) CPU E5-2650v4 which operates at 2.20 GHz with 256 GB RAM. "Iteration" refers to a single pass through (4) for [12], [13], and to the iterations within Lines 8 and 12 of Algorithm 1 for OR-HO-RLS. Each iteration includes also the 100 recursions of the off-the-shelf solvers which are employed to generate the estimates $\hat{\mathbf{o}}_n$ in (4a) and in Line 8 of Algorithm 1. It can be seen that the proposed OR-HO-RLS(ADMM) and OR-HO-RLS(FMHSDM) are the fastest solutions among all methods.

REFERENCES

[1] A. H. Sayed, *Adaptive Filters*. Hoboken, NJ, USA: Wiley, 2008.
[2] G.-B. Huang, N. Liang, H.-J. Rong, P. Saratchandran, and N. Sundararajan, "On-line sequential extreme learning machine," in *Proc. Intern. Conf. Comput. Intell.*, vol. 2005, Jan. 2005, pp. 232–237.
[3] P. J. Rousseeuw and A. M. Leroy, *Robust Regression and Outlier Detection*. New York, NY, USA: Wiley, 1987.
[4] R. Fano, "A theory of impulse noise in telephone networks," *IEEE Trans. Commun.*, vol. COM-25, no. 6, pp. 577–588, Jun. 1977.
[5] J. Lin, M. Nassar, and B. L. Evans, "Impulsive noise mitigation in powerline communications using sparse Bayesian learning," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 7, pp. 1172–1183, Jul. 2013.
[6] M. Nikolova, "A variational approach to remove outliers and impulse noise," *J. Math. Imag. Vis.*, vol. 20, no. 1, pp. 99–120, Jan. 2004.
[7] P. Petrus, "Robust Huber adaptive filter," *IEEE Trans. Signal Process.*, vol. 47, no. 4, pp. 1129–1133, Apr. 1999.
[8] Y.-X. Zou, S.-C. Chan, and T. S. Ng, "A recursive least M-estimate (RLM) adaptive filter for robust filtering in impulse noise," *IEEE Signal Process. Lett.*, vol. 7, no. 11, pp. 324–326, Nov. 2000.
[9] S.-C. Chan and Y.-X. Zou, "A recursive least M-estimate algorithm for robust adaptive filtering in impulsive noise: Fast algorithm and convergence performance analysis," *IEEE Trans. Signal Process.*, vol. 52, no. 4, pp. 975–991, Apr. 2004.
[10] L. Rey Vega, H. Rey, J. Benesty, and S. Tressens, "A fast robust recursive least-squares algorithm," *IEEE Trans. Signal Process.*, vol. 57, no. 3, pp. 1209–1216, Mar. 2009.
[11] M. Z. A. Bhotto and A. Antoniou, "Robust recursive least-squares adaptive-filtering algorithm for impulsive-noise environments," *IEEE Signal Process. Lett.*, vol. 18, no. 3, pp. 185–188, Mar. 2011.
[12] S. Farahmand and G. B. Giannakis, "Robust RLS in the presence of correlated noise using outlier sparsity," *IEEE Trans. Signal Process.*, vol. 60, no. 6, pp. 3308–3313, Jun. 2012.
[13] L. Xiao, M. Wu, J. Yang, and J. Tian, "Robust RLS via the nonconvex sparsity prompting penalties of outlier components," in *Proc. IEEE ChinaSIP*, Jul. 2015, pp. 997–1001.
[14] N. Gallagher and G. Wise, "A theoretical analysis of the properties of median filters," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 29, no. 6, pp. 1136–1141, Dec. 1981.
[15] R. Settineri, M. Najim, and D. Ottaviani, "Order statistic fast Kalman filter," in *Proc. IEEE Int. Symp. Circuits Syst. Circuits Syst. Connecting World*, vol. 2, May 1996, pp. 116–119.
[16] K. Slavakis, "The stochastic Fejér-monotone hybrid steepest descent method and the hierarchical RLS," *IEEE Trans. Signal Process.*, vol. 67, no. 11, pp. 2868–2883, Jun. 2019.
[17] D. Williams, *Probability with Martingales*. New York, NY, USA: Cambridge Univ. Press, 1991.
[18] A. Bereyhi, S. Asaad, R. R. Müller, and S. Chatzinotas, "RLS precoding for massive MIMO systems with nonlinear front-end," in *Proc. IEEE 20th Int. Workshop Signal Process. Advances Wireless Commun.*, Cannes, France, Jul. 2019, pp. 1–5.
[19] J. Benesty, T. Gänsler, Y. Huang, and M. Rupp, *Adaptive Algorithms for MIMO Acoustic Echo Cancellation*. Boston: Springer, 2004, pp. 119–147.
[20] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed. New York: Springer, 2009.
[21] P. Gong, C. Zhang, Z. Lu, J. Z. Huang, and J. Ye, "A general iterative shrinkage and thresholding algorithm for non-convex regularized optimization problems," in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. II–37–II–45.
[22] R. Glowinski and A. Marrocco, "Sur l'approximation par éléments finis et la résolution par pénalisation-dualité d'une classe de problèmes de Dirichlet non linéaires," *Rev. Francaise d'Aut. Inf. Rech. Oper.*, vol. 9, no. 2, pp. 41–76, 1975.
[23] D. Gabay and B. Mercier, "A dual algorithm for the solution of nonlinear variational problems via finite-element approximations," *Comp. Math. Appl.*, vol. 2, pp. 17–40, 1976.
[24] K. Slavakis and I. Yamada, "Fejér-monotone hybrid steepest descent method for affinely constrained and composite convex minimization tasks," *Optimization*, vol. 67, no. 11, pp. 1963–2001, 2018.
[25] H. H. Bauschke and P. L. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. New York: Springer, 2011.
[26] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed. Baltimore: MD: USA: Johns Hopkins Univ. Press, 1996.
[27] K. Petersen, *Ergodic Theory*. Cambridge, U.K.: Cambridge Univ. Press, 1983.
[28] D. W. K. Andrews, "Laws of large numbers for dependent non-identically distributed random variables," *Econom. Theory*, vol. 4, no. 3, pp. 458–467, 1988.
[29] H. J. Kushner and G. G. Yin, *Stochastic Approximation and Recursive Algorithms and Applications*, 2nd ed. New York: Springer-Verlag, 2003.
[30] B. Porat, *Digital Processing of Random Signals: Theory and Methods*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1994.
[31] J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah, "Julia: A fresh approach to numerical computing," *SIAM Rev.*, vol. 59, no. 1, pp. 65–98, 2017.
[32] I. Dunning, J. Huchette, and M. Lubin, "JuMP: A modeling language for mathematical optimization," *SIAM Rev.*, vol. 59, no. 2, pp. 295–320, 2017.
[33] Gurobi Optimization LLC, "Gurobi optimizer reference manual," 2019. [Online]. Available: http://www.gurobi.com