Constrained Sparse Component Analysis with Applications to EEG/MEG Signal Processing

Nasser Mourad[1,3], T. Kirubarajan[1], James P. Reilly[1], Gary Hasey[2] and Duncan MacCrimmon[2]

[1] Department of Electrical and Computer Engineering

McMaster University, Hamilton, Ont. CANADA L8S 4K1.

[2] Department of Psychiatry and Behavioural Neurosciences

McMaster University.

[3] Department of Electrical Engineering

South Valley University, Aswan, EGYPT.

Abstract

In this paper we propose two different algorithms for extracting a desired *sparse* source signal from a linear mixture of hidden source signals in the presence of noise when prior information about the desired source signal is available. The first algorithm extracts a desired source signal using a temporal reference signal, while the second algorithm uses a spatial reference vector that conveys information about the mixing column associated with the desired source signal. Previous methods addressing this problem have used algorithms based on the ℓ_1 norm for encouraging sparsity. However, it has been shown e.g., [20] that certain non-convex functions perform better than ℓ_1 based methods for this purpose, in terms of probability of exact reconstruction and the required number of iterations for convergence. In this paper, we extend the previous work in this area by exploiting recent algorithms e.g., [21] developed by the authors that use these improved non-convex functions. In the temporally constrained case, the desired separating vector is obtained iteratively as the minimum latent vector of a particular matrix. In the spatially constrained case, the separating vector is determined as the solution to a set of linear equations. The proposed algorithms are readily implementable and have useful applications in the field of EEG/MEG signal processing. Numerical results show that these algorithms significantly outperform some of the well known algorithms that are usually employed for solving the same problem particularly in terms of computational speed.

I. INTRODUCTION

The blind source separation (BSS) problem is defined as the problem of reconstructing n *unknown* source signals from m linear measurements when the mixing matrix is *unknown*.

Over the last two decades, the BSS problem has been solved using the independent component analysis technique (ICA). Generally speaking, there are three different approaches for solving the BSS problem via ICA. The first approach is based on estimating all hidden source signals simultaneously. This is usually done by finding a separating matrix $\boldsymbol{B}$ such that the estimated sources $\boldsymbol{Y} = \boldsymbol{BX}$ are mutually independent [1]. This approach has been utilized successfully for removing different kinds of artifacts from electroencephalogram/magnetoencephalogram (EEG/MEG) data [2]–[4]. However, when the number of sensors is large, as in this case, the process of identifying sources of interest from amongst the separated sources is generally not trivial and requires human intervention.

The second approach for solving the BSS problem is based on sequentially extracting the source signals, a technique known in the literature as blind signal extraction (BSE) [5]–[7]. The most popular ICA algorithm that follows this strategy is called FastICA [5]. FastICA is based on finding a separating *row* vector $\bar{\boldsymbol{b}}$ by maximizing the *non Gaussianity* of the separated signal $\bar{\boldsymbol{y}} = \bar{\boldsymbol{b}}\boldsymbol{X}$. Therefore, the algorithm will converge (ideally) to the most non-Gaussian source signal. However, when one desires a specific source, there is no guarantee that the first extracted source signal corresponds to the desired source signal. The constrained ICA (cICA) approach addresses this difficulty [8]–[14] by exploiting any available *a priori* temporal or spatial information about the desired source signal.

The cICA technique which was first proposed in [8], extracts a source signal that is statistically independent of the other sources and is closest to some reference signal $r(t)$ which contains *a priori* temporal information. The technique was then further developed in [9], [10] to handle the case of multiple source signals for simultaneous extraction of multiple independent components. Further development was then suggested in [12], [15], [16] for the case of a spatial constraint. In all these cases, the reference signal must be selected carefully to have enough information to guide the algorithm towards the desired signal. Note that the reference signal is not required to be an exact replica of the desired signal.

There are several forms of available *a priori* information, e.g., in the field of biomedical signal analysis. For example, it is known that ocular artifacts from EEG/MEG data most likely contaminate only the frontal electrodes. Accordingly, a reference signal in this case can be easily derived by thresholding the signal from one of the frontal electrodes, as in [11], [13], [17]. Also, the frequency of the line interference is known; further, a reference signal for electrocardiogram (ECG) contamination can be

obtained by using a co-recording from a separate electrode placed on the chest. In each of these cases, a reference signal can be automatically derived to serve as a constraint in the cICA algorithm [11]. Using prior information eliminates the need for post-processing of the separated components, and reduces the required computational resources and cost.

In this paper we consider the case of utilizing prior information to extract a desired *sparse* source signal, where a sparse signal is defined as one for which most of its samples are zero. The derived algorithms are also applicable to non–sparse source signals that can be sparsely represented under a suitable linear transformation, (e.g., the short time Fourier transform, the wavelet transform, the wavelet packets transform,... etc.). The prior information considered in this paper is in either a temporal or a spatial form, and is represented as a reference vector $\boldsymbol{r}$. Spatial information in EEG/MEG signals is available in the form of the "*topographic map*" of the desired source signal, and represents the mixing column of $\boldsymbol{A}$ associated with the desired source signal. Since the desired source signal is sparse, we propose utilizing the *diversity* (the number of nonzero samples) of the separated signal as an objective function. We have shown in [18] that a sparse source signal can be extracted with high accuracy by minimizing the diversity of the separated signal rather than maximizing its non–Gaussianity, as in FastICA. A previous approach that uses an objective function that explicitly measures the diversity of the extracted source signal was proposed in [19]. A brief revision of some algorithms that have been developed in the literature for solving the same problem, as well as the advantages of the proposed algorithms over these algorithms, are presented in Section II.

The remaining part of this paper is organized as follows. A brief revision of some algorithms which have been previously proposed in the literature for extracting a desired source signal using either temporal or spatial constraints, is presented in Section II. In Section III we propose a new algorithm for extracting the *sparsest* source signal from a linear mixture of some hidden source signals. New algorithms for extracting a sparse source signal using a prior temporal or spatial information about the desired source signal are derived in Section IV and Section V, respectively. Section VI presents some computer simulations for assessing the performance of the proposed algorithms. Finally, conclusion remarks are given in Section VII.

Notation: A bold upper case symbol, e.g., $\boldsymbol{A}$ refers to a matrix; similarly $\boldsymbol{a}$ refers to a column vector, $\bar{\boldsymbol{a}}$ refers to a row vector, $\boldsymbol{a}_i$ refers to the ith column of $\boldsymbol{A}$, $\boldsymbol{a}^i$ refers to the ith row of $\boldsymbol{A}$, and $a[i]$ refers to the ith element of $\boldsymbol{a}$ or $\bar{\boldsymbol{a}}$.

II. Problem formulation and previous approaches

The relation between the measured signals and the original source signals can be expressed mathematically as

$$\boldsymbol{X} = \boldsymbol{A}\boldsymbol{S} + \boldsymbol{W}, \tag{1}$$

where $\boldsymbol{X} \in \mathbb{R}^{m\times T}$ is a matrix of measured signals, $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ is an unknown mixing matrix, $\boldsymbol{S} \in \mathbb{R}^{n\times T}$ is a matrix of unknown source signals, $\boldsymbol{W} \in \mathbb{R}^{m\times T}$ is zero-mean *i.i.d.* noise, assumed uncorrelated with the signal, with variance σ_w^2, m is the number of sensors, n is the number of sources, and T is the number of samples. It is assumed that $m \geq n$ and that $\boldsymbol{A}$ and $\boldsymbol{S}$ are each of rank n.

A single source signal can be extracted from $\boldsymbol{X}$ in (1) by finding a separating row vector $\bar{\boldsymbol{b}} \in \mathbb{R}^n$ such that the separated signal $\bar{\boldsymbol{y}} = \bar{\boldsymbol{b}}\boldsymbol{X}$ by solving

$$\hat{\bar{\boldsymbol{b}}} = \arg\max_{\bar{\boldsymbol{b}}} \mathcal{J}\left(\bar{\boldsymbol{b}}\boldsymbol{X}\right) \quad \text{subject to} \quad ||\bar{\boldsymbol{b}}||_{\ell_2} = 1, \tag{2}$$

where the constraint $||\bar{\boldsymbol{b}}||_{\ell_2} = 1$ is imposed to bound the entries of the solution vector. The objective function $\mathcal{J}\left(\bar{\boldsymbol{y}}\right)$ can take different forms depending on the statistical properties assumed about the desired source signal $\bar{\boldsymbol{y}}$. For instance, in [5] it was assumed that the source signals are independent and at most one source signal has Gaussian distribution, in which case $\mathcal{J}\left(\bar{\boldsymbol{y}}\right)$ measures the non-Gaussianity of the separated signal $\bar{\boldsymbol{y}}$. In [18] it is assumed the desired source signal is sparse, in which case $\mathcal{J}\left(\bar{\boldsymbol{y}}\right)$ measures the sparsity of the separated signal.

If one is looking for a specific source signal, ($\boldsymbol{s}^i$ for instance), then these approaches might not converge to $\boldsymbol{s}^i$ unless $\mathcal{J}\left(\boldsymbol{s}^i\right) > \mathcal{J}\left(\boldsymbol{s}^j\right), \forall i \neq j$. To increase the probability of extracting the desired source signal, prior information may be incorporated into (2). This available prior information is incorporated into the problem through the use of a reference signal $\bar{\boldsymbol{r}} \in \mathbb{R}^T$ that conveys either temporal or spatial information about the desired source signal. Examples of how $\bar{\boldsymbol{r}}$ is obtained in different scenarios are given in Sect. VI.

The least squares method with a temporal reference signal is the simplest means of incorporating *a priori* information, where the separating vector $\bar{\boldsymbol{b}}_{ls} \in \mathbb{R}^n$ is estimated by minimizing the objective function

$$\bar{\boldsymbol{b}}_{ls} = \arg\min_{\bar{\boldsymbol{b}}} \mathcal{J}_{ls}\left(\bar{\boldsymbol{b}}\right) \triangleq ||\bar{\boldsymbol{r}}_t - \bar{\boldsymbol{b}}\boldsymbol{X}||_{\ell_2}^2 \tag{3}$$

for which the corresponding extracted signal is given by

$$\bar{\boldsymbol{y}}_{ls} = \bar{\boldsymbol{b}}_{ls}\boldsymbol{X} = \bar{\boldsymbol{r}}_t\boldsymbol{X}^T\left(\boldsymbol{X}\boldsymbol{X}^T\right)^{-1}\boldsymbol{X}. \tag{4}$$

However, this approach cannot exploit any known statistical properties of the extracted signal, such as sparsity or non-Gaussianity of the extracted signal. In this respect, a temporally constrained FastICA algorithm has been developed [8]–[10] which extracts a non–Gaussian source signal that is statistically independent of the other source signals yet is close to a specified temporal reference signal $\bar{\boldsymbol{r}}_t$. The method is based on maximizing a function which encourages non–Gaussianity, subject to a suitably–defined norm being below a specified threshold value, which must be pre-determined. If this threshold value is not chosen properly, the algorithm will either fail to converge or the resulting signal will be a mixture of two or more sources. Furthermore, the method cannot exploit the potential sparsity of a particular source signal.

An algorithm that exploits both sparsity and a reference signal is presented in [19]. This algorithm is henceforth referred to in this paper as SCA with reference (SCA-R). The method is based on the minimization of a function which is a weighted combination of a term based on an ℓ_1–norm which encourages sparsity of the extracted signal, and a term which encourages agreement with the specified reference signal.

As proposed by Chartrand in [20], and further addressed in [21]–[23], there is a class of non–convex objective functions that perform better than the ℓ_1–norm in encouraging the sparsity of a given solution vector, with respect to the probability of exact reconstruction and the number of iterations required for convergence. (Such functions include the $\ell_q, q < 1$–norms. Other examples are given later in the paper.) In this paper we propose two different algorithms for extracting a desired source signal which is assumed to be sparse that incorporate these improved sparsity measures. The first algorithm, (referred to as *temporally constrained* SCA, or tcSCA) extracts a desired source signal using a temporal reference signal, while the second algorithm (spatially constrained SCA, or spcSCA) uses a reference vector exploiting spatial information. Both algorithms minimize *diversity* (anti–sparsity) of the extracted source signal through the use of these improved non–convex sparsity functions. The proposed algorithms are therefore expected to perform better than ℓ_1–based methods, such as the SCA-R algorithm [19]. In terms of mean–squared error (between the desired source signal and the extracted one), it is shown in Sect. VI that the tcSCA algorithm performs better than the SCA-R algorithm for the case of noise–free measurements ($\boldsymbol{W} = 0$ in (1)), while the two algorithms are roughly equivalent for the case of noisy measurements. However, the tcSCA is significantly faster in both cases. Moreover, the proposed algorithms are also expected to outperform the least-squares approach (3) and the cICA algorithm for sparse signals. In addition, and as will be shown later, tcSCA does not depend on a critical threshold parameter, as does the cICA algorithm. We also propose a rigorous method for calculating the regularization parameter γ, which is used with

the proposed spcSCA algorithm.

III. Extracting the sparsest source signal

In this section we consider the problem of extracting the "*sparsest*" source signal from the linear mixture (1). We start by considering the simplest case for which the *support* of the desired source signal is known a priori, where the support of a signal $s(t)$ is defined as $Sp(s(t)) \triangleq \{t : s(t) \neq 0\}$. Even though this case occurs rarely in practice, the solution of this problem provides the main idea of the algorithms to follow.

A. Support Constrained SCA (scSCA)

For the time being, consider the case where the set of indices of the zero entries of the sparsest source signal, say the ith source, is known. Let $\mathcal{I}_i$ refer to these indices, i.e., $\boldsymbol{s}^i(\mathcal{I}_i) = 0$. Note that $\mathcal{I}_i$ is the complement of $Sp(\boldsymbol{s}^i)$, the support of $\boldsymbol{s}^i$. Since $\boldsymbol{s}^i$ is the sparsest source signal, then there are some indices for which $\boldsymbol{s}^j(\mathcal{I}_i) \neq 0, \forall j \neq i$, i.e., $||\boldsymbol{s}^j(\mathcal{I}_i)||_{\ell_2} > 0$. Therefore, if $\boldsymbol{B}$ is a separating matrix, and neglecting the permutation indeterminacy and the presence of noise[1] , the entries of the ith row of the matrix

$$\boldsymbol{Y}_{\mathcal{I}_i} = \boldsymbol{B}\boldsymbol{X}_{\mathcal{I}_i} \tag{5}$$

are all zeros, where $\boldsymbol{X}_{\mathcal{I}_i}$ is a sub–matrix of $\boldsymbol{X}$ with columns corresponding to the indices in $\mathcal{I}_i$, i.e., $\boldsymbol{X}_{\mathcal{I}_i} = \boldsymbol{X}(:, \mathcal{I}_i)$ using Matlab® notation. Accordingly, in this case, the ith row of the separating matrix $\boldsymbol{B}$ can be obtained by solving the following optimization problem

$$\hat{\boldsymbol{b}}^i = \arg\min_{\bar{\boldsymbol{b}}} \mathcal{J}_{sup}(\bar{\boldsymbol{b}}) \triangleq ||\bar{\boldsymbol{b}}\boldsymbol{X}_{\mathcal{I}_i}||_{\ell_2}^2 \quad \text{subject to} \quad ||\bar{\boldsymbol{b}}||_{\ell_2} = 1, \tag{6}$$

where the constraint $||\bar{\boldsymbol{b}}||_{\ell_2} = 1$ prevents the trivial solution $\bar{\boldsymbol{b}} = 0$.

The solution of relevance to (6) is recognized to be

$$\left(\hat{\boldsymbol{b}}^i\right)^T = \text{eig}_{min}(\boldsymbol{X}_{\mathcal{I}_i}\boldsymbol{X}_{\mathcal{I}_i}^T), \tag{7}$$

where $\text{eig}_{min}(\boldsymbol{R})$ is the *minimal signal* eigenvector of $\boldsymbol{R}$, i.e., the eigenvector corresponding to the n^{th}–largest eigenvalue.

In most cases of interest, the number of sources n is unknown, and so therefore must be estimated. Fortunately, this is a well–studied problem for which many attractive solutions exist. Prominent among

[1]This assumption simplifies the exposition of the problem, but the derived algorithms are also applicable for the noisy measurements as shown in Sec. VI

these methods are the minimum description length (MDL) and the Akaike information criteria (AIC) algorithms, e.g., [24]. An additional method is presented in [25]. These methods are applied to the matrix $\boldsymbol{R}_x = \boldsymbol{X}\boldsymbol{X}^T$.

Equation (7) states that the separating vector of the sparsest source signal can be readily estimated from the measured signals once the indices at which this source signal equals zero are known. We refer to this algorithm as support constrained SCA (scSCA). A useful application of the scSCA algorithm in the field of EEG/MEG signal processing is removing the (50/60 Hz) line voltage interference from the recorded EEG/MEG data without using a notch filter. Even though the (50/60 Hz) line voltage interference is not sparse in the time domain, it is very sparse in the frequency domain and its support is unambiguously known. Hence, the separating vector associated with this signal can be estimated in the frequency domain using (7). See Example 4 in Section VI.

Generally, with the exception of the above example, the scSCA algorithm is of limited value, since the support of the sparsest source signal is usually unknown. In the next subsection we review an iterative algorithm that converges in its limit to (7). Further details are available in [18], [23].

B. An iterative algorithm for extracting the sparsest source signal

Since the desired source signal is assumed to be the sparsest, the separating row vector associated with this signal can be estimated by solving the following optimization problem

$$\hat{\boldsymbol{b}} = \arg\min_{\bar{\boldsymbol{b}}} ||\bar{\boldsymbol{y}}||_{\ell_0} = ||\bar{\boldsymbol{b}}\boldsymbol{X}||_{\ell_0} \quad \text{subject to} \quad ||\bar{\boldsymbol{b}}||_{\ell_2} = 1, \tag{8}$$

where $||\bar{\boldsymbol{y}}||_{\ell_0} = \sum_t |y[t]|^0$ counts the number of nonzero entries in the separated signal $\bar{\boldsymbol{y}} = \bar{\boldsymbol{b}}\boldsymbol{X}$, and the "hat" denotes an optimized solution. Unfortunately, this objective function is flat over all values of y except at $y = 0$, where its value is zero. This implies that any gradient descent technique will fail to converge to any of the sparse source signals. This limitation motivated researchers to replace the ℓ_0-norm by other more efficient functions that encourage the sparsity of its argument.

Over the last two decades, the ℓ_0-norm has been replaced by the convex ℓ_1–norm in an effort to improve computational efficiency. See [26] and the references therein for the history of ℓ_1–minimization and its applications. As previously stated, there is a class of non–convex objective functions that perform better than the ℓ_1–norm in encouraging the sparsity of a given solution vector, e.g., [20]. This class includes functions of the form $g(\bar{\boldsymbol{y}}) = \sum_i g_c(y[i])$, where $g_c(y)$ has the following property

Property 1: The function $g_c(y) : \mathbb{R} \to \mathbb{R}$ is differentiable, sign invariant, and concave-and-monotonically increasing on the nonnegative orthant $\mathcal{O}_1$.

Examples of such functions that have been extensively utilized in the literature are $g_{log}(s) = \log(|s|)$ and $g_q(s) = |s|^q$, where $0 < q < 1$ [20], [21], [27], [28].

The main difficulty associated with replacing the ℓ_0–norm by a function $g(\bar{\boldsymbol{y}}) = \sum_i g_c(y[i])$, rather than the ℓ_1–norm, is the non–convexity of $g(\bar{\boldsymbol{y}})$. However, in [22], [23] we have developed an efficient technique for minimizing this class of non–convex functions. Given an initial estimate of the separating vector $\bar{\boldsymbol{b}}_0$, the proposed technique is based on iteratively replacing the non–convex objective function $g(\bar{\boldsymbol{y}})$ by a convex function $f(\bar{\boldsymbol{y}})$. Selecting $f(\bar{\boldsymbol{y}})$ to be a quadratic convex function, and for a special selection of the parameters of this quadratic function, at the kth iteration $f_k(\bar{\boldsymbol{y}})$ takes the expression $f_k(\bar{\boldsymbol{y}}) = f_k(\bar{\boldsymbol{b}}\boldsymbol{X}) = ||\bar{\boldsymbol{b}}\boldsymbol{X}\boldsymbol{\Sigma}_k||_{\ell_2}^2$ [23], where $\boldsymbol{\Sigma}_k = \mathrm{diag}(\bar{\boldsymbol{\sigma}}_k)$, and the tth entry of $\bar{\boldsymbol{\sigma}}_k$ is given by

$$\sigma_k[t] = \sqrt{\frac{g_c'(y_k[t])}{2y_k[t]}}, \tag{9}$$

where $g_c'(y_k[t])$ is the gradient of $g_c(y)$ at $y = y_k[t]$, $\bar{\boldsymbol{y}}_k = \bar{\boldsymbol{b}}_k\boldsymbol{X}$, and $\bar{\boldsymbol{b}}_k$ is an estimate of the separating vector $\bar{\boldsymbol{b}}$ at the kth iteration. Therefore, replacing $g(\bar{\boldsymbol{y}})$ by $f_k(\bar{\boldsymbol{y}})$, and starting from an initial estimate of the separating vector, an estimate of the separating vector at the $(k+1)$th iteration can be obtained by solving the following optimization problem, which minimizes $f_k(\bar{\boldsymbol{y}})$ at each step:

$$\hat{\boldsymbol{b}}_{k+1} = \arg\min_{\bar{\boldsymbol{b}}} ||\bar{\boldsymbol{b}}\boldsymbol{X}\boldsymbol{\Sigma}_k||_{\ell_2}^2 \quad \text{subject to} \quad ||\bar{\boldsymbol{b}}||_{\ell_2} = 1, \tag{10}$$

the solution to which is

$$\hat{\boldsymbol{b}}_{k+1}^T = \mathrm{eig}_{min}\left(\boldsymbol{X}_k\boldsymbol{X}_k^T\right), \quad k = 0, 1, \ldots, \tag{11}$$

where $\boldsymbol{X}_k = \boldsymbol{X}\boldsymbol{\Sigma}_k$. As proved in [22], [23] the solution vector (11) is guaranteed to converge to a fixed point of the original optimization problem

$$\min_{\bar{\boldsymbol{b}}} g(\bar{\boldsymbol{b}}\boldsymbol{X}) \text{ subject to } ||\bar{\boldsymbol{b}}||_{\ell_2} = 1 \tag{12}$$

where $g(\bar{\boldsymbol{y}}) = \sum_i g_c(y[i])$, and $g_c(\cdot)$ obeys property 1.

Note that the tth column of $\boldsymbol{X}_k$ is a scaled version of the tth column of $\boldsymbol{X}$, with a scaling factor given by $\sigma_k[t]$ defined in (9). Since $\sigma_k[t]$ is inversely proportional to $y_k[t]$, multiplying the data matrix $\boldsymbol{X}$ by the diagonal matrix $\boldsymbol{\Sigma}_k$ has the effect of selecting the columns of the data matrix with indices corresponding to the small entries of the estimated source vector $\bar{\boldsymbol{y}}_k$. Thus, the algorithm automatically selects the support of $\boldsymbol{s}$. As the algorithm converges to the desired sparse signal, the diagonal matrix becomes more selective and, as $k \to \infty$, the eigenvectors of $\left(\boldsymbol{X}_k\boldsymbol{X}_k^T\right)$ converges to those of $\left(\boldsymbol{X}_{\mathcal{I}_i}\boldsymbol{X}_{\mathcal{I}_i}^T\right)$, defined in (7).

The main problem associated with the proposed algorithm is that the values of $\sigma_k[t]$ are not upper bounded. As the algorithm converges to the desired separating vector, many entries of the separated signal $\bar{\boldsymbol{y}}_k$ will have very small values. As a result, small differences between the small entries of $\bar{\boldsymbol{y}}_k$ can result in large weighting differences of the corresponding columns of the matrix $\boldsymbol{X}_k = \boldsymbol{X}\boldsymbol{\Sigma}_k$. This in turn will affect the value of the minimal eigenvector of the matrix $\left(\boldsymbol{X}_k\boldsymbol{X}_k^T\right)$. To reduce this effect we propose changing the expression of $\sigma_k[t]$ by incorporating a constant parameter $\varrho > 0$ as follows. From Property 1, it is straight forward to show that both $g_c'(y_k[t])$ and $y_k[t]$ have the same sign. Therefore, we suggest changing the expression of $\sigma_k[t]$ into the following expression

$$\sigma_k[t] = \sqrt{\frac{|g_c'(y_k[t])|}{2|y_k[t]|}}\Big|_{|y_k[t]|\leftarrow|y_k[t]|+\varrho}. \tag{13}$$

Eq. (13) means that the value of $\sigma_k[t]$ is calculated by first calculating the quantity $\sqrt{|g_c'(y_k[t])|/2|y_k[t]|}$, and then replacing $|y_k[t]|$ by $|y_k[t]| + \varrho$. For example, for $g_c(y) = |y|^q$ we have $g_c'(y) = qy|y|^{q-2}$, and hence the quantity $\sqrt{|g_c'(y)|/2|y|}$ equals $\sqrt{\frac{q}{2}|y|^{q-2}}$. Therefore, the value of σ in this case is given by $\sigma = \sqrt{\frac{q}{2}(|y| + \varrho)^{q-2}}$. The expressions for σ corresponding to various objective functions that obey Property 1 are presented in Table I.

Unfortunately, we found that the performance of the proposed algorithm depends on the value of ϱ. To overcome this difficulty, we follow a procedure similar to the one suggested in [28]. Here, ϱ is initiated to a relatively large value, e.g. $\varrho = 0.1$. Its value is then reduced by a factor of 2 at iterations where the condition $||\bar{\boldsymbol{y}}_k - \bar{\boldsymbol{y}}_{k-1}||/||\bar{\boldsymbol{y}}_k|| < \sqrt{\varrho}/2$ is satisfied. The value of ϱ is also lower bounded by 10^{-6}.

The use of these non–convex sparsity functions results in a non–convex optimization problem. Thus, the final solution is sensitive to the initial value of $\bar{\boldsymbol{b}}$. In our implementations, good performance is obtained in this respect when the separating vector is initialized using the available reference vectors, as explained further in the sequel.

The algorithms proposed in this section are suitable for extracting the sparsest source signal. Therefore, if one is interested in extracting a single source signal that has certain properties, then (ideally) the desired source signal will be extracted using the algorithm proposed in this section provided it is the sparsest source signal among all the sources. However, if the desired source signal is not the sparsest one, or if all sources have the same sparsity, then there is no guarantee that the extracted source is the desired one. A reasonable approach to overcome this difficulty is to explicitly incorporate all (or some of) the information available about the desired source signal into the optimization problem (10). In the next two sections we propose two different algorithms that incorporate either temporal or spatial information into the optimization problem (10), respectively.

TABLE I

SOME OBJECTIVE FUNCTIONS $g_c(y)$ THAT CAN BE USED WITH THE PROPOSED ALGORITHM.

$g_c(y)$	$g_c'(y)$	$\sigma = \sqrt{\lvert g_c'(y)\rvert/2\lvert y\rvert}\Big\rvert_{\lvert y\rvert\leftarrow\lvert y\rvert+\varrho}$
$\lvert y\rvert^q,\ 0<q<1$	$qy\lvert y\rvert^{q-2}$	$\sqrt{\frac{q}{2}(\lvert y\rvert+\varrho)^{q-2}}$
$\log(\lvert y\rvert)$	$\frac{y}{\lvert y\rvert^2}$	$\frac{1}{\sqrt{2}(\lvert y\rvert+\varrho)}$
$\operatorname{atan}(\lvert y\rvert/\delta),\ 0<\delta\ll 1$	$\frac{\delta y}{\lvert y\rvert(\delta^2+y^2)}$	$\sqrt{\frac{\delta}{2(\lvert y\rvert+\varrho)(\delta^2+(\lvert y\rvert+\varrho)^2)}}$
$\frac{\lvert y\rvert}{\lvert y\rvert+\delta},\ 0<\delta\ll 1$	$\frac{\delta y}{\lvert y\rvert(\delta+\lvert y\rvert)^2}$	$\sqrt{\frac{\delta}{2(\lvert y\rvert+\varrho)(\delta+\varrho+\lvert y\rvert)^2}}$

IV. PROPOSED SCA ALGORITHM WITH A TEMPORAL CONSTRAINT (TCSCA)

In this section we consider the case of finding a separating vector $\bar{\boldsymbol{b}}_t$ such that the extracted source signal $\bar{\boldsymbol{y}} = \bar{\boldsymbol{b}}_t\boldsymbol{X}$ is sparse and close to a temporal reference signal $\bar{\boldsymbol{r}}_t$. Means of determining a suitable $\bar{\boldsymbol{r}}_t$ are discussed later. Given an estimate $\bar{\boldsymbol{b}}_t(k)$ of the temporally constrained separating vector $\bar{\boldsymbol{b}}_t$ at the kth iteration, the optimal separating vector $\hat{\boldsymbol{b}}_t(k+1)$ at the $(k+1)$th iteration satisfies

$$\begin{aligned}\hat{\boldsymbol{b}}_t(k+1) &= \arg\min_{\bar{\boldsymbol{b}}} ||\bar{\boldsymbol{b}}\boldsymbol{X}\boldsymbol{\Sigma}_k||_{\ell_2}^2 + \gamma\mathcal{J}_r(\bar{\boldsymbol{b}}\boldsymbol{X};\bar{\boldsymbol{r}}_t)\\ &\quad \text{subject to} \quad ||\bar{\boldsymbol{b}}||_{\ell_2} = 1.\end{aligned} \tag{14}$$

where $\gamma \geq 0$ is a regularization parameter, $\mathcal{J}_r(\bar{\boldsymbol{b}}\boldsymbol{X};\bar{\boldsymbol{r}}_t)$ is a regularizing function that measures the closeness between the separated signal $\bar{\boldsymbol{y}} = \bar{\boldsymbol{b}}\boldsymbol{X}$ and the reference signal $\bar{\boldsymbol{r}}_t$, and $\boldsymbol{\Sigma}_k$ is as defined in (10).

We consider two convenient choices for the function $\mathcal{J}_r(\bar{\boldsymbol{b}}\boldsymbol{X};\bar{\boldsymbol{r}}_t)$. They are 1) the correlation coefficient between the desired signal and the reference signal, and 2) the squared ℓ_2–distance between the two signals. The respective solutions are then given by the form

$$\hat{\boldsymbol{b}}_{t1}(k+1) = \arg\min_{\bar{\boldsymbol{b}}} \mathcal{J}_{t1}\left(\bar{\boldsymbol{b}}\right) \triangleq ||\bar{\boldsymbol{b}}\boldsymbol{X}\boldsymbol{\Sigma}_k||_{\ell_2}^2 - \gamma\bar{\boldsymbol{b}}\boldsymbol{X}\bar{\boldsymbol{r}}_t^T \tag{15}$$

and

$$\begin{aligned}\hat{\boldsymbol{b}}_{t2}(k+1) &= \arg\min_{\bar{\boldsymbol{b}}} \mathcal{J}_{t2}\left(\bar{\boldsymbol{b}}\right) \triangleq ||\bar{\boldsymbol{b}}\boldsymbol{X}\boldsymbol{\Sigma}_k||_{\ell_2}^2 + \gamma||\bar{\boldsymbol{b}}\boldsymbol{X} - \bar{\boldsymbol{r}}_t||_{\ell_2}^2\\ &\quad \text{each subject to} \quad ||\bar{\boldsymbol{b}}||_{\ell_2} = 1.\end{aligned} \tag{16}$$

It is straightforward to show that the respective unconstrained solutions are

$$\hat{\boldsymbol{b}}_{t1}^{\star}(k+1) = \gamma\bar{\boldsymbol{r}}_t\boldsymbol{X}^T\left(\boldsymbol{X}\boldsymbol{\Sigma}_k^2\boldsymbol{X}^T\right)^{-1} \tag{17}$$

TABLE II

TEMPORALLY CONSTRAINED SCA

$[\bar{\boldsymbol{y}}, \bar{\boldsymbol{b}}]$ = tcSCA($\boldsymbol{X}$, $\bar{\boldsymbol{r}}_t$, α)

Select an objective function from Table I, a small threshold parameter ϵ, an initial value for ϱ, e.g., $\varrho = 0.1$, and calculate $\bar{\boldsymbol{y}}_0 = \bar{\boldsymbol{r}}_t$.

1) For $k = 0, 1, \ldots$, repeat until convergence:
   - Calculate $\sigma_k[t], \quad t = 1, \ldots, T$ from Table I, and set $\Sigma_k = \mathrm{diag}(\bar{\boldsymbol{\sigma}}_k)$.
   - if $\alpha = 1$, then calculate the value of γ using the L–curve method (see text for more details).
   - Calculate the separating vector:
     $\bar{\boldsymbol{b}}^{\star}_{k+1} = \bar{\boldsymbol{r}}_t \boldsymbol{X}^T \left(\boldsymbol{X} \left(\boldsymbol{\Sigma}_k^2 + \alpha\gamma \boldsymbol{I} \right) \boldsymbol{X}^T \right)^{-1}$
     $\bar{\boldsymbol{b}}_{k+1} = \dfrac{\bar{\boldsymbol{b}}^{\star}_{k+1}}{||\bar{\boldsymbol{b}}^{\star}_{k+1}||_{\ell_2}}$.
   - Get a new estimate of the desired signal: $\bar{\boldsymbol{y}}_{k+1} = \bar{\boldsymbol{b}}_{k+1} \boldsymbol{X}$.
   - if $||\bar{\boldsymbol{y}}_{k+1} - \bar{\boldsymbol{y}}_k||_{\ell_2} / ||\bar{\boldsymbol{y}}_{k+1}||_{\ell_2} < \epsilon$, stop.
   - if $||\bar{\boldsymbol{y}}_{k+1} - \bar{\boldsymbol{y}}_k||_{\ell_2} / ||\bar{\boldsymbol{y}}_{k+1}||_{\ell_2} < \sqrt{\varrho}/2$, update $\varrho \leftarrow \varrho/2$.
   - $\varrho = \max\left\{ \varrho \ , \ 10^{-6} \right\}$
2) Output $\bar{\boldsymbol{y}}_{k+1}$ and $\bar{\boldsymbol{b}}_{k+1}$ as the solutions.

End

---

and

$$\hat{\boldsymbol{b}}^{\star}_{t2}(k+1) = \gamma \bar{\boldsymbol{r}}_t \boldsymbol{X}^T \left(\boldsymbol{X} \left(\boldsymbol{\Sigma}_k^2 + \gamma \boldsymbol{I} \right) \boldsymbol{X}^T \right)^{-1}. \tag{18}$$

Due to the inherent scale ambiguity of the BSS probelm, if $\bar{\boldsymbol{b}}$ is a separating vector, then $c\bar{\boldsymbol{b}}, c \neq 0$ is also a separating vector. Therefore the constraint can be implemented simply by taking

$$\hat{\boldsymbol{b}}_t(k+1) = \frac{\bar{\boldsymbol{b}}^{\star}_{tp}}{||\bar{\boldsymbol{b}}^{\star}_{tp}||_{\ell_2}} \tag{19}$$

where $p \in [1, 2]$ corresponding to either (15) or (16). In the correlation case, due to the normalization step, the value of $\bar{\boldsymbol{b}}_{t1}$ does not depend on the value of the regularization parameter γ.

For both approaches, a new estimate of the desired signal is calculated from the new separating vector, and the diagonal elements of $\boldsymbol{\Sigma}_k$ are updated using (13). The process is repeated until convergence. The regularization parameter γ in (16) can be calculated by first converting the objective function $\mathcal{J}_{t2}\left(\bar{\boldsymbol{b}}\right)$ into a "standard form" for Tikhonov regularization, then calculating the regularization parameter using one of the standard techniques, e.g., the L–curve method and the generalized cross–validation approach. These two methods, as well as other methods [34], are implemented in the Regularization Toolbox[2].

[2]This is a free software available at http://www2.imm.dtu.dk/~pch/Regutools/

The objective function $\mathcal{J}_{t2}\left(\bar{\boldsymbol{b}}\right)$ can be converted into standard form by defining $\bar{\boldsymbol{d}} = \bar{\boldsymbol{b}}\boldsymbol{X}\boldsymbol{\Sigma}_k$ and $\boldsymbol{C}_0 = \boldsymbol{\Sigma}_k^{-1}\boldsymbol{X}^T\left(\boldsymbol{X}\boldsymbol{X}^T\right)^{-1}\boldsymbol{X}$, then expressing $\mathcal{J}_{t2}\left(\bar{\boldsymbol{b}}\right)$ as

$$\mathcal{J}_{t2}\left(\bar{\boldsymbol{d}}\right) = ||\bar{\boldsymbol{d}}||_{\ell_2}^2 + \gamma||\bar{\boldsymbol{d}}\boldsymbol{C}_0 - \bar{\boldsymbol{r}}_t||_{\ell_2}^2. \tag{20}$$

The proposed algorithm is summarized in Table II. The input parameter α is a flag indicating which method is used as a measure of closeness, i.e., $\alpha = 1$ for the MSE case, while $\alpha = 0$ for the correlation coefficient case. On the basis of our simulation examples in Sect. VI, we have observed that initializing the extracted source signal $\bar{\boldsymbol{y}}$ to the reference vector $\bar{\boldsymbol{r}}_t$, as shown in Table II, results in convergence close to the true separating vector with high probability.

V. Proposed SCA algorithm with a spatial constraint (spcSCA)

In this section we propose a new algorithm for estimating the desired signal when prior information is available about a source's spatial configuration. For our purposes, this spatial information is most conveniently represented through the mixing vector $\boldsymbol{a}_j$ associated with the respective source. Henceforth, we refer to the derived algorithm as spatially constrained SCA (spcSCA).

In the field of EEG/MEG signal processing, the vector $\boldsymbol{a}_j$ represents the *topographic map* corresponding to the jth source. In many cases, the topographic maps of some of the desired signals can be determined *a priori*, or can be estimated by visual inspection. In these cases, a spatial reference vector $\boldsymbol{r}_s \in \mathbb{R}^n$ for each of these maps can be readily constructed. For example, since ocular artifacts contaminate only the frontal electrodes, the spatial reference vector for these types of sources can consist only of zeros and ones, where the indices of the "ones" correspond to the frontal electrodes. Further details are given in Example 3 of Section VI.

In this case, a separating vector can be estimated using an objective function similar to (14) with the regularizing function $\mathcal{J}_r(\bar{\boldsymbol{b}}\boldsymbol{X}; \bar{\boldsymbol{r}}_t)$ replaced by $\mathcal{J}_r(\hat{\boldsymbol{a}}; \boldsymbol{r}_s) = -\hat{\boldsymbol{a}}^T\boldsymbol{r}_s$, where $\hat{\boldsymbol{a}}$ is an estimate of the mixing column $\boldsymbol{a}$ associated with the desired source signal.

The regularizing function can be expressed in terms of the separating row vector $\bar{\boldsymbol{b}}$ by expressing $\hat{\boldsymbol{a}}$ in terms of $\bar{\boldsymbol{b}}$. Towards that end, assume that the separating vector $\bar{\boldsymbol{b}}$ is already known, and hence the separated signal is given by $\bar{\boldsymbol{y}} = \bar{\boldsymbol{b}}\boldsymbol{X}$. Therefore, the least-squares estimate of the mixing vector $\boldsymbol{a}$ is the solution of the following optimization problem [7]

$$\hat{\boldsymbol{a}} = \arg\min_{\boldsymbol{a}} = \left|\left|\boldsymbol{X} - \boldsymbol{a}\bar{\boldsymbol{y}}\right|\right|_F^2 \tag{21}$$

TABLE III

SPATIALLY CONSTRAINED SCA

$[\bar{\boldsymbol{y}}, \bar{\boldsymbol{b}}]$ = spcSCA($\boldsymbol{X}$, $\boldsymbol{r}_s$)
Select an objective function from Table I, a small threshold parameter ϵ, an initial value for ϱ, e.g., $\varrho = 0.1$, and calculate $\boldsymbol{R}_x = \boldsymbol{X}\boldsymbol{X}^T$. Initialize the separating vector as $\bar{\boldsymbol{b}}_0 = \boldsymbol{r}_s^T \boldsymbol{R}_x^{-1}$, and calculate $\bar{\boldsymbol{y}}_0^+ = \bar{\boldsymbol{b}}_0 \boldsymbol{X}$, then set the initial estimate of the desired source as $\bar{\boldsymbol{y}}_0 = \frac{\bar{\boldsymbol{y}}_0^+}{\|\bar{\boldsymbol{y}}_0^+\|_{\ell_2}}$ 1) For $k = 0, 1, \ldots$, repeat until convergence: • Calculate $\sigma_k[t], \quad t = 1, \ldots, T$ from Table I, and set $\Sigma_k = \mathrm{diag}(\bar{\boldsymbol{\sigma}}_k)$. • Calculate the separating vector: $\bar{\boldsymbol{b}}_{k+1} = \boldsymbol{r}_s^T \boldsymbol{R}_x \left(\boldsymbol{X}\boldsymbol{\Sigma}_k^2 \boldsymbol{X}^T\right)^{-1}$ • Get a new estimate of the desired signal: $\bar{\boldsymbol{y}}_{k+1} = \frac{\bar{\boldsymbol{b}}_{k+1}\boldsymbol{X}}{\left\|\bar{\boldsymbol{b}}_{k+1}\boldsymbol{X}\right\|_{\ell_2}}$. • if $\|\bar{\boldsymbol{y}}_{k+1} - \bar{\boldsymbol{y}}_k\|_{\ell_2} < \epsilon$, stop. • if $\|\bar{\boldsymbol{y}}_{k+1} - \bar{\boldsymbol{y}}_k\|_{\ell_2} < \sqrt{\varrho}/2$, update $\varrho \leftarrow \varrho/2$. • $\varrho = \max\{\varrho, 10^{-6}\}$ 2) Output $\bar{\boldsymbol{y}}_{k+1}$ and $\bar{\boldsymbol{b}}_{k+1}$ as the solutions. End

where $\left|\left| \cdot \right|\right|_F$ denotes the Frobenious norm. The solution of this optimization problem is readily given by

$$\hat{\boldsymbol{a}} = \frac{\boldsymbol{X}\bar{\boldsymbol{y}}^T}{\|\bar{\boldsymbol{y}}\|_{\ell_2}^2} = \frac{\boldsymbol{X}\boldsymbol{X}^T\bar{\boldsymbol{b}}^T}{\|\bar{\boldsymbol{y}}\|_{\ell_2}^2}, \tag{22}$$

Substituting this into the expression for $\mathcal{J}_r(\hat{\boldsymbol{a}}; \boldsymbol{r})$, the optimal spatially constrained separating vector $\hat{\boldsymbol{b}}_s$ satisfies the following criterion, which corresponds to (14)

$$\begin{aligned} \hat{\boldsymbol{b}}_s &= \arg\min_{\bar{\boldsymbol{b}}} \|\bar{\boldsymbol{b}}\boldsymbol{X}\boldsymbol{\Sigma}_k\|_{\ell_2}^2 - \gamma \frac{\bar{\boldsymbol{b}}\boldsymbol{X}\boldsymbol{X}^T\boldsymbol{r}_s}{\|\bar{\boldsymbol{y}}\|_{\ell_2}^2} \\ &\qquad \text{subject to} \quad \|\bar{\boldsymbol{b}}\|_{\ell_2} = 1. \end{aligned} \tag{23}$$

The presence of the term $\|\bar{\boldsymbol{y}}\|_{\ell_2}^2$ in the denominator above makes the problem unsolvable by analytic means. However, since $\bar{\boldsymbol{y}} = \bar{\boldsymbol{b}}\boldsymbol{X}$ and due to the inherent scale ambiguity in recovering $\bar{\boldsymbol{y}}$, then a constraint on $\|\bar{\boldsymbol{b}}\|_{\ell_2}$ is equivalent to a constraint on $\|\bar{\boldsymbol{y}}\|_{\ell_2}$. Hence, we are at liberty to choose $\|\bar{\boldsymbol{y}}\|_{\ell_2} = 1$. The problem then becomes

$$\begin{aligned} \hat{\boldsymbol{b}}_s &= \arg\min_{\bar{\boldsymbol{b}}} \mathcal{J}_s(\bar{\boldsymbol{b}}; \boldsymbol{r}) \triangleq \|\bar{\boldsymbol{b}}\boldsymbol{X}\boldsymbol{\Sigma}_k\|_{\ell_2}^2 - \gamma \bar{\boldsymbol{b}}\boldsymbol{R}_x\boldsymbol{r}_s \\ &\qquad \text{subject to} \quad \|\bar{\boldsymbol{y}}\|_{\ell_2}^2 = \bar{\boldsymbol{b}}\boldsymbol{R}_x\boldsymbol{b} = 1, \end{aligned} \tag{24}$$

where $\boldsymbol{R}_x = \boldsymbol{X}\boldsymbol{X}^T$. It is readily verified that an unconstrained solution $\hat{\boldsymbol{b}}_s^\star$ to the above satisfies

$$\boldsymbol{X}\boldsymbol{\Sigma}_k^2\boldsymbol{X}^T\hat{\boldsymbol{b}}_s^\star = \gamma\boldsymbol{R}_x\boldsymbol{r}_s. \tag{25}$$

provided $\boldsymbol{X}\boldsymbol{\Sigma}_k^2\boldsymbol{X}^T$ is full rank. Using the same argument as for the temporally constrained case, the constraint can be incorporated by assigning

$$\hat{\boldsymbol{b}}_s = \frac{\hat{\boldsymbol{b}}^\star}{\left|\left|\hat{\boldsymbol{b}}^\star\boldsymbol{X}\right|\right|_{\ell_2}}. \tag{26}$$

Given the new $\hat{\boldsymbol{b}}_s$, a new $\bar{\boldsymbol{y}} = \hat{\boldsymbol{b}}_s\boldsymbol{X}$ is found, whereupon the diagonal matrix $\boldsymbol{\Sigma}_{k+1}$ is evaluated. The procedure is repeated until convergence.

The separating vector for the spcSCA algorithm is initialized using the available reference vector $\bar{\boldsymbol{r}}_s$ as an estimate of the respective steering vector $\boldsymbol{a}$. With reference to (22), and assuming that $||\bar{\boldsymbol{y}}||_{\ell_2} = 1$, the separating vector can be initialized as

$$\bar{\boldsymbol{b}}_0 = \boldsymbol{r}_s^T\boldsymbol{R}_x^{-1}. \tag{27}$$

The overall algorithm is summarized in Table III. Note that the value of the solution vector does not depend on the value of the regularization parameter γ due to the normalization step.

VI. Simulation results

In this section we present four different examples that demonstrate the ability of the proposed algorithms in extracting a desired sparse source signal. In the first example we provide a comparison between the proposed algorithms, while in the second example we compare the performance of the proposed tcSCA algorithm with the least-squares solution (4) and the SCA-R algorithm [19]. In these two examples, the comparison between the algorithms is made in terms of the average convergence time i.e., the average of the total computational times required for the respective algorithm to reach convergence, and the normalized mean square error (NMSE) between the extracted signal and the desired source signal. The NMSE between two vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ is defined as

$$NMSE = \left|\left|\frac{\boldsymbol{x}}{||\boldsymbol{x}||_{\ell_2}} - \frac{\boldsymbol{y}}{||\boldsymbol{y}||_{\ell_2}}\right|\right|_{\ell_2}. \tag{28}$$

The remaining two examples demonstrate the ability of the cSCA algorithms in extracting some desired signals from real EEG data when different prior information about the desired signals is available. In all these examples, the first tcSCA algorithm is used, i.e., we set $\alpha = 0$ in Table II. Moreover, the non–convex objective function used with the proposed algorithms is selected as $g_c(y) = \mathrm{atan}(|y|/\delta)$ with the value of $\delta = \varrho/10$, where ϱ is calculated as in Tables II and III.

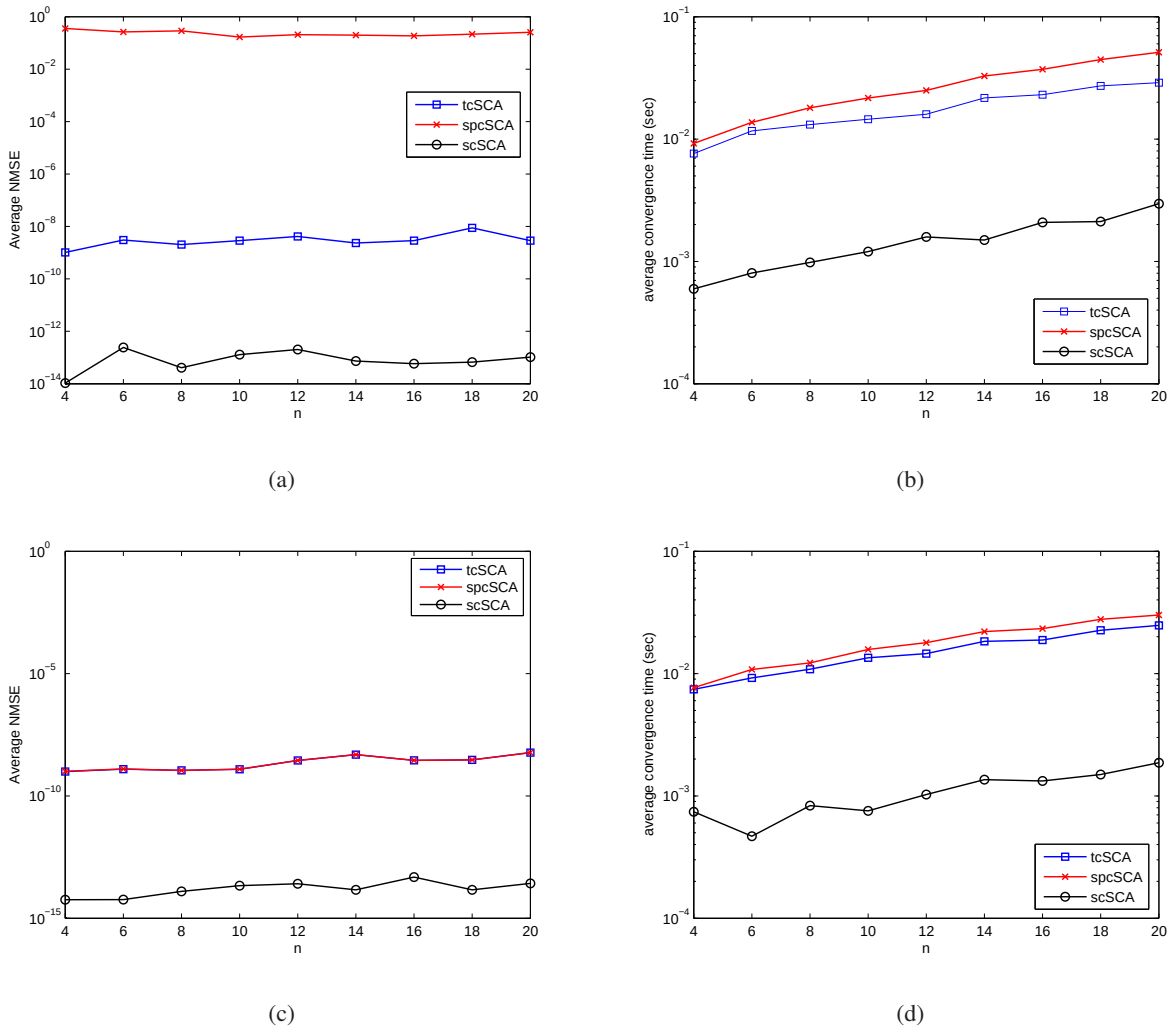


Fig. 1. Comparison between the tcSCA, spcSCA, and scSCA algorithms. (a)-(b) The average NMSE and the average convergence time, respectively, for the case of all sources have the same amplitude, (c)-(d) The average NMSE and the average convergence time, respectively, for the case when the amplitude of the desired source signal is 5 times larger than that of the other source signals.

Example 1. *Comparison between the tcSCA, spcSCA, and scSCA algorithms in the noise-free case:* In this example we compare the performance of the three proposed cSCA algorithms in estimating a desired sparse source signal in the absence of noise. We assign $m = n$, and we assume the number of sources is known. The experiment is performed for the value $T = 1000$, for the cases $n = 4, 6, \ldots, 20$. For each value of n, a sparse source matrix $\boldsymbol{S} \in \mathbb{R}^{n\times 1000}$ is generated such that 90% of its columns satisfy the *disjoint orthogonality* property. The disjoint orthogonality property of the tth column of $\boldsymbol{S}$ is defined as $s_t[i] \cdot s_t[j] = 0, \forall i \neq j$ [36], i.e., the tth column has at most one nonzero entry. For each row of $\boldsymbol{S}$, the indices of the nonzero entries are randomly selected, and their amplitudes are chosen from a

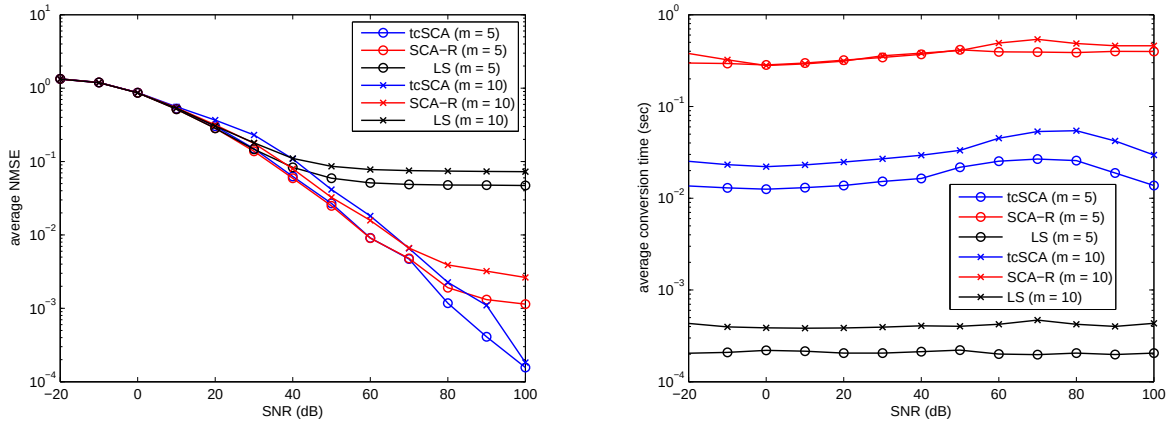
uniform distribution between ± 1. The measured data matrix $\boldsymbol{X} \in \mathbb{R}^{n \times 1000}$ is constructed by multiplying the source matrix $\boldsymbol{S}$ by a square mixing matrix $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ randomly generated from a white normal distribution with zero mean and unit variance. The first source signal $\boldsymbol{s}^1$, the first row of $\boldsymbol{S}$, is selected as the desired source signal.

The reference vector $\bar{\boldsymbol{r}}_t$ for the tcSCA algorithm is constructed as $\bar{\boldsymbol{r}}_t = \boldsymbol{s}^1 + \bar{\boldsymbol{v}}_t$, where $\bar{\boldsymbol{v}}_t$ is zero mean Gaussian random noise. The amplitude of $\bar{\boldsymbol{v}}_t$ is adjusted to produce a 2 dB SNR. The reference vector $\boldsymbol{r}_s$ for the spcSCA algorithm is constructed as $\boldsymbol{r}_s = \boldsymbol{a}_1 + \boldsymbol{v}_s$, where $\boldsymbol{a}_1$ is the mixing column associated with the desired source signal, and $\boldsymbol{v}_s$ is a zero mean Gaussian random noise. The amplitude of $\boldsymbol{v}_s$ is adjusted to produce a 20 dB SNR between $\boldsymbol{a}_1$ and $\boldsymbol{v}_s$. For the scSCA algorithm (7), we assume that the indices at which the desired source signal $\boldsymbol{s}^1$ equals zero are known. For each value of n, the average NMSE and the average convergence time are calculated as the average over 200 different runs, where in each run a new mixing matrix and a new sparse source matrix are generated. The results are shown in Fig. 1(a)–(b).

As shown in these two figures, the scSCA algorithm has the best performance in terms of the average NMSE and the average convergence time, whereas the spcSCA algorithm has the poorest performance. This result is expected since the scSCA algorithm has more information about the desired source signal than the other two algorithms. In addition, the scSCA algorithm has a closed form solution (7), i.e., the algorithm requires only one iteration. On the other hand, the tcSCA algorithm has a very low average NMSE compared with the spcSCA algorithm. The results presented in Fig. 1(a)–(b) are obtained for the value of $\epsilon = 10^{-8}$. The poorer performance of the spcSCA algorithm is due to the following two factors: 1) the spcSCA algorithm does not have any temporal information about the desired source signal, and 2) all the source signals are sparse and have the same sparsity. Therefore, for the spcSCA algorithm to produce satisfactory results, the desired source signal must be distinct (in terms of sparsity or amplitude) from the other source signals.

To check the effect of the amplitude of the desired source signal on the performance of the spcSCA algorithm, the experiment is repeated with the amplitude of the desired source signal 5 times larger than that of the other source signals. The results of this new experiment are shown in Fig. 1(c)–(d). As shown in these figures, the performance of the spcSCA algorithm is remarkably enhanced and it is almost identical to that of the tcSCA algorithm. Similar performance enhancement is expected if the desired source signal is the sparsest source signal.

Example 2. *Comparison between the tcSCA, SCA-R, and least–squares algorithms for the noisy measurements case:* In this example we compare the performance of the proposed tcSCA algorithm



(a) (b)

Fig. 2. Comparison between the tcSCA, SCA-R, and Least–squares algorithms for noisy measurements. (a) The average NMSE; and (b) the average convergence time.

(summarized in Table II) with the least-squares solution (4) and the SCA-R algorithm proposed in [19][3]. In this case the three algorithms are compared for SNR values over the range $-20, \ldots, +100$ dB, and $m = n = [5, 10]$. The SNR value is defined with respect to the desired signal power, rather than total signal power. The experimental conditions are identical to those in Experiment 1. The results are shown in Fig. 2.

As shown in Fig. 2(a), the normalized MSE curve for the noisy measurements case can be divided into three different regions according to the value of the SNR. For the low SNR range ($-20 \leq SNR \leq +20$ dB) the performance of the three algorithms is virtually equivalent, with the error being dominated by the noise. Increasing the SNR beyond +20 dB enhances the performance of both tcSCA and SCA-R, while it has no effect on the performance of the LS algorithm. The poorer performance of the least–squares solution is due to the fact that this solution does not exploit the sparsity of the desired source signal into the objective function (3). For the second range of SNR ($+20 \leq SNR \leq +70$ dB), the performances of tcSCA and SCA-R are still similar. However, increasing the SNR beyond $+70$ dB enhances the performance of the proposed tcSCA algorithm, yet SCA-R exhibits a saturation effect. Since the measured signals in this high range of SNR can be considered as being approximately noise free, the difference in the performance of tcSCA and SCA-R can be interpreted as a consequence of the objective function used by each algorithm for measuring the sparsity of the extracted source signal. Even though the proposed non-convex sparsity functions result in improved performance with regard to

[3]The Matlab code for the SCA-R algorithm was downloaded from the following URL :http://iew3.technion.ac.il/~mcib/.

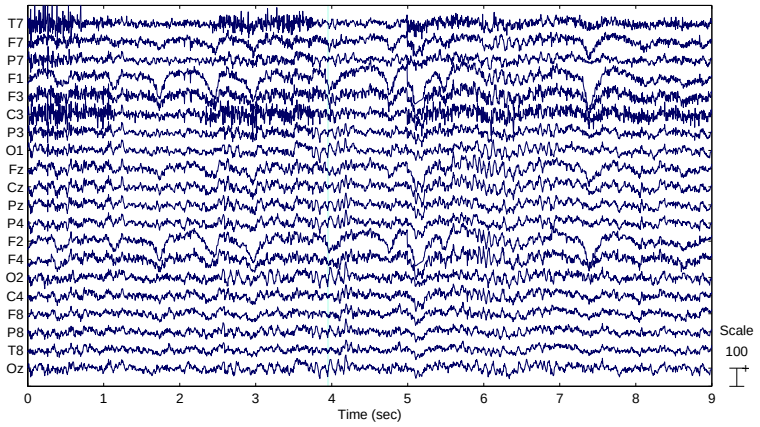
mean–squared error, the effect occurs at such high values of SNR that it is not likely to be of practical significance. However, it is noted that the impact of the proposed technique for minimizing the non-convex objective functions (used for measuring the *diversity* (antisparsity) of the extracted source signal) is much more striking with regard to the required computational effort. As is evident from Fig. 2(b), the proposed tcSCA algorithm displays a one to one–and–a–half order of magnitude improvement in speed over the SCA-R algorithm.

There are many very effective algorithms for computing the minimal eigenvector required by the tcSCA algorithm [31]–[33]. However, for the sake of convenience, in these examples, the complete eigendecomposition is computed using the very efficient built-in Matlab function "*eig.m*", from which the minimum eigenvector is available. This procedure places the tcSCA algorithm at a relative disadvantage with respect to computational comparisons.

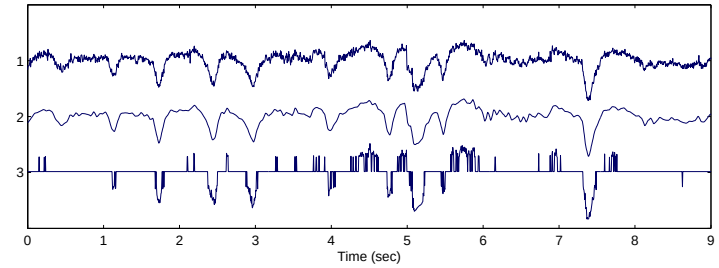
Example 3. *Rejection of eye blink artifact from real EEG data:* In this example we demonstrate the ability of the tcSCA algorithm and the spcSCA algorithm in extracting the eye blink artifact contaminating real EEG data, as shown in Fig. 3(a). The EEG data were collected using 20 electrodes placed according to the International 10-20 System, with sampling frequency 205 Hz. Removing the eye blink artifact from a recorded EEG data using ICA is usually done in three steps. In the first step, called the *processing* step, the ICA algorithm is applied on the EEG data and an estimate of the independent components and the associated mixing columns is obtained. In the second step, which is called the *investigation* step, both the estimated mixing matrix and the estimated components are carefully investigated to identify the components corresponding to the eye blink artifact. The decision is usually made based on the waveform of the suspected component and its mixing column which is usually depicted by a topographic map that represents the contribution of the eye blink artifact at each electrode. If the topographic map and the waveform of one of the components correspond to those of the eye blink artifact, this component is marked as the desired artifact. Finally, in the third step the component corresponding to the eye blink artifact is set to zero and all the remaining components are recombined to reconstruct the clean EEG data.

Clearly, the most tedious step in the aforementioned procedure is the investigation step, during which all the sources and their associated maps are checked carefully to identify the artifact. This process is difficult to automate. To overcome this difficulty, some algorithms have been proposed in the literature to automatically extract [11], [13], [17], [35] or block [37], [38] eye blink artifact from the recorded EEG/MEG data.

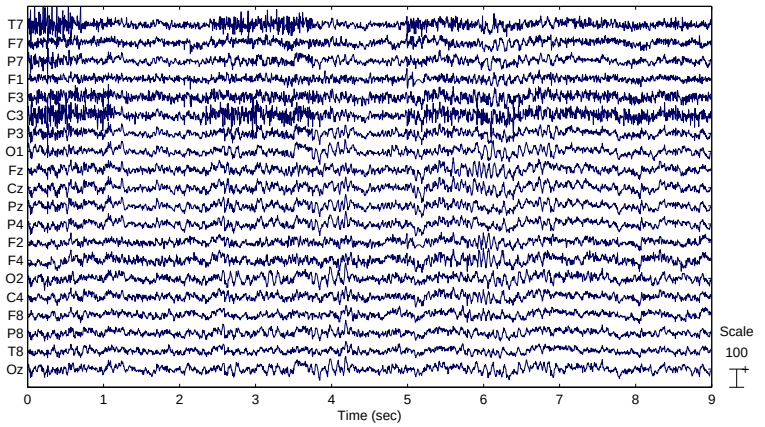
In this example we show that the second step can be completely eliminated using either the tcSCA



(a)

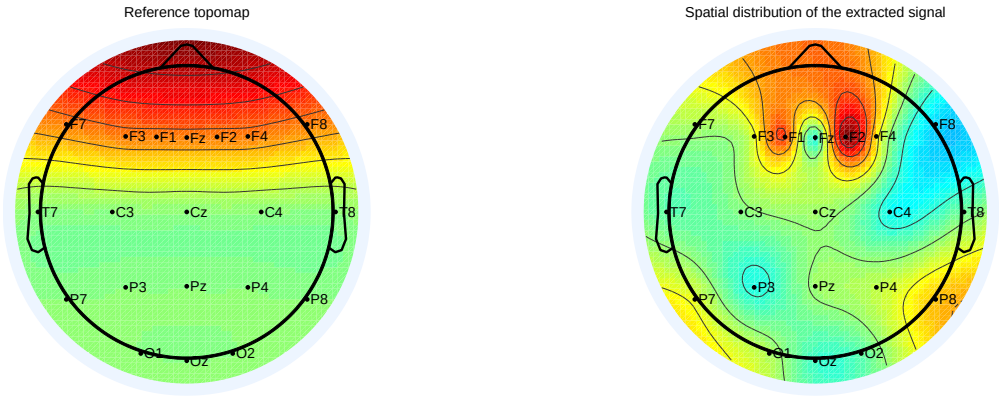


(b)

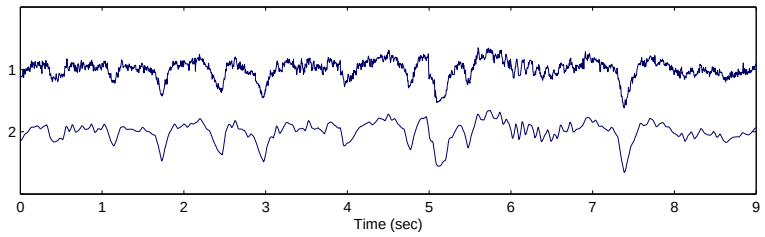


(c)

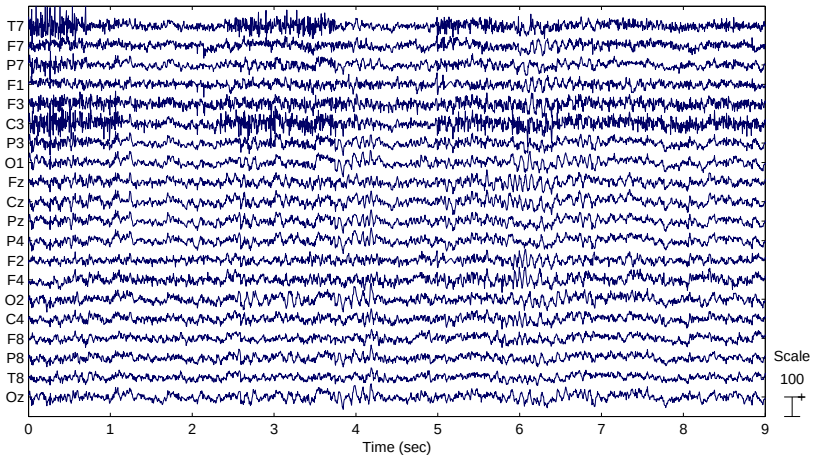
Fig. 3. Automatic removal of eye blink artifact from real EEG data using the tcSCA algorithm. (a) Original EEG data, (b) The signals from top to bottom correspond to the extracted eye blink artifact, the denoised eye blink artifact, and the reference signal, respectively, (c) Clean version of the EEG data shown in (a).



(a) (b)



(c)



(d)

Fig. 4. Automatic removal of eye blink artifact from real EEG data using the spcSCA algorithm. (a) Reference topographic map, (b) The topographic map of the extracted artifact, (c) The upper signal corresponds to the extracted eye blink, while the lower signal is the denoised eye blink artifact, (d) Clean version of the EEG data shown in Fig. 3(a) .

algorithm or the spcSCA algorithm for extracting only the desired source signal. In the tcSCA algorithm we use a reference signal which conveys temporal information about the eye blink artifact. This reference signal is extracted from one of the frontal electrodes, say F1, by setting to zero all samples with absolute value less than a threshold of 50 μv. This reference signal is shown by the third waveform in Fig. 3(b). On the other hand, the reference vector for the spcSCA algorithm is a (20×1) vector of zeros except at those indices corresponding to the frontal electrodes, for which the corresponding value is one. The topographic map of this reference vector is shown in Fig. 4(a).

After running the tcSCA algorithm and the spcSCA algorithm on the EEG data shown in Fig. 3(a), each with its own reference vector, both algorithms converge successfully to the desired artifact signal. The extracted signals are shown by the first waveform in Fig. 3(b) and Fig. 4(c), respectively. The second waveform in each of these figures represents a denoised version of the estimated artifact, where the denoising step is performed using wavelet analysis. Fig. 4(b) shows the topographic map of the extracted source using spcSCA algorithm. Clearly this topographic map corresponds to the expected topographic map of the eye blink artifact, for which F1 and F2 are the most contaminated electrodes. Finally, the clean EEG data is obtained by removing the denoised signals from the original EEG data using the deflation technique [7]. The cleaned EEG data obtained after running the tcSCA algorithm and the spcSCA algorithm are shown in Fig. 3(c) and Fig. 4(d), respectively. Comparing these two figures with the original data set in Fig. 3(a), it is clear that the eye blink artifact is removed successfully in both cases.

Example 4. *Elimination of a line voltage interference from a recorded EEG data using the scSCA algorithm:* In this example we present an application for which the scSCA algorithm (7) is appropriate. In the field of EEG signal processing, the measured signal is usually contaminated by 50/60 Hz line voltage. This line voltage interference is usually removed from the measured EEG data using a notch filter with the correct center frequency. However, this filtering approach is not appropriate if a useful brain signal, e.g., an event related potential (ERP) contains frequencies in this band (i.e., gamma band). To overcome this difficulty, we propose utilizing the scSCA algorithm for extracting the line voltage interference in the frequency domain. The EEG data used in this example is the same EEG data used in Example 3. The average (over all electrodes) power spectral density (PSD) of the EEG data is shown in the upper panel of Fig. 5. As shown in this figure, the EEG data is contaminated by a strong 60 Hz line interference component.

Since the 60 Hz interference signal is not sparse in the time domain but very sparse in the frequency domain, all the analysis is performed in the frequency domain. Let $\boldsymbol{X}(f)$ denote the discrete time

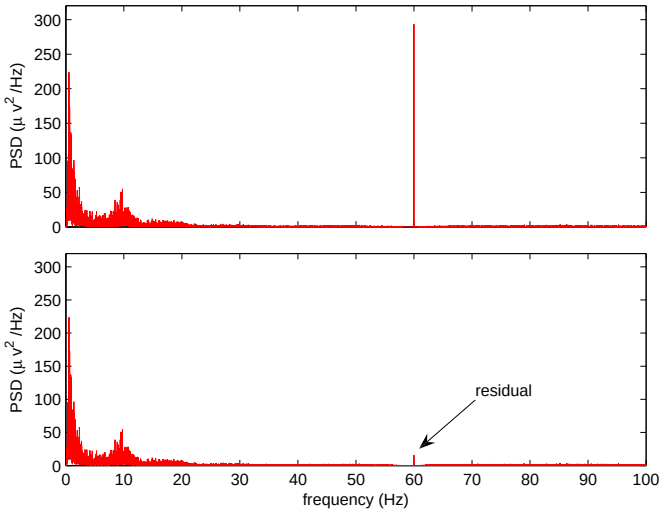


Fig. 5. Removing the line voltage interference from a real EEG data. The upper panel is the average PSD of the original EEG data, while the lower panel is the average PSD of the cleaned EEG data when the separating vector is estimated using (7).

Fourier transform of the original EEG data matrix $\boldsymbol{X}$. The input data matrix $\hat{\boldsymbol{X}}(f) \in \mathbb{R}^{20\times 2T}$ to the scSCA algorithm is constructed by concatenating the real and the imaginary parts of $\boldsymbol{X}(f)$. Assuming that the desired source signal (the 60 Hz line voltage interference) is the ith source $\boldsymbol{s}^i$, then the indices $\mathcal{I}_i$ at which $\hat{\boldsymbol{s}}^i(f) = 0$ are identified as follows, where $\hat{\boldsymbol{s}}^i(f) = \left[\text{real}(\boldsymbol{s}^i(f)) \;\; \text{imag}(\boldsymbol{s}^i(f))\right]$ and $\boldsymbol{s}^i(f)$ is the discrete time Fourier transform of the desired source signal $\boldsymbol{s}^i$. First a time domain reference signal $r(t) = \cos(2\pi 60t)$ is generated. The reference signal $r(t)$ is generated such that it has the same sampling frequency and number of samples as the recorded EEG data. Then $r(t)$ is transformed into the frequency domain using the fast Fourier transform, i.e., $\bar{\boldsymbol{r}}(f) = \text{fft}\,(r(t))$. After calculating $\bar{\boldsymbol{r}}(f)$, a new row vector $\hat{\bar{\boldsymbol{r}}}(f)$ is constructed as $\hat{\bar{\boldsymbol{r}}}(f) = [\text{real}(\bar{\boldsymbol{r}}(f)) \;\; \text{imag}(\bar{\boldsymbol{r}}(f))]$. Since both $r(t)$ and $\boldsymbol{s}^i$ are two sinusoidal signals with the same frequency, the support of $\hat{\bar{\boldsymbol{r}}}(f)$ and $\hat{\boldsymbol{s}}^i(f)$ are identical regardless of the phase difference between the two signals. Therefore, $\mathcal{I}_i$ can be identified as the samples of $\hat{\bar{\boldsymbol{r}}}(f)$ with absolute values less than a certain threshold θ_f. In this example, we selected θ_f to be a fraction of r_{max}, where r_{max} is the largest absolute value of $\hat{\bar{\boldsymbol{r}}}(f)$. After identifying the indices $\mathcal{I}_i$, the matrix $\boldsymbol{X}_{\mathcal{I}_i}$ is constructed from $\hat{\boldsymbol{X}}$ as $\boldsymbol{X}_{\mathcal{I}_i} = \hat{\boldsymbol{X}}(:, \mathcal{I}_i)$, and an estimate of the separating vector $\boldsymbol{b}^i$ is calculated using (7).

After estimating the separating vector $\tilde{\boldsymbol{b}}^i$, an estimate of $\hat{\boldsymbol{s}}^i(f)$ is calculated as $\tilde{\boldsymbol{s}}^i(f) = \tilde{\boldsymbol{b}}^i \hat{\boldsymbol{X}}(f)$. The

samples of $\tilde{\boldsymbol{s}}^i(f)$ with indices in $\mathcal{I}_i$ are set to zero, and the resulting vector is then removed from $\hat{\boldsymbol{X}}(f)$ using the deflation approach. The cleaned time domain EEG signals are then reconstructed from the resulting matrix. The average PSD of the cleaned EEG data is shown in the lower panel of Fig. 5. As shown in this figure, the 60 Hz interference is greatly reduced from the EEG data, with a small residual interference left. One possible reason for the residual interference in the lower panel of Fig. 5 is the possibility that the number of hidden source signals is greater than the number of the measuring sensors, which equals 20 in this example. Therefore, better performance is expected if the number of sensors is greater than or equals the number of hidden source signals.

VII. Conclusions

In this paper we have proposed two different algorithms that can extract a desired *sparse* source signal from the linear mixture (1) when prior information about the desired source signal is available. The first algorithm extracts a desired source signal using a temporal reference signal, while the second algorithm uses a reference vector that conveys spatial information about the mixing column associated with the desired source signal. In deriving the two algorithms, and since the desired source signal is sparse, we used the *diversity* of the separated signal as an objective function. In contrast to the previous approach [19], which utilized the ℓ_1–norm for measuring the diversity of the separated source signal, we used a non–convex objective function, which was shown in the literature to outperform the ℓ_1–norm in encouraging the sparsity of the extracted source signal. In this paper we utilized an efficient technique, which we have previously developed in [22], for minimizing the non–convex objective function. The proposed algorithms were evaluated using simulations in the noisy and noise–free cases and were also successfully employed in removing ocular and line interference artifacts from real EEG data. The numerical results show that the proposed algorithms significantly outperform some of the previous algorithms, in terms of computational efficiency, that have been applied to the same problem.

References

[1] A.J. Bell, and T.J. Sejnowski, " An information-maximization approach to blind source separation and blind deconvolution," *Neural Comp.*, vol. 7, no. 6, pp. 1129-1159, 1995.

[2] T.-P. Jung, C. Humphries, T.-W. Lee, S. Makeig, M. J. McKeown, V. Iragui, and T. Sejnowski, "Extended ICA removes artifacts from electroencephalographic recordings," presented at the Neural Information Processing Systems 10 (Proc. NIPS97), 1998.

[3] R. Vigrio, "Extraction of ocular artifacts from EEG using independent component analysis," *Electroenceph. Clin. Neurophysiol.*, vol. 103, pp. 395404, 1997.

[4] R. Vigrio, V. Jousmki, M. Hmlinen, R. Hari, and E. Oja, "Independent component analysis for identification of artifacts in magnetoencephalographic recordings," Neural Information Processing Systems 10 (Proc. NIPS97), M. I. Jordan, M. J. Kearns, and S. A. Solla, Eds. Cambridge, MA: MIT Press, 1998.

[5] A. Hyvrinen, "Fast and robust fixed-point algorithms for independent component analyis," *IEEE Trans. Neural Net.*, vol. 10, no. 3, pp. 626-634, 1999.

[6] A. Hyvrinen and E. Oja, "Independent component analyis: algorithms and applications," Neural Networks, vol. 13, no. (4-5), pp. 411-430, 2000.

[7] A. Cichocki and S. Amari, *Adaptive Blind Signal and Image Processing: Learning Algorithms and Applications,* John Wiley & Sons, Ltd.

[8] W. Lu and J. C. Rajapakse, "ICA with reference" Proc. 3rd Int. Conf. on Independent Component Analysis and Blind Signal Separation: *ICA2001*, pp 1201255.

[9] L. K. L. Joshua and J. C. Rajapackse, "Extraction of event-related potentials from EEG signals using ICA with reference," Proceedings of International Joint Conference on Neural Networks, Montreal, Canada, July 31 - August 4, 2005, pp. 2525–2531.

[10] W. Lu and J. C. Rajapakse, "Approach and applications of constrained ICA," *IEEE Trans. Neural. Net.*, vol. 16, no. 1, pp. 203-212, Jan. 2005.

[11] C. J. James, and O. J. Gibson, " Temporally consstrained ICA: an application to artifact rejection in electromagnetic brain signal analysis," *IEEE Trans. on Biomed. Eng.*, vol. 50, no. 9, pp. 1108–1116, 2003.

[12] C. W. Hesse and C. J. James "The FastICA algorithm with spatial constraints," *IEEE Signal Process. Let.*, vol. 12, nO. 11, pp. 792–795, 2005.

[13] C. J. James and C. W. Hesse, "Semi-blind source separation in EM brain signal processing," Medical Applications of Signal Processing, 2005. The 3rd IEE International Seminar on (Ref. No. 2005-1119).

[14] Z. L. Zhang and Z. Yi, "Robust extraction of specific signals with temporal structure," Neurocomputing letters, vol. 69 pp. 888893, 2006.

[15] N. Ille, R. Beucker, and M. Scherg, "Spatially constrained independent component analysis for artifact correction in EEG and MEG," Neuroimage, vol. 13, p. S159, 2001.

[16] N. Ille, P. Berg, and M. Scherg, "Artifact correction of the ongoing EEG using spatial filters based on artifact and brain signal topographies," *J. Clin. Neurophysiol.*, vol. 19, no. 2, pp. 113124, 2002.

[17] C. J. James and C. W. Hesse "Independent component analysis for biomedical signals," *Physiol. Meas.*, vol. 26 (2005) R15R39.

[18] N. Mourad and J. Reilly, "Blind extraction of sparse sources", *ICASSP2010*, Dallas, Texas, USA, March 14-19, 2010.

[19] M. Zibulevsky and Y. Y. Zeevi, "Extraction of a source from multichannel data using sparse decomposition," Neurocomputing, vol. 49, pp. 163–173, 2002.

[20] R. Chartrand, "Exact reconstruction of sparse signals via nonconvex minimization," *IEEE Signal Process. Let.*, vol. 14, no. 10, Oct. 2007.

[21] N. Mourad and J. Reilly, "ℓ_p Minimization for Sparse Vector Reconstruction", *ICASSP2009*, Taipei, Taiwan, Apr. 19-24, 2009.

[22] N. Mourad and J. Reilly, "Minimizing Nonconvex Functions for Sparse Vector Reconstruction", *IEEE Trans. Signal. Proc.*, vol. 58, no. 7, pp. 3485–3496, 2010.

[23] N. Mourad, *Advances in sparse signal analysis with applications to blind source separation and EEG/MEG signal processing,* Ph.D. dissertation, McMaster University, Hamilton, On., Canada, 2009.

[24] M. Wax and T. Kailath, "Detection of signals by information theoretic criteria," *IEEE Trans. Acoust., Speech, Signal Proc.*, vol. ASSP-33, no. 2, pp. 387-392, 1985.

[25] W.G. Chen, K.M. Wong and J.P. Reilly, "Detection of the number of signals: A predicted eigen-threshold approach," *IEEE Trans. Signal Proc.*, May 1991, pp. 1088-1098.

[26] E. J. Candès, M. B. Wakin, and S. P. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," Journal of Fourier Analysis and Applications, vol. 14, no. 5, pp. 877-905, special issue on sparsity, December 2008.

[27] B. D. Rao and K. Kreutz-Delgado, "An affine scaling methodology for best basis selection," *IEEE Trans. Signal Process.*, vol. 47, no. 1, pp. 187–200, Jan. 1999.

[28] R. Chartrand and W. Yin, "Iteratively Reweighted Algorithms for Compressive Sensing," *ICASSP 2008*, Las Vegas, USA, pp. 3869-3872, March 31–April 4, 2008

[29] E. Oja and M. Plubley, "Blind separation of positive sources by globally convergent gradient search," Neural Computation, vol. 16, pp. 1811-1825, 2004.

[30] S. Fiori "Fully-mltiplicative orthogonal-group ICA neural algorithm", IEEE Letters, vol. 39. no. 24, 2003.

[31] B. N. Parlett, "The rayleigh quotient iteration and some generalizations for nonnormal matrices," *Math Comp.*, vol. 28, pp. 679–693, 1974.

[32] G. H. Golub and C. F. Van Loan, *Matrix computations,* 3rd ed. Baltimore, MD: Johns Hopkins Univ. Press, 1996.

[33] J. H. Manton, " Optimization algorithms exploiting unitary constraint", *IEEE Trans. Signal Proc.*, vol. 50, no. 3, pp. 635-650, 2002.

[34] P. C. Hansen, "Analysis of discrete ill-posed problems by means of the L-curve," *SIAM Rev.*, vol. 34, pp. 56180, Dec. 1992.

[35] K. Nazarpour, H. R. Mohseni, C. W. Hesse, J. A. Chambers, and S. Sanei, "A novel semiblind signal extraction approach for the removal of eye-blink artifact from EEGs," EURASIP Journal on Advances in Signal Processing, vol. 2008, Article ID 857459, 2008.

[36] Z. He, A. Cichocki, Y. Li, S. Xie, and S. Sanei, "K-hyperline clustering learning for sparse component analysis," *Signal Processing*, vol. 89, no. 6, pp. 1011-1022, 2009.

[37] L. Shoker, S. San ei, W. Wang, J. A. Chambers, "Removal of eye blinking artifact from the electro-encephalogram, incorporating a new constrained blind source separation algorithm," Medical & Biological Engineering & Computing, vol. 43, pp. 290-295, 2005.

[38] N. Mourad, J. Reilly, H. de Bruin, G. Hasey, and D. MacCrimmond, "A simple and fast algorithm for automatic suppression of high-amplitude artifacts in EEG data," *ICASSP2007*, Honolulu, Hawaii, USA, April 15-20, 2007.