2020 59th IEEE Conference on Decision and Control (CDC)
Jeju Island, Republic of Korea, December 14-18, 2020

On Reducing the Coherence in Sparse System Identification

Javad Parsa*, Håkan Hjalmarsson*

Abstract— **One of the major contributions to sparse learning has been to quantify how the correlation between the regressors affect the ability to recover a sparse parameter vector. Roughly, the inverse of the maximum correlation controls how many non-zero parameters can be exactly recovered from an under-determined system of equations. This result is of importance also in system identification where observations are noisy. Unfortunately, for such problems the regressors are highly correlated making sparse identification difficult. In this contribution we address this problem by applying a linear transformation to the regressors, selected in such a way that the correlation is reduced. The latter is achieved by formulating a constrained optimization problem which is solved using a proximal method. The method can be seen as a pre-processing step that can be used prior to any sparse estimation algorithm. Simulations are used to demonstrate the usefulness of the method.**

I. Introduction

Allowing many free parameters in system identification leads to a large variance error. This problem can be mitigated using regularization or by model structure selection [1], [2]. While a range of methods have been developed to handle the latter problem, e.g. information based criteria such as AIC and BIC, and statistical hypothesis tests [1], model structure selection remains one of the core problems in system identification, especially for a non-experienced user. Current methods are well adapted to model order selection where the selection is made among a set of nested model structures. However, when there is no obvious ordering of the model structures the number of alternatives that need to be examined becomes combinatorial in the maximum number of parameters, see, e.g., [3], [4]. Consider for example a static multinomial model with m inputs u_k, $k = 1, \dots, m$

$$y(t) = \left(\begin{bmatrix} 1 \\ u_1(t) \\ u_1^2(t) \\ \vdots \\ u_1^p(t) \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} 1 \\ u_m(t) \\ u_m^2(t) \\ \vdots \\ u_m^p(t) \end{bmatrix} \right)^T \theta \tag{1}$$

Even if the redundancies in the above model are removed, the dimension of θ is of the order $\mathcal{O}(p^m)$. Even moderate numbers of p and m such as $p = m = 6$ gives thousands of parameters. Here, each model with certain elements set to zero can be considered as a model structure as since there is no particular ordering of these model structures, all need to be examined and compared to each other, unless some prior information is available. We let the reader do the math for how many different model structures one have to examine in the above example. Clearly, estimating models for all these model structures becomes unfeasible. For problems like this, sparse estimation has become an attractive approach.

In sparse estimation, the model structure selection problem is integrated with the parameter estimation problem. To fix ideas, consider a linear regression model (such as the one in (1))

$$\mathbf{y} = \mathbf{\Phi}\mathbf{w} + \mathbf{e}, \tag{2}$$

where $\mathbf{w} \in \mathbb{R}^{n\times 1}$, $\mathbf{\Phi} \in \mathbb{R}^{m\times n}$ and $\mathbf{e} \in \mathbb{R}^{m\times 1}$ denote the vector of unknown parameters, which is assumed to be sparse, the regressor matrix and noise, respectively. In compressed sensing $\mathbf{\Phi}$ and $\mathbf{w}$ are known as a dictionary and sparse representation, respectively. To estimate a sparse $\mathbf{w}$ in (2), the following minimization is used [5]–[7]:

$$\begin{aligned} \mathbf{w}^* = &\underset{\mathbf{w}}{\operatorname{argmin}} \|\mathbf{y} - \mathbf{\Phi}\mathbf{w}\|_2 \\ &\text{subject to} \quad \|\mathbf{w}\|_0 \le s \end{aligned} \tag{3}$$

in which s denotes the number of non-zero elements (the sparsity level). As already mentioned, solving (3) requires solving a number of least-squares problems that is combinatorial in the dimension of $\mathbf{w}$ since each possible split of the elements of $\mathbf{w}$ into at most s non-zero coefficients has to be tested. In [8], Tibshirani proposed to replace the l_0 pseudo-norm by the l_1 norm, leading to the convex relaxation

$$\begin{aligned} \mathbf{w}^* = &\underset{\mathbf{w}}{\operatorname{argmin}} \|\mathbf{y} - \mathbf{\Phi}\mathbf{w}\|_2 \\ &\text{subject to} \quad \|\mathbf{w}\|_1 \le \beta \end{aligned} \tag{4}$$

where β has to be tuned such that the desired sparsity level is achieved. By now there is a very large body of contributions to this type of algorithm in a range of disciplines, e.g. [5], [9]–[16]. The leading to compressed sensing [17]. Also in system identification sparsity has been explored. Sparse regularization is considered in [18], while structure identification of networked systems is considered in [19]. Sparse rational models are considered in [20]. However, the focus has been on the regular case with the number of measurements m exceeding the number of parameters n, and the regressor matrix $\mathbf{\Phi}$ being full rank. Here we will be interested in the challenging situation that $n > m$, but where $\mathbf{w}$ is sparse having sparsity parameter s, i.e. s non-zero elements.

From a sparsity point of view, an important property of a regressor matrix is the maximum correlation between the columns [21]. This quantity is called mutual coherence and

Javad Parsa, and Håkan Hjalmarsson are with the Division of Decision and Control Systems, School of Electrical Engineering and Computer Science, KTH Royal Institute of Technology, 100 44 Stockholm, Sweden javadp,hjalmars@kth.se

Authorized licensed use limited to: KTH Royal Institute of Technology. Downloaded on April 19,2021 at 10:28:24 UTC from IEEE Xplore. Restrictions apply.

is defined as

$$\mu_{\mathbf{\Phi}} = \max_{i \neq j} \frac{|\phi_i^T \phi_j|}{\|\phi_i\|_2 \|\phi_j\|_2} \tag{5}$$

The importance of the mutual coherence is that a signal with a sparse representation $\mathbf{w}$ with sparsity parameter s can be recovered from $\mathbf{y} = \mathbf{\Phi w}$ through l_1 minimization when [22]

$$s \leq \frac{1}{2}\left(1 + \frac{1}{\mu_{\mathbf{\Phi}}}\right) \tag{6}$$

According to (6), regressors with low mutual coherence are required for high s. It is therefore desired to have regressors with low mutual coherence to accurately estimate the sparse coefficients.

In system identification, since $\mathbf{y}$ is constructed from time series data, usually some columns of the regressors have high correlation. To better understand, this consider an AR model:

$$y(t) = -\sum_{k=1}^{n} w_k y(t-k) + e(t),\ t = n+1, \ldots, m+n \tag{7}$$

which can be written in the form (2) with

$$\mathbf{\Phi} = -\begin{bmatrix} y(n) & y(n-1) & \cdots & y(1) \\ y(n+1) & y(n) & \cdots & y(2) \\ \vdots & \vdots & \ddots & \vdots \\ y(m+n-1) & y(N-2) & \cdots & y(m) \end{bmatrix}$$

The columns of the regressor matrix thus consist of shifted versions of the output resulting in a large mutual coherence (unless all model parameters are zero). For this reason, in sparse system identification, methods that use l_1-regularized minimization, have difficulties estimating a sparse parameter vector [23]. Here, we will address this by modifying the regressor matrix so that the mutual coherence is reduced.

The paper is organized as follows. At first, in next section we propose a new method to solve the before problem of sparse system identification that was explained. Our contribution in this paper is that by using proximal mapping method, a new algorithms will be introduced to solve our new method in the subsection II-A. Also, our proposed method has same form with the function of [24] and according to this similarity, in the subsection II-B will be shown that proposed method is converged to the critical point. To better evaluation of proposed method, numbers of simulation are provided and our approach has compared with other methods.

II. The Proposed Algorithm

As mentioned before, mutual coherence of the regressor matrix is one of the important issues in the sparse coding problem. In the sparse system identification problem, the regressor matrix has high mutual coherence since, as we have seen in the AR-case, the regressors consist of shifted versions of the input sequence and for this reason it is not possible to directly use the sparse coding algorithms such as OMP [25], SPGL1 [26], SL0 [6], ADMM [10], etc to recover the true parameters. In this section, we propose a new method to decrease the coherence between the columns of the regressor so that, subsequently, the sparse coding algorithm can be applied.

The overall form of linear regression in sparse system identification was shown in (2). Our idea is to replace the data $\mathbf{y}$ by $\mathbf{My}$ where $\mathbf{M}$ is a user designed square matrix. From (2) it follows that:

$$\mathbf{My} = \mathbf{M\Phi w} + \mathbf{Me}. \tag{8}$$

The idea is now to design $\mathbf{M}$ such that the mutual coherence of the new regressor ($\mathbf{M\Phi}$) is low. An undesired effect of $\mathbf{M}$ is that it also changes the additive noise from $\mathbf{e}$ to $\mathbf{e_M} = \mathbf{Me}$. This may deteriorate the statistical properties. If the noise $\mathbf{e}$ is Gaussian white noise $\mathbf{e} \sim \mathcal{N}(0, \Sigma)$, we obtain

$$\mathbf{e_M} \sim \mathcal{N}(0, \Sigma_{\mathbf{M}}), \quad \Sigma_{\mathbf{M}} = \mathbf{M}\Sigma\mathbf{M}^T \tag{9}$$

In this setting it is statistically efficient to use the maximum likelihood estimation for (2) to estimate $\mathbf{w}$ which in this case corresponds to minimization of the weighted least-squares problem

$$\underset{\mathbf{w}}{\operatorname{argmin}} \left[\frac{(\mathbf{y} - \mathbf{\Phi w})^T \Sigma^{-1} (\mathbf{y} - \mathbf{\Phi w})}{2}\right] \tag{10}$$

This criterion remains invariant under square and invertible matrix multiplication (with M), i.e. (10) is also the maximum likelihood estimation for the model (8). It will thus be necessary to balance the objectives of low coherence of $\mathbf{M\Phi}$ and estimation accuracy.

As mentioned, the matrix $\mathbf{M}$ may have an undesired effect on the estimation error. One of the undesired effects of matrix $\mathbf{M}$ is that the estimation error increases with condition number of $\mathbf{M}$. The matrix $\mathbf{M}$ also changes the variance of the noise, e.g.

$$\mathbb{E}(\mathbf{e_M^T e_M}) = \mathbb{E}\left((\mathbf{Me})^T(\mathbf{Me})\right) = \mathbb{E}(\mathrm{Tr}((\mathbf{Me})^T(\mathbf{Me}))) = \mathrm{Tr}\big((\mathbf{M}^T\mathbf{M})\,\mathbb{E}(\mathbf{ee}^T)\big) = \mathrm{Tr}\big(\mathbf{M}^T\mathbf{M}\Sigma\big).$$

Assuming that $\Sigma = \sigma^2\mathbf{I}$ results in

$$\mathbb{E}(\mathbf{e_M^T e_M}) = \|\mathbf{M}\|_F^2 \sigma. \tag{11}$$

According to (9) and (11), if $\mathbf{M}$ is an orthonormal matrix, this matrix does not have any undesired effect on the noise but unfortunately such a matrix cannot decrease the coherence of the regressor matrix. Thus, we can encode the trade-off between decreasing coherence and maintaining the original statistical properties in terms of decreasing coherence and keeping $\mathbf{M}$ close to orthonormal.

In general there is a trade-off between decreasing the coherence of the regressor and the statistical properties of the estimate.

A. Proposed method to obtain $\mathbf{M}$

In this section, we propose a new method to update $\mathbf{M}$ such that the mutual coherence of the new regressor matrix $\mathbf{M\Phi}$ is low and this new regressor matrix is suited to directly use sparse coding algorithms such as OMP [25]. Our algorithm uses a proximal method and for this reason, at first we define the proximal mapping.

Authorized licensed use limited to: KTH Royal Institute of Technology. Downloaded on April 19,2021 at 10:28:24 UTC from IEEE Xplore. Restrictions apply.

Definition 1: [27] The proximal mapping of a proper and lower semicontinuous function $g : \text{dom}_g \to (-\infty, +\infty]$ at $\mathbf{x} \in \mathbb{R}^n$ is defined as

$$\text{prox}_g(\mathbf{x}) = \underset{\mathbf{u}\in\text{dom}_g}{\text{argmin}}\{\frac{1}{2}\|\mathbf{x}-\mathbf{u}\|_2^2 + g(\mathbf{u})\} \tag{12}$$

An orthonormal matrix has 0 mutual correlation. Therefore we encode our desire of low mutual coherence in the cost function $\|(\mathbf{M\Phi})^T(\mathbf{M\Phi}) - \mathbf{I}_n\|_F^2$, where $\mathbf{I}_n$ is the $n \times n$ identity matrix. Also, according to (11), the variance of the noise varies directly with $\|\mathbf{M}\|_F^2$ and we add the constraint $\|\mathbf{M}\|_F^2 \leq m$ to restrict the variance of the noise. This means that $\mathbf{M}$ should belong to

$$\mathcal{M} = \{\mathbf{M} : \|\mathbf{M}\|_F \leq m\}, \tag{13}$$

in which, m denotes the first dimension of $\mathbf{\Phi}$ ($\mathbf{\Phi} \in \mathbb{R}^{m\times n}$). However, as noted above the more $\mathbf{M}$ deviates itself from an orthonormal matrix, the worse the statistical properties will be. To balance these objectives, we add the constraint [28]–[30]

$$\|\mathbf{M}^T\mathbf{M} - \mathbf{I}_m\|_F^2 \leq \alpha,$$

where α is a design variable, to the problem.

$$\begin{aligned} \mathbf{M}^* &= \underset{\mathbf{M}\in\mathcal{M}}{\text{argmin}}\ \|(\mathbf{M\Phi})^T(\mathbf{M\Phi}) - \mathbf{I}_n\|_F^2 \\ &\text{subject to} \quad \|\mathbf{M}^T\mathbf{M} - \mathbf{I}_m\|_F^2 \leq \alpha \end{aligned} \tag{14}$$

One of the challenges of the above problem is to find $\mathbf{M}$ such that, the new regressor matrix $\mathbf{M\Phi}$ is nearest to the orthonormal matrix that fulfils our constraints i.e. the l_2 norm of each column of the new regressor be approximately one and the mutual coherence of this regressor is low. For this reason, we rewrite the above problem as:

$$\begin{aligned} (\mathbf{A}^*, \mathbf{M}^*) &= \underset{\mathbf{A}\in\mathcal{A},\mathbf{M}\in\mathcal{M}}{\text{argmin}}\ \|\mathbf{A}^T\mathbf{A} - \mathbf{I}_n\|_F^2 \\ &\text{subject to} \quad \mathbf{A} = \mathbf{M\Phi}, \quad \|\mathbf{M}^T\mathbf{M} - \mathbf{I}_m\|_F^2 \leq \alpha \end{aligned}$$

in which, $\mathcal{A}$ is defined as $\mathcal{A} = \left\{\mathbf{A} : \forall i, \|\mathbf{a}_i\|_2^2 = 1\right\}$, respectively. We can eliminate $\mathcal{A}$ and $\mathcal{M}$ by introducing $r_a(\mathbf{A})$ and $r_m(\mathbf{M})$

$$\begin{aligned} r_a(\mathbf{A}) &= \begin{cases} 0 & \text{if } \|\mathbf{a}_i\|_2 = 1 \\ \infty & \text{otherwise} \end{cases} \\ r_m(\mathbf{M}) &= \begin{cases} 0 & \text{if } \|\mathbf{M}\|_F \leq m \\ \infty & \text{otherwise} \end{cases} \end{aligned} \tag{15}$$

giving

$$\begin{aligned} (\mathbf{A}^*, \mathbf{M}^*) &= \underset{\mathbf{A},\mathbf{M}}{\text{argmin}}\ \|\mathbf{A}^T\mathbf{A} - \mathbf{I}_n\|_F^2 + r_a(\mathbf{A}) + r_m(\mathbf{M}) \\ &\text{subject to} \quad \mathbf{A} = \mathbf{M\Phi}, \quad \|\mathbf{M}^T\mathbf{M} - \mathbf{I}_m\|_F^2 \leq \alpha \end{aligned}$$

By applying the Lagrangian method, we can write:

$$(\mathbf{A}^*, \mathbf{M}^*) = \underset{\mathbf{A},\mathbf{M}}{\text{argmin}}\ F(\mathbf{A}, \mathbf{M}) + r_a(\mathbf{A}) + r_m(\mathbf{M})$$

in which

$$\begin{aligned} F(\mathbf{A}, \mathbf{M}) &= \|\mathbf{A}^T\mathbf{A} - \mathbf{I}_n\|_F^2 + \frac{\lambda}{2}\|\mathbf{A} - \mathbf{M\Phi}\|_F^2 \\ &+ \frac{\lambda'}{2}\|\mathbf{M}^T\mathbf{M} - \mathbf{I}_m\|_F^2. \end{aligned} \tag{16}$$

To use proximal mapping, instead of using the above cost function, we propose to update $\mathbf{A}$ and $\mathbf{M}$ by replacing $F(\mathbf{A}, \mathbf{M})$ with its quadratic approximation [24]. This gives the iterations

1) Update $\mathbf{A}$:

$$\begin{aligned} \mathbf{A}_{k+1} = \underset{\mathbf{A}}{\text{argmin}} &< \nabla_{\mathbf{A}} F(\mathbf{A}_k, \mathbf{M}_k), \mathbf{A} - \mathbf{A}_k > \\ &+ \frac{1}{2\mu_A}\|\mathbf{A} - \mathbf{A}_k\|_F^2 + r_a(\mathbf{A}), \end{aligned} \tag{17}$$

in which, $\mu_{\mathbf{A}}$ is a step size.

Problem (17) is equivalent to the following cost function:

$$\mathbf{A}_{k+1} = \underset{\mathbf{A}}{\text{argmin}}\ \frac{1}{2}\|\mathbf{A} - \tilde{\mathbf{A}}_k\|_F^2 + \mu_{\mathbf{A}} r_a(\mathbf{A}) \tag{18}$$

in which, $\tilde{\mathbf{A}}_k = \mathbf{A}_k - \mu_{\mathbf{A}}\nabla_{\mathbf{A}} F(\mathbf{A}_k, \mathbf{M}_k)$. Using the proximal mapping definition (12), we can write (18) as

$$\mathbf{A}_{k+1} = \text{prox}_{\mu_{\mathbf{A}} r_a}(\mathbf{A}_k - \mu_{\mathbf{A}}\nabla_{\mathbf{A}} F(\mathbf{A}_k, \mathbf{M}_k)) \tag{19}$$

According to the above update equation for $\mathbf{A}$, at the first gradient descent method with step size $\mu_{\mathbf{A}}$ is used i.e. $\mathbf{A}_k - \mu_{\mathbf{A}}\nabla_{\mathbf{A}} F(\mathbf{A}_k, \mathbf{M}_k)$ and then the result is projected so that the columns have unit norm, i.e. $\text{prox}_{\mu_{\mathbf{A}} r_a}(\tilde{\mathbf{A}}_k)$.

2) Update $\mathbf{M}$:

$$\begin{aligned} \mathbf{M}_{k+1} = \underset{\mathbf{M}}{\text{argmin}} &< \nabla_{\mathbf{M}} F(\mathbf{A}_{k+1}, \mathbf{M}_k), \mathbf{M} - \mathbf{M}_k > \\ &+ \frac{1}{2\mu_{\mathbf{M}}}\|\mathbf{M} - \mathbf{M}_k\|_F^2 + r_m(\mathbf{M}) \end{aligned} \tag{20}$$

in which, $\mu_{\mathbf{M}}$ is a constant and step size.

The above problem (20) is equivalent to the following cost function:

$$\mathbf{M}_{k+1} = \underset{\mathbf{M}}{\text{argmin}}\ \frac{1}{2}\|\mathbf{M} - \tilde{\mathbf{M}}_k\|_F^2 + \mu_{\mathbf{M}} r_m(\mathbf{M}) \tag{21}$$

in which, $\tilde{\mathbf{M}}_k = \mathbf{M}_k - \mu_{\mathbf{M}}\nabla_{\mathbf{M}} F(\mathbf{A}_{k+1}, \mathbf{M}_k)$. As for $\mathbf{A}$, this results in

$$\mathbf{M}_{k+1} = \text{prox}_{\mu_{\mathbf{M}} r_m}(\mathbf{M}_k - \mu_{\mathbf{M}}\nabla_{\mathbf{M}} F(\mathbf{A}_{k+1}, \mathbf{M}_k)) \tag{22}$$

again corresponding to a gradient step followed by a projection.

Summarizing, $\mathbf{A}$ and $\mathbf{M}$ are alternatively updated by the following two steps

$$\begin{cases} \mathbf{A}_{k+1} = \text{prox}_{\mu_{\mathbf{A}} r_a}(\tilde{\mathbf{A}}_k) \\ \mathbf{M}_{k+1} = \text{prox}_{\mu_{\mathbf{M}} r_m}(\tilde{\mathbf{M}}_k) \end{cases} \tag{23}$$

Our algorithm is summarized in the pseudo code in Algorithm 1. Here, OMP denotes a sparse coding algorithm that can be found in [25].

Authorized licensed use limited to: KTH Royal Institute of Technology. Downloaded on April 19,2021 at 10:28:24 UTC from IEEE Xplore. Restrictions apply.

Algorithm 1 The proposed algorithm (RECOS-OMP)

Input:$\mathbf{y}, \mathbf{\Phi}, s$ (sparsity parameter), λ, λ', $\mu_{\mathbf{A}}$, $\mu_{\mathbf{M}}$
for $k = 1$ to MaxIteration **do**
$\mathbf{A}_{k+1} = \text{prox}_{\mu_{\mathbf{A}} r_a}(\mathbf{A}_k - \mu_{\mathbf{A}} \nabla_{\mathbf{A}} F(\mathbf{A}_k, \mathbf{M}_k))$
$\mathbf{M}_{k+1} = \text{prox}_{\mu_{\mathbf{M}} r_m}(\mathbf{M}_k - \mu_{\mathbf{M}} \nabla_{\mathbf{M}} F(\mathbf{A}_{k+1}, \mathbf{M}_k))$
end for
Sparse approximation: $\mathbf{w}$ =OMP($\mathbf{My}$, $\mathbf{M}\Phi$, s).

B. Convergence analysis

In this section we will establish that the iterations (23) converge. To this end let us define

$$R(\mathbf{A}, \mathbf{M}) = F(\mathbf{A}, \mathbf{M}) + r_m(\mathbf{M}) + r_a(\mathbf{A}) \tag{24}$$

Also, we suppose that $\mathcal{L}_{\mathbf{A}}$ and $\mathcal{L}_{\mathbf{M}}$ are Lipschitz constants of $\nabla_{\mathbf{A}} F(\mathbf{A}, \mathbf{M})$ and $\nabla_{\mathbf{M}} F(\mathbf{A}, \mathbf{M})$, respectively.

Definition 2: [31] The subdifferential of a proper, lower semi- continuous function g at $\mathbf{x} \in \mathbb{R}^n$ is defined as

$$\partial g(\mathbf{x}) \triangleq \{\zeta \in \mathbb{R}^n | \exists \mathbf{x}_k \to \mathbf{x}, g(\mathbf{x}_k) \to g(\mathbf{x}), \zeta_k \to \zeta \in \hat{\partial} g(\mathbf{x}_k)\}$$

in which, $\hat{\partial} g(\mathbf{x}_k)$ is the Frèchet subdifferential of g at $\mathbf{x} \in \mathbb{R}^n$ defined as

$$\hat{\partial} g(\mathbf{x}_k) \triangleq \Bigg\{ \zeta \in \mathbb{R}^n | \liminf_{\mathbf{v} \neq \mathbf{x}, \mathbf{v} \to \mathbf{x}} \frac{1}{\|\mathbf{v} - \mathbf{x}\|_2^2} \Big(g(\mathbf{v}) - g(\mathbf{x}) - < \mathbf{v} - \mathbf{x}, \zeta > \Big) \geq 0 \Bigg\}$$

It is said that $\mathbf{x}^*$ is a critical point of a proper, lower semi-continuous (PLSC) function f if $0 \in \partial f(\mathbf{x}^*)$ [31].

The form of $R(\mathbf{A}, \mathbf{M})$, defined in (24), consisting of non-convex and non-smooth functions belongs to the class of the functions treated in [24]. We have the following theorem:

Theorem 1: Let $\{\mathbf{A}_k, \mathbf{M}_k\}$ be the sequence generated by (23). Assume further that $\mu_{\mathbf{A}} \in (0, 1/\mathcal{L}_{\mathbf{A}}]$ and $\mu_{\mathbf{M}} \in (0, 1/\mathcal{L}_{\mathbf{M}}]$. Then, any accumulation point of $\{\mathbf{A}_k, \mathbf{M}_k\}$ converges to a critical point of $R(\mathbf{A}, \mathbf{M})$ defined in (24).

In [24], Theorem 1 has been proved for any non smooth and non convex cost function. Theorem 1 results in the sequence $\{\mathbf{A}_k, \mathbf{M}_k\}_{k=0}^{k=\infty}$ globally converges to $\{\mathbf{A}^*, \mathbf{M}^*\}$ in which $\{\mathbf{A}^*, \mathbf{M}^*\}$ is a critical point of $R(\mathbf{A}, \mathbf{M})$, i.e. $0 \in \partial R(\mathbf{A}^*, \mathbf{M}^*)$. However, for this result to hold we need to prove that the sequence $\{\mathbf{A}_k, \mathbf{M}_k\}$ is bounded. According to (23), $\mathbf{A}_{k+1} = \text{prox}_{\mu_{\mathbf{A}} r_a}(\tilde{\mathbf{A}}_k)$ and in each iteration, it is projected to the set defined by that $r_a(\mathbf{A}) = 0$. This projection causes that $\{\mathbf{A}_k\}_{k=0}^{\infty}$ is bounded. The boundedness of $\{\mathbf{M}_k\}_{k=0}^{\infty}$ follows in the same way. Our method thus satisfies all conditions of [24] and, hence, Theorem 1 follows.

III. Choice of step-sizes

Theorem 1 requires $\mu_{\mathbf{A}} \in (0, 1/\mathcal{L}_{\mathbf{A}}]$ and $\mu_{\mathbf{M}} \in (0, 1/\mathcal{L}_{\mathbf{M}}]$. In this section, we will derive upper bounds for $\mathcal{L}_{\mathbf{A}}$ and $\mathcal{L}_{\mathbf{M}}$.

A Lipschitz constant is defined by that

$$\|\nabla_{\mathbf{X}} H(\mathbf{X}_2) - \nabla_{\mathbf{X}} H(\mathbf{X}_1)\|_F \leq \mathcal{L}_H \|\mathbf{X}_2 - \mathbf{X}_1\|_F. \tag{25}$$

According to (25), we have

$$\begin{aligned}
&\nabla_{\mathbf{A}} F(\mathbf{A}, \mathbf{M}) = 2\mathbf{A}(\mathbf{A}^T\mathbf{A} - \mathbf{I}_n) + \lambda(\mathbf{A} - \mathbf{M}\mathbf{\Phi}) \\
&\nabla_{\mathbf{A}} F(\mathbf{A}_2, \mathbf{M}) - \nabla_{\mathbf{A}} F(\mathbf{A}_1, \mathbf{M}) = \\
&2(\mathbf{A}_2\mathbf{A}_2^T\mathbf{A}_2 - \mathbf{A}_1\mathbf{A}_1^T\mathbf{A}_1) - 2(\mathbf{A}_2 - \mathbf{A}_1) + \lambda(\mathbf{A}_2 - \mathbf{A}_1) \\
&= 2((\mathbf{A}_2 - \mathbf{A}_1)\mathbf{A}_2^T\mathbf{A}_2 + \mathbf{A}_1(\mathbf{A}_2 - \mathbf{A}_1)^T\mathbf{A}_2 \\
&+ \mathbf{A}_1\mathbf{A}_1^T(\mathbf{A}_2 - \mathbf{A}_1)) + (\lambda - 2)(\mathbf{A}_2 - \mathbf{A}_1).
\end{aligned}$$

Then, utilizing the inequalities $\|\mathbf{A} + \mathbf{B}\|_F \leq \|\mathbf{A}\|_F + \|\mathbf{B}\|_F$ and $\|\mathbf{AB}\|_F \leq \|\mathbf{A}\|_F \|\mathbf{B}\|_F$

$$\begin{aligned}
&\|\nabla_{\mathbf{A}} F(\mathbf{A}_2, \mathbf{X}) - \nabla_{\mathbf{A}} F(\mathbf{A}_1, \mathbf{X})\|_F \leq (2\|\mathbf{A}_2\|_F^2 + \\
&2\|\mathbf{A}_1\|_F\|\mathbf{A}_2\|_F + 2\|\mathbf{A}_1\|_F^2 + (2 - \lambda))\|\mathbf{A}_2 - \mathbf{A}_1\|_F.
\end{aligned} \tag{26}$$

Assuming normalized columns of $\mathbf{A}$ and $\mathbf{\Phi}$, we have:

$$\|\mathbf{\Phi}\|_F = \|\mathbf{A}_1\|_F = \|\mathbf{A}_2\|_F = \sqrt{n}, \mathbf{A}, \mathbf{\Phi} \in \mathbb{R}^{m \times n}. \tag{27}$$

Use (26) to derive this and conclude that

$$\mathcal{L}_{\mathbf{A}} \leq 6n + (2 - \lambda). \tag{28}$$

Similarly, we can find the Lipschitz constant $\mathcal{L}_{\mathbf{M}}$ as follows:

$$\begin{aligned}
&\nabla_{\mathbf{M}} F(\mathbf{A}, \mathbf{M}) = \lambda'\mathbf{M}(\mathbf{M}^T\mathbf{M} - \mathbf{I}_m) + \lambda(-\mathbf{A} + \mathbf{M}\mathbf{\Phi})\mathbf{\Phi}^T \\
&\nabla_{\mathbf{M}} F(\mathbf{A}, \mathbf{M}_2) - \nabla_{\mathbf{M}} F(\mathbf{A}, \mathbf{M}_1) \\
&= \lambda'((\mathbf{M}_2 - \mathbf{M}_1)\mathbf{M}_2^T\mathbf{M}_2 + \mathbf{M}_1(\mathbf{M}_2 - \mathbf{M}_1)^T\mathbf{M}_2 \\
&+ \mathbf{M}_1\mathbf{M}_1^T(\mathbf{M}_2 - \mathbf{M}_1)) + \lambda(\mathbf{M}_2 - \mathbf{M}_1)\mathbf{\Phi}\mathbf{\Phi}^T,
\end{aligned}$$

results in

$$\begin{aligned}
&\|\nabla_{\mathbf{M}} F(\mathbf{A}, \mathbf{M}_2) - \nabla_{\mathbf{M}} F(\mathbf{A}, \mathbf{M}_1)\|_F \leq (\lambda'\|\mathbf{M}_2\|_F^2 + \\
&\lambda'\|\mathbf{M}_1\|_F\|\mathbf{M}_2\|_F + \lambda'\|\mathbf{M}_1\|_F^2 + \lambda\|\mathbf{\Phi}\|_F^2)\|\mathbf{M}_2 - \mathbf{M}_1\|_F.
\end{aligned} \tag{29}$$

According to $r_m(\mathbf{M})$, $\|\mathbf{M}\|_F \leq m$, we have

$$\mathcal{L}_{\mathbf{M}} \leq 3\lambda' m + \lambda n \tag{30}$$

Now Theorem 1 requires

$$\begin{cases} \mu_{\mathbf{A}} \leq \frac{1}{\mathcal{L}_{\mathbf{A}}} \\ \mu_{\mathbf{M}} \leq \frac{1}{\mathcal{L}_{\mathbf{M}}} \end{cases} \tag{31}$$

but from (28) and (30)

$$\begin{cases} \frac{1}{\mathcal{L}_{\mathbf{A}}} \geq \frac{1}{6n + (2 - \lambda)} \\ \frac{1}{\mathcal{L}_{\mathbf{M}}} \geq \frac{1}{3\lambda' m + \lambda n} \end{cases} \tag{32}$$

so taking

$$\begin{cases} \mu_{\mathbf{A}} \leq \frac{1}{6n + (2 - \lambda)} \\ \mu_{\mathbf{M}} \leq \frac{1}{3\lambda' m + \lambda n} \end{cases} \tag{33}$$

ensures the condition in Theorem 1.

Authorized licensed use limited to: KTH Royal Institute of Technology. Downloaded on April 19,2021 at 10:28:24 UTC from IEEE Xplore. Restrictions apply.

IV. Simulation results

In this section, we experimentally evaluate the performance of our method.

One can see the method as a pre-processing step to be used in conjunction with any sparse coding algorithm. In the simulations it is used in conjunction with the OMP and SPGL1 algorithms and we denote these combinations RECOS-OMP (RECOS-OMP stands REduce COherence Sparse system identification-OMP) and RECOS-SPGL1, respectively. Also, we compare our methods with some common sparse coding algorithms such as OMP, SPGL1 and a Sparse Bayesian Approach to System Identification [32] (it is denoted with SPBSI)[1]. Our simulations were performed in MATLAB R2019b environment on a system with 4.00 GHz I7 CPU and 16 GB RAM, under Microsoft Windows 10 operating system. As a rough measure of complexity, we will mention the run times of the algorithms. We assume that $\mathbf{w}_t$ and $\mathbf{w}_p$ are the true and predicted sparse coefficient, respectively. The performance measures are squared sparsity error (SSE) defined as SSE= $\|\mathbf{w}_t - \mathbf{w}_p\|_2$ and mutual coherence, defined by (5). Another criterion to evaluate is the percentage of recovery of the non-zero coefficient positions which is defined as follows:

$$POD = \frac{\text{nnz}(\mathbf{w}_t \odot \mathbf{w}_p)}{s} \tag{34}$$

in which, nnz and $\odot$ denote the number of non-zero elements and Hadamard product (element-wise product), respectively.

We generated random matrix $\mathbf{\Phi} \in \mathbb{R}^{200\times 2000}$ with zero mean, unit variance and high mutual coherence. Mutual coherence of regressor is '0.99'. Also, we produce a random sparse vector with sparsity level s as $\mathbf{w}_t$ then, this sparse vector is multiplied to the regressor matrix $\mathbf{\Phi}$ such that after adding with noise, it results in $\mathbf{y}$. Our purpose is to estimate the sparse coefficient by using the output $\mathbf{y}$ and $\mathbf{\Phi}$ as $\mathbf{w}_p$. The criteria to evaluate the proposed method in comparison to the other methods are SSE and POD such that these measure the accuracy of the values and positions of the non-zero elements of $\mathbf{w}$, respectively. Also, SNR denotes the signal-to-noise ratio and it is defined as

$$\text{SNR} = 10\log\frac{\|\Phi\mathbf{w}\|_2^2}{\|\mathbf{y}-\Phi\mathbf{w}\|_2^2} \tag{35}$$

In our simulation, three values of 10, 25, 34 have chosen for s. Other algorithms have setting parameters that these are tuned by choosing the best values according to the simulation. The setting parameters of SPBSI are $\lambda_{SPBSI} =$ 0.001, 0.002 and 0.003 for sparsity level= 34, 25 and 10, respectively (MAXITER is equal to 25). Also, the parameters of our algorithms are set as $\lambda = 0.1, \lambda' = 0.0008$ and $\mu_{\mathbf{M}} = \mu_{\mathbf{A}} = 0.001$.

[1]For OMP, we used the available 'Sparse-Plex' MATLAB toolbox at '`https://github.com/indigits/sparse-plex`'.
The MATLAB code of SPGL1 is taken from '`https://github.com/hezhangsprinter/SROSR/tree/master/tools/sparse20coding/spgl1-1.9`'.
SPBSI has implemented by using '`https://github.com/panweihit/BSID/tree/master/VanillaID`'.

According to Fig.1, by applying the proposed idea, the accuracy of sparse estimation increases significantly. For example, the estimation error of OMP and SPGL1 algorithms are 51.15 and 47, receptively while, the error of RECOS-OMP and RECOS-SPGL1 in high SNR are 0.5 and 1, respectively. Also, our algorithms have better performance in comparison to the SPBSI algorithm for instance, SSE of SPBSI is 10.2 while this error for our proposed method (RECOS-OMP) is 0.5. In Fig.2, proposed methods have higher accuracy in the position estimation of non-zero elements especially at high SNR, for example at small noise, POD of RECOS-OMP and RECOS-SPGL1 are 1 while POD of SPBSI and OMP are 0.72 and 0.04, respectively.

In addition, the proposed method in this simulated example decreases the mutual coherence from 0.99 to 0.4.

According to Fig.3 and 4 the proposed pre-processing results in both lower SSE and higher POD in comparison to the other algorithms in $s = 34$, especially for small noise. In Table I, the SSE, POD and average running time in $s = 25$ have been reported and according to this table RECOS-OMP and RECOS-SPGL1 have lower SSE and higher POD than the other methods while RECOS-OMP, RECOS-SPGL1 and SPBSI need more time than the other methods. Also, we can say that RECOS-OMP has better performance than the SPBSI while both methods take almost same time. In total, it is clear that RECOS-OMP has better performance for this simulation example than the other methods. The improvement of RECOS-OMP in SSE and POD are more significant for higher SNR so that, when the SNR is high, the benefits of the pre-processing step are more significant in comparison to when the SNR is low.

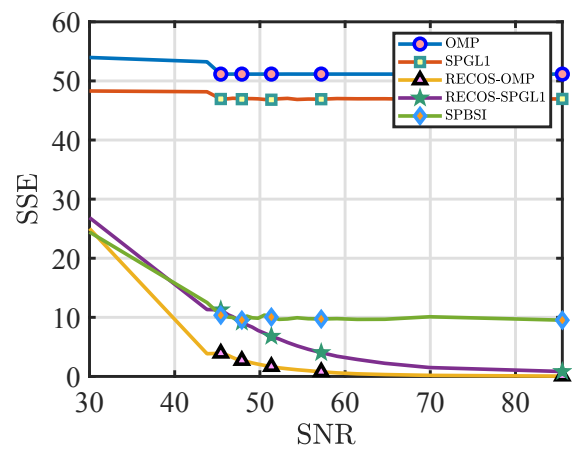

Fig. 1. Evaluation of sparse estimation error with assumptions: $s = 25$.

V. Conclusion

In this paper, we proposed a new approach to improve the mutual coherence in sparse system identification by pre-processing data. The idea is to use a linear transformation on the data, where the transformation is chosen to balance mutual coherence and statistical properties . We have proposed a proximal method for solving the associated optimization

Authorized licensed use limited to: KTH Royal Institute of Technology. Downloaded on April 19,2021 at 10:28:24 UTC from IEEE Xplore. Restrictions apply.

problem, with proven convergence properties. We have illustrated in simulations that this pre-processing can improve both the squared sparsity error and the percentage of recovery of the non-zero coefficient positions.

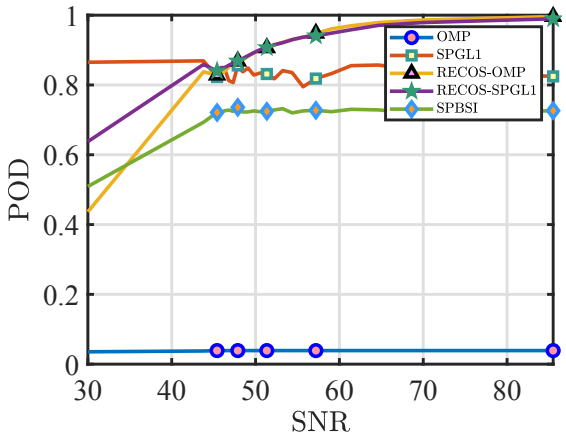

Fig. 2. Evaluation of sparse recovery with assumptions: $s = 25$.

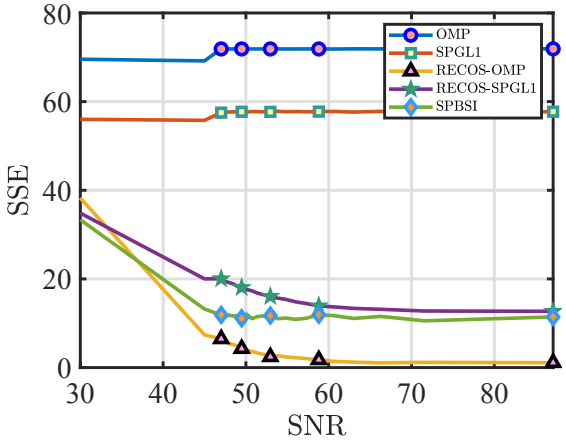

Fig. 3. Evaluation of sparse estimation error with assumptions: $s = 34$.

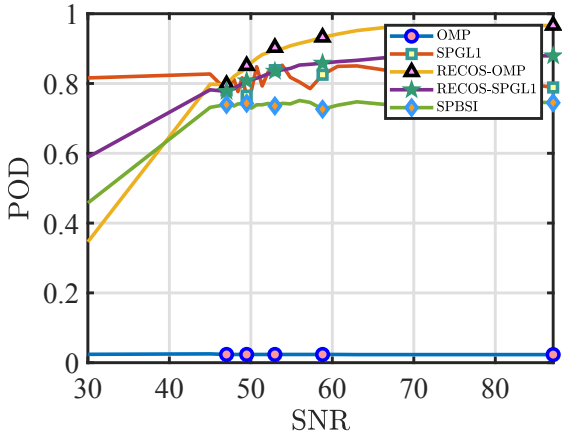

Fig. 4. Evaluation of sparse recovery with assumptions: $s = 34$.

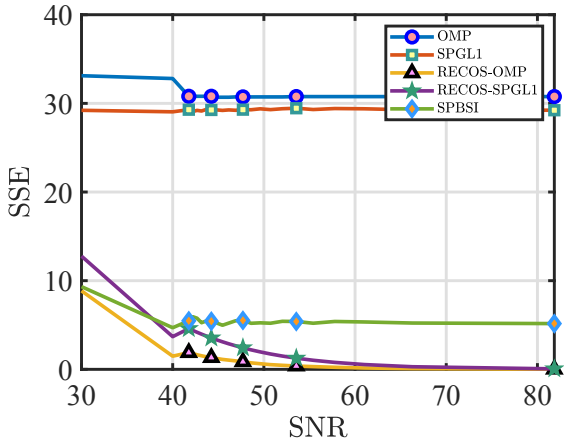

Fig. 5. Evaluation of sparse estimation error with assumptions: $s = 10$.

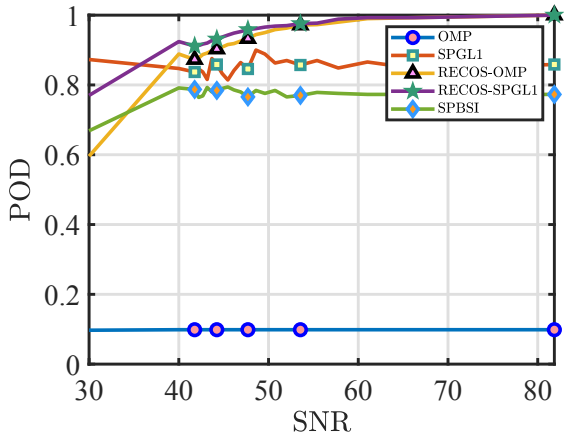

Fig. 6. Evaluation of sparse recovery with assumptions: $s = 10$.

References

[1] L. Ljung, *System Identification: Theory for the User*, 2nd, Ed. Englewood Cliffs, NJ: Prentice-Hal, 1999.

[2] G. Pillonetto, F. Dinuzzo, T. Chen, G. De Nicolao, and L. Ljung, "Kernel methods in system identification, machine learning and function estimation: A survey," *Automatica*, vol. 50, no. 3, pp. 657–682, 2014.

[3] T. Söderström and P. Stoica, *System Identification*. Hertfordshire, U.K.: Prentice-Hall, 1989.

[4] J. Stoddard, J. Welsh, and H. Hjalmarsson, "Em-based hyperparameter optimization for regularized volterra kernel estimation," *IEEE Control-Systems Letters*, vol. 1, no. 2, pp. 388–393, 2017.

[5] M. Elad, *Sparse and Redundant Representations*. Springer, 2010.

[6] H. Mohimani, M. Babaie-Zadeh, and C. Jutten, "A fast approach for overcomplete sparse decomposition based on smoothed ℓ^0 norm," *IEEE Trans. on Signal Processing*, vol. 57, pp. 289–301, 2009.

[7] Y. Eldar and G. Kutyniok, *Compressed sensing: theory and applications*. Cambridge university press, 2012.

[8] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B*, vol. 58, no. 1, pp. 267–288, 1996.

[9] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, 2009.

[10] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[11] I. Daubechies, R. DeVore, M. Fornasier, and C. S. Gntrk, "Iteratively re-weighted least squares minimization for sparse recovery," *Communications on Pure and Applied Mathematics*, vol. 63, no. 1, pp. 1–38, Jan. 2010.

Authorized licensed use limited to: KTH Royal Institute of Technology. Downloaded on April 19,2021 at 10:28:24 UTC from IEEE Xplore. Restrictions apply.

TABLE I

SSE, POD AND AVERAGE RUNNING TIME (IN SECONDS). AVERAGE RUNNING TIMES AND POD ARE REPORTED IN PARENTHESES AND BRACKET (BY PERCENTAGE ALIGNMENT), RESPECTIVELY. IN THIS TABLE $s = 25$ IS SUPPOSED.

Algorithm	SNR = 80	SNR = 70	SNR = 60	SNR = 50	SNR = 40	SNR = 30
RECOS-OMP	0.052 [99.96%] (49.57s)	0.18 [98.63%] (49.51s)	0.6 [96.06%] (49.3s)	1.86 [90%] (49.4s)	3.81 [83.77%] (49.01s)	25 [45.7%] (49.2s)
RECOS-SPGL1	0.83 [98.97%] (49.69s)	1.5 [97.9%] (49.71s)	3.41 [95%] (49.52s)	7.32 [90.11%] (49.59s)	11.32 [85.89%] (49.30s)	26.86 [63.77%] (49.33s)
SPBSI	9.52 [72.63%] (46.42s)	10.1 [71.71%] (47.41s)	9.8 [72.34%] (48.85s)	10.39 [72.34%] (47.47s)	12.6 [68.31%] (49s)	24.5 [51%] (48.8s)
SPGL1	47 [82.3%] (0.363s)	47 [82.3%] (0.524s)	47 [82.3%] (0.54s)	46.86 [82.3%] (0.687s)	48.16 [82.91%] (0.5s)	48.3 [82.51%] (0.8s)
OMP	51.152 [3.9%] (0.0025s)	51.152 [3.9%] (0.0031s)	51.152 [3.9%] (0.0026s)	51.152 [3.9%] (0.0027s)	53.23 [3.77%] (0.0041s)	53.23 [3.54%] (0.0029s)

[12] T. Blumensath and M. E. Davies, "Iterative hard thresholding for compressed sensing," *Applied and Computational Harmonic Analysis*, vol. 27, no. 3, pp. 265–274, 2009.

[13] P. Stoica, D. Zachariah, and J. Li, "Weighted spice: A unifyingapproach for hyperparameter-free sparse estimation," *Digital Signal Processing*, vol. 33, pp. 1–12, 2014.

[14] H. Leeb and B. Ptscher, "Sparse estimators and the oracle property, or the return of hodges estimator," *Journal of Econometrics*, vol. 142, pp. 201–211, 2008.

[15] H. Zhou, "The adaptive lasso and its oracle properties," *Journal of the American Statistical Association*, vol. 101, no. 476, pp. 1418–1429, 2006.

[16] Y. Li, M. Lei, Y. Guo, Z. Hu, and H.-L. Wei, "Time-varying nonlinear causality detection using regularized orthogonal least squares and multi-wavelets with applications to eeg," *IEEE Access*, vol. 6, pp. 17 826–17 840, 2018.

[17] E. Candès, J. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications on pure and applied mathematics*, vol. 59, pp. 1207–1233, 2006.

[18] T. Chen, M. Andersen, L. Ljung, A. Chiuso, and G. Pillonetto, "System identification via sparse multiple kernel-based regularizationusing sequential convex optimization techniques," *IEEE Transactionson Automatic Control*, vol. 59, no. 11, pp. 2933–2945, 2014.

[19] A. Chiuso and G. Pillonetto, "A bayesian approach to sparse dynamicnetwork identification,," *Automatica*, vol. 48, no. 8, pp. 1553–1565, 2012.

[20] C. Rojas, R. Tth, and H. Hjalmarsson, "Sparse estimation of polynomial and rational dynamical models," *IEEE Trans. on Automatic Control*, vol. 59, no. 11, 2014.

[21] D. L. Donoho and X. Huo, "Uncertainty principles and ideal atomic decomposition," *IEEE Trans. Information Theory*, vol. 47, no. 7, pp. 2845–2862, 2001.

[22] D. L. Donoho and M. Elad, "Optimally sparse representation in general (nonorthogonal) dictionaries via ℓ^1 minimization," *Proc. Nat. Aca. Sci*, vol. 100, no. 5, pp. 2197–2202, 2003.

[23] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Info. Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[24] J. Bolte, S. Sabach, and M. Teboulle, "Proximal alternating linearized minimization for nonconvex and nonsmooth problems," *Mathematical Programming*, vol. 146, no. 1-2, pp. 459–494, 2014. [Online]. Available: http://dx.doi.org/10.1007/s10107-013-0701-9

[25] D. L. Donoho, Y. Tsaig, I. Drori, and J. L. Starck, "Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit," *IEEE Trans. on Information Theory*, vol. 58, no. 2, pp. 1094–1121, 2012.

[26] E. van den Berg and M. P. Friedlander, "Probing the pareto frontier for basis pursuit solutions," *SIAM Journal on Scientific Computing*, vol. 31, no. 2, 2008.

[27] N. Parikh and S. Boyd, "Proximal algorithms," *Foundations and Trends in Optimization*, vol. 1, no. 3, pp. 123–231, 2014.

[28] J. Parsa, M. Sadeghi, M. Babaie-Zadeh, and C. Jutten, "Low mutual and average coherence dictionary learning using convex approximation," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2020)*, 2020.

[29] C. D. Sigg, T. Dikk, and J. M. Buhmann, "Learning dictionaries with bounded self-coherence," *IEEE Signal Proc. Letters*, vol. 19, no. 12, pp. 861–864, 2012.

[30] J. Parsa, M. Sadeghi, M. Babaie-Zadeh, and C. Jutten, "Joint low mutual and average coherence dictionary learning," in *Proceeding of 26th European Signal Processing Conference(EUSIPCO 2018)*, 2018.

[31] H. Attouch, J. Bolte, and B. F. Svaiter, "Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forwardbackward splitting, and regularized Gauss–Seidel methods," *Mathematical Programming*, vol. 137, no. 1, pp. 91–129, 2013.

[32] W. Pan, Y. Yuan, J. Gonalves, and G. Stan, "A sparse bayesian approach to the identification of nonlinear state-space systems," *IEEE Trans. on Automatic Control*, vol. 61, no. 1, 2016.

Authorized licensed use limited to: KTH Royal Institute of Technology. Downloaded on April 19,2021 at 10:28:24 UTC from IEEE Xplore. Restrictions apply.