# A HOMOTOPY RECURSIVE-IN-MODEL-ORDER ALGORITHM FOR WEIGHTED LASSO

*Zbyněk Koldovský*[1,2] *and Petr Tichavský*[2]

[1]Faculty of Mechatronic and Interdisciplinary Studies
Technical University of Liberec, Studentská 2, 461 17 Liberec, Czech Republic
[2]Institute of Information Theory and Automation,
P.O.Box 18, 182 08 Prague 8, Czech Republic

## ABSTRACT

A fast algorithm to solve weighted $\ell_1$-minimization problems with $N \times N$ square "measuring" matrices is proposed. The method is recursive-in-model-order and tracks a homotopy path that goes through solutions of the optimization sub-tasks in the order of 1 through $N$. It thus yields solutions for all model orders and performs this task faster than the other compared methods. We show applications of this method in sparse linear system identification, in particular, the estimation of sparse target-cancellation filters for audio source separation.

***Index Terms*—** Sparse Linear Regression, Homotopy, $\ell_1$ norm, System Identification, Levinson-Durbin Algorithm

## 1. INTRODUCTION

Recently, optimization problems leading to sparse solutions have opened new directions in signal processing fields, e.g., in linear regression models, compressive sensing, channel identification and equalization, and so forth. Classical quadratic criteria are modified by adding regularization terms or constraints. The typical term is the $\ell_1$ norm or the weighted $\ell_1$ norm or, most recently, also mixed norms such as $\ell_{1,2}$ or $\ell_{1,\infty}$ norm. These norms induce sparsity of the solution and ensure that the implied optimization problem remains convex and could be resolved in polynomial time.

A popular convex optimization program is the (weighted) Lasso given by [1, 2]

$$\min_{\mathbf{x}\in\mathcal{R}^N} \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{y}\|_2^2 + \|\mathbf{W}\mathbf{x}\|_1 \qquad (1)$$

where $\mathbf{A} = (a_{ij})$ is an $M \times N$ "measuring" matrix, $\mathbf{y} = [y_1, \ldots, y_M]^T$, and $\mathbf{W} = \mathtt{diag}(w_1, \ldots, w_N)$ is a diagonal weighting matrix with positive weights $w_1, \ldots, w_N$ on its diagonal. An equivalent problem, in the sense that the sets of solutions are the same for all the possible choices of parameters $w_1, \ldots, w_N$ and $\sigma$, is [3]

$$\min_{\mathbf{x}\in\mathcal{R}^N} \|\mathbf{W}\mathbf{x}\|_1 \quad \text{w.r.t.} \quad \|\mathbf{A}\mathbf{x}-\mathbf{y}\|_2^2 \le \sigma. \qquad (2)$$

[0]This work was supported by the Czech Science Foundation through Project No. 14-11898S.

Motivated by channel identification problems, we address (1) where $\mathbf{A}$ is square ($M = N$) and, for simplicity, symmetric (note that non-symmetric $\mathbf{A}$ could also be considered).

Numerous methods have been proposed to solve Lasso and related programs; see, for instance, [4, 5]. Homotopy algorithms [6, 7, 8] solve Lasso by tracing the entire solution path for a range of decreasing values of parameters $(w_1, \ldots, w_N)$, starting from the zero solution. They take advantage of the fact that the path is piecewise linear. These approaches are handiest in cases where the solution is known for certain values of parameters that are "close" to parameters of the problem we need to solve. For example, they could be used in adaptive algorithms where the solution must be sequentially updated according to new data [9, 10, 11, 12].

In this paper, we derive a novel homotopy algorithm to find solutions of all reduced-order programs

$$\min_{\mathbf{x}\in\mathcal{R}^n} \frac{1}{2}\|\mathbf{A}^n\mathbf{x}-\mathbf{y}^n\|_2^2 + \|\mathbf{W}^n\mathbf{x}\|_1, \qquad (3)$$

that is, for $n = 1, \ldots, N$, where $\mathbf{A}^n$ denotes the $n \times n$ upper-left corner submatrix of $\mathbf{A}$, $\mathbf{y}^n = [y_1, \ldots, y_n]^T$, and $\mathbf{W}^n = \mathtt{diag}(w_1, \ldots, w_n)$. The algorithm tracks the homotopy path that goes through the solutions of (3). It might be viewed as an analogy to the famous Levinson-Durbin recursive algorithm [13], designed to solve the purely quadratic case ($\mathbf{W} = \mathbf{0}$) where $\mathbf{A}$ is a square symmetric Toeplitz matrix, and the problem falls back on finding solutions of $\mathbf{A}^n\mathbf{x} = \mathbf{y}^n$, $n = 1, \ldots, N$.

## 2. HOMOTOPY RECURSION

The solution of (3), from now on denoted by $\mathbf{x}^n$, satisfies the following optimality conditions [8]:

$$(\mathbf{a}_i^n)^T(\mathbf{A}^n\mathbf{x}^n - \mathbf{y}^n) = -w_i z_i, \qquad i \in \Gamma_n, \qquad (4)$$

$$|(\mathbf{a}_i^n)^T(\mathbf{A}^n\mathbf{x}^n - \mathbf{y}^n)| < w_i, \qquad i \in \Gamma_n^c, \qquad (5)$$

where $\mathbf{a}_i^n$ is the $i$th column of $\mathbf{A}^n$, $z_i$ is the sign of $(\mathbf{x}^n)_i$, $\Gamma_n$ is the set of indices of nonzero elements of $\mathbf{x}^n$ (the active set), and $\Gamma_n^c$ its complement to $\{1, \ldots, n\}$.

The algorithm begins by finding the solution for $n = 1$. Then, assuming that $\mathbf{x}^{n-1}$ is known, $\mathbf{x}^n$ is sought by considering a parameterized extended program

$$\min_{\mathbf{x}\in\mathcal{R}^n} \frac{1}{2}\|\mathbf{A}^n\mathbf{x} - \mathbf{y}^n(\omega)\|_2^2 + \|\mathbf{W}^n(\lambda)\mathbf{x}\|_1 \tag{6}$$

where

$$\mathbf{A}^n = \begin{pmatrix} \mathbf{A}^{n-1} & \mathbf{a} \\ \mathbf{a}^T & a_{nn} \end{pmatrix}, \quad \mathbf{y}^n(\omega) = \begin{pmatrix} \mathbf{y}^{n-1} \\ \omega \end{pmatrix} \tag{7}$$

$\mathbf{W}^n(\lambda) = \mathtt{diag}(w_1, \ldots, w_{n-1}, \lambda)$, and $\mathbf{a} = (a_{1n}, \ldots, a_{n-1,n})^T$.

Using (4) and (5), it can be shown that $[(\mathbf{x}^{n-1})^T\ 0]^T$ is the solution of (6) when $\omega = \mathbf{a}^T\mathbf{x}^{n-1}$ and $\lambda$ is sufficiently large. At this stage, we do not need to know the exact value of $\lambda$; its existence follows from (5). The proposed algorithm goes through two homotopy paths to compute $\mathbf{x}^n$ that corresponds to the solution of (6) for $\omega = y_n$ and $\lambda = w_n$.

### 2.1. Homotopy path 1

The first homotopy path tracks the solutions when $\omega$ changes linearly from $\mathbf{a}^T\mathbf{x}^{n-1}$ to $y_n$ while $\lambda$ remains fixed. We define $\omega$ dependent on $\epsilon$ as

$$\omega_\epsilon = (1-\epsilon)\mathbf{a}^T\mathbf{x}^{n-1} + \epsilon y_n \tag{8}$$

where $\epsilon \in [0, 1]$. Let $\mathbf{x}^*(\epsilon)$ be the solution of (6) for $\omega = \omega_\epsilon$, and $\Gamma_\epsilon$ denote its active set. We note that $n \in \Gamma_\epsilon^c$ for every $\epsilon$. This path begins with $\epsilon = 0$ and starts from $\mathbf{x}^*(0) = [(\mathbf{x}^{n-1})^T\ 0]^T$.

In one homotopy step, we increase $\epsilon$ by $\delta \in [0, 1-\epsilon]$ and change $\mathbf{x}^*(\epsilon)$ by $\delta\mathbf{\Delta x}$ as long as the optimality conditions are satisfied. From (4) it follows that $\mathbf{\Delta x}$ must satisfy

$$(\mathbf{a}_i^n)^T\left(\mathbf{A}^n(\mathbf{x}^*(\epsilon) + \delta\mathbf{\Delta x}) - \mathbf{y}^n(\omega_{\epsilon+\delta})\right) = -w_i z_i \tag{9}$$

for $i \in \Gamma_\epsilon$. Using the fact that (9) holds for $\delta = 0$, after some simplifications we arrive at

$$(\mathbf{\Delta x})_{\Gamma_\epsilon} = -(\omega_0 - \omega_1)\left[(\mathbf{A}^n_{\Gamma_\epsilon})^T\mathbf{A}^n_{\Gamma_\epsilon}\right]^{-1}\mathbf{a}_{\Gamma_\epsilon} \tag{10}$$

where the subscript $(\cdot)_{\Gamma_\epsilon}$ denotes the restriction to indices (columns in case of a matrix) in the set $\Gamma_\epsilon$.

The solution $\mathbf{x}^*(\epsilon)$ can be changed by $\delta\mathbf{\Delta x}$ until at least one of conditions (4) or (5) is violated, which corresponds to changes in $\Gamma_\epsilon$. The condition (9) is violated when the $i$th element of $\mathbf{x}^*(\epsilon)+\delta\mathbf{\Delta x}$ shrinks to zero for some $i \in \Gamma_\epsilon$. This means that the element leaves the active set. By contrast, an element from $\Gamma_\epsilon^c$ enters the active set when the corresponding inequality in (5) turns to equality.

The maximum step without changes in the active set is thus equal to $\delta^* = \min\{\delta^+, \delta^-, 1-\epsilon\}$, where

$$\delta^+ = \min_{i\in\Gamma_\epsilon^c}\left(\frac{w_i - b_i}{c_i}, -\frac{w_i + b_i}{c_i}\right)_+ \tag{11}$$

$$\delta^- = \min_{i\in\Gamma_\epsilon}\left(\frac{-(\mathbf{x}^*(\epsilon))_i}{(\mathbf{\Delta x})_i}\right)_+ \tag{12}$$

where $\min(\cdot)_+$ means that the minimum is taken over only positive values, and $b_i$ and $c_i$ denote, respectively, the $i$th element of vectors

$$\mathbf{b} = (\mathbf{A}^n)^T(\mathbf{A}^n\mathbf{x}^*(\epsilon) - \mathbf{y}^n(\omega_\epsilon)) \tag{13}$$

$$\mathbf{c} = (\mathbf{A}^n)^T\left(\mathbf{A}^n\mathbf{\Delta x} - \begin{bmatrix} \mathbf{0} \\ \mathbf{a}^T_{\Gamma_\epsilon}\mathbf{\Delta x}_{\Gamma_\epsilon} + \omega_0 - \omega_1 \end{bmatrix}\right). \tag{14}$$

Now we arrive at a new solution $\mathbf{x}^*(\epsilon+\delta^*) = \mathbf{x}^*(\epsilon)+\delta^*\mathbf{\Delta x}$. The first homotopy path reaches its end if $\delta^* = 1-\epsilon$. Otherwise, the active set $\Gamma_{\epsilon+\delta^*}$ is obtained by an element adding to or removing it from $\Gamma_\epsilon$ as described above, and the homotopy step is repeated starting with $\epsilon \leftarrow \epsilon + \delta^*$. We denote the final solution $\mathbf{x}^*(1)$ by $\tilde{\mathbf{x}}^n$.

### 2.2. Homotopy path 2

Now we can verify whether $\tilde{\mathbf{x}}^n$ is already the desired solution $\mathbf{x}^n$. Since the last element of $\tilde{\mathbf{x}}^n$ is not contained in the active set ($\tilde{\mathbf{x}}^n_n = 0$), it satisfies the condition following from (5)

$$|(\mathbf{a}_n^n)^T\left(\mathbf{A}^n\tilde{\mathbf{x}}^n - \mathbf{y}^n\right)| < \lambda. \tag{15}$$

If this condition is satisfied for $\lambda = w_n$, $\tilde{\mathbf{x}}^n$ is already equal to $\mathbf{x}^n$. Otherwise, we have to start the second homotopy path that tracks the solutions when $\lambda$ changes to $w_n$.

The path begins at the point where condition (15) becomes violated, that is,

$$\lambda^{\mathrm{ini}} = |(\mathbf{a}_n^n)^T\left(\mathbf{A}^n\tilde{\mathbf{x}}^n - \mathbf{y}^n\right)|, \tag{16}$$

and the index $n$ enters the active set. We again parameterize the path via $\epsilon \in [0, 1]$. The weight changes according to

$$\lambda_\epsilon = (1-\epsilon)\lambda^{\mathrm{ini}} + \epsilon w_n. \tag{17}$$

The initial optimum for $\epsilon = 0$ is $\mathbf{x}^*(0) = \tilde{\mathbf{x}}^n$.

Further derivations are analogous to those carried out in the previous subsection. The homotopy update is

$$(\mathbf{\Delta x})_{\Gamma_\epsilon} = -\kappa(\lambda_0 - \lambda_1)\left[(\mathbf{A}^n_{\Gamma_\epsilon})^T\mathbf{A}^n_{\Gamma_\epsilon}\right]^{-1}\mathbf{e} \tag{18}$$

where $\kappa = \mathtt{sign}[(\mathbf{a}_n^n)^T\left(\mathbf{A}^n\tilde{\mathbf{x}}^n - \mathbf{y}^n\right)]$, that is the sign of left-hand side in (16), and $\mathbf{e} = (0, \ldots, 0, 1)^T$ of size $|\Gamma_\epsilon| \times 1$. The optimum is updated as $\mathbf{x}^*(\epsilon+\delta^*) = \mathbf{x}^*(\epsilon)+\delta^*\mathbf{\Delta x}$ where $\delta^* = \min\{\delta^+, \delta^-, 1-\epsilon\}$. $\delta^+$ and $\delta^-$ are defined as in (11) and (12), respectively, but instead of (13) and (14) the vectors $\mathbf{b}$ and $\mathbf{c}$ are defined as

$$\mathbf{b} = (\mathbf{A}^n)^T(\mathbf{A}^n\mathbf{x}^*(\epsilon) - \mathbf{y}^n), \tag{19}$$

$$\mathbf{c} = (\mathbf{A}^n)^T\mathbf{A}^n\mathbf{\Delta x}. \tag{20}$$

### 2.3. Solution for $n = 1$

There are three potential candidates for being the solution $\mathbf{x}^1$. Namely, $(a_{11}y_1 + w_1)/a_{11}^2$, $(a_{11}y_1 - w_1)/a_{11}^2$ and 0. The minimum is established by evaluating the objective function in (1) for these three values. The proposed algorithm is summarized in Algorithm 1.

**Algorithm 1:** Pseudocode of the proposed algorithm

**Input**: $\mathbf{A}, \mathbf{y}, \mathbf{W}$
**Output**: $\mathbf{x}^1, \ldots, \mathbf{x}^N$
$n = 1$;
Compute $\mathbf{x}^1$ as described in Subsection 2.3;
**for** $n = 2, \ldots, N$ **do**
  $\mathbf{a} = [a_{n1}, \ldots, a_{n,n-1}]^T$, $\epsilon = 0$, $\mathbf{x}^*(0) = [(\mathbf{x}^{n-1})^T \; 0]^T$,
  $\Gamma_\epsilon = \Gamma_{n-1}$;
  **while** $\epsilon < 1$ **do** `/* homotopy path 1 */`
    Compute $\mathbf{\Delta x}$ according to (10) and $\delta^*$ using (11)-(14);
    $\mathbf{x}^*(\epsilon + \delta^*) = \mathbf{x}^*(\epsilon) + \delta^* \mathbf{\Delta x}$, update $\Gamma_{\epsilon+\delta^*}$;
    $\epsilon = \epsilon + \delta^*$;
  **end**
  $\tilde{\mathbf{x}}^n = \mathbf{x}^*(1)$;
  **if** $|(\mathbf{a}_n^n)^T (\mathbf{A}^n \tilde{\mathbf{x}}^n - \mathbf{y}^n)| < w_n$ **then**
    $\mathbf{x}^n = \tilde{\mathbf{x}}^n$;
  **else**
    $\epsilon = 0$, $\mathbf{x}^*(0) = \tilde{\mathbf{x}}^n$, add $n$ to the active set $\Gamma_\epsilon$;
    **while** $\epsilon < 1$ **do** `/* homotopy path 2 */`
      Compute $\mathbf{\Delta x}$ according to (18) and $\delta^*$ using (11), (12), (19), and (20);
      $\mathbf{x}^*(\epsilon + \delta^*) = \mathbf{x}^*(\epsilon) + \delta^* \mathbf{\Delta x}$, update $\Gamma_{\epsilon+\delta^*}$;
      $\epsilon = \epsilon + \delta^*$;
    **end**
    $\mathbf{x}^n = \mathbf{x}^*(1)$;
  **end**
**end**

## 3. EXPERIMENTS

### 3.1. Sparse channel recovery

In this experiment, we consider the estimation of a sparse relative impulse response between two sensors. Signals observed by the sensors will be denoted by $u$ and $v$. Their relationship is such that $v = g * u + f$, where $g$ is the impulse response to be estimated, $*$ denotes the convolution, and $f$ is a noise signal. Let the length of $g$ be $L$, and the number of its nonzero coefficients be $S$. The coefficients are randomly (uniformly) distributed within $g$, and their values are generated from $\mathcal{N}(0,1)$. $Q$ samples $u_1, \ldots, u_Q$ and $f_1, \ldots, f_Q$ of $u$ and $f$ are, respectively, generated from $\mathcal{N}(0,1)$ and $\mathcal{N}(0, \sigma_f^2)$.

The minimum squared error (MSE) estimate of $g$, denoted by $\mathbf{h}^{L'}$, is defined as the minimizer of $\|\mathbf{U}\mathbf{h} - \mathbf{v}\|_2^2$, where $L'$ is the length of $\mathbf{h}$, $\mathbf{U}$ is the $(Q + L' - 1) \times L'$ Toeplitz matrix whose first row and first column are $[u_1, 0, \ldots, 0]$ and $[u_1, \ldots, u_Q, 0, \ldots, 0]^T$, respectively, and $\mathbf{v}$ is the vector of length $Q + L' - 1$ containing the samples of $v$. Then, $\mathbf{h}^{L'}$ is the solution of $\mathbf{R}\mathbf{h} = \mathbf{p}$ and can be interpreted as the solution of

$$\min_{\mathbf{h} \in \mathcal{R}^{L'}} \|\mathbf{R}\mathbf{h} - \mathbf{p}\|_2^2 \tag{21}$$

where $\mathbf{R} = \mathbf{U}^T\mathbf{U}/Q$ and $\mathbf{p} = \mathbf{U}^T\mathbf{y}/Q$. Note that $\mathbf{R}$ is an $L' \times L'$ symmetric Toeplitz matrix, so $\mathbf{h}^{L'}$ can be computed using the Levinson-Durbin algorithm for $L' = 1, 2, \ldots$

Now we consider sparse estimators of $g$ obtained through solving (1) and (2) where $\mathbf{A} = \mathbf{R}$ and $\mathbf{y} = \mathbf{p}$. To compare, the solutions of (1) are computed using the proposed algorithm, from this point forward denoted as ALG, and the homotopy approach HOM[1] from [8], and (2) is solved using the root-finding algorithm SPGL1[2] from [5].

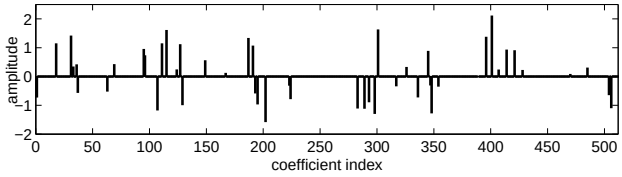


**Fig. 1**. The random sparse impulse response.

Two weighting matrices $\mathbf{W}_1$ and $\mathbf{W}_2$ are considered. The weights in $\mathbf{W}_1$ are all equal to $\tau$. The same holds for $\mathbf{W}_2$ but those weights whose indices correspond to nonzero elements in $g$ are equal to $\tau/100$. $\mathbf{W}_2$ thus incorporates our a priori knowledge of the support of $g$. The compared methods using $\mathbf{W}_2$ will be denoted as ALGw, HOMw, and SPGL1w. In SPGL1, we use $\tau = 1$ and $\sigma = 0.075\sqrt{L'}$.

Fig. 2 shows results of one run of the experiment where $N = 1000$, $L = 512$, $S = 50$, $\tau = 0.2$, $\sigma_f^2$ is selected so that the signal-to-noise ratio in $v$ is 10 dB, and $L' = 1, \ldots, L$. The generated $g$ is shown in Fig. 1. Two signal-to-error ratio (SER) criteria depending on $L'$ are used for the evaluation:

$$\text{SER}_{L'} = \frac{\|\mathbf{g}^{L'}\|_2}{\|\mathbf{g}^{L'} - \mathbf{x}^{L'}\|_2} \quad \text{and} \quad \text{SER} = \frac{\|\mathbf{g}^{L}\|_2}{\|\mathbf{g}^{L} - \mathbf{x}^{L',0}\|_2},$$

where $\mathbf{g}^n = [g_1, \ldots, g_n]^T$, $\mathbf{x}^n$ is the solution for $L' = n$, and $\mathbf{x}^{L',0}$ is $\mathbf{x}^{L'}$ padded with zeros to a total length of $L$ elements. The value of SER expresses how closely $\mathbf{x}^{L'}$ approaches $\mathbf{g}^L$. Similarly, $\text{SER}_{L'}$ evaluates the proximity of $\mathbf{x}^{L'}$ to the truncated version of $\mathbf{g}^L$, i.e., $\mathbf{g}^{L'}$.

Both SER and $\text{SER}_{L'}$ in Fig. 2 show that the sparse estimators yield better estimates than MSE. The weighting $\mathbf{W}_2$ gives better results than $\mathbf{W}_1$ both by ALGw and SPGL1w, because the weighting tells the estimator which elements should be nonzero. The solutions reached by SPGL1 and SPGL1w are similar to those of ALG and ALGw, respectively, in terms of SER but slightly different in terms of $\text{SER}_{L'}$. (We select $\sigma$ such that the methods yield approximately the same results for $L' = L$.)

To compare the computational complexity of these methods, Table 1 shows the average number of iterations (homotopy steps in the cases of ALG and HOM) and an average computational time in Matlab to compute the estimates $\mathbf{x}^{L'}$, for all $L' = 1, \ldots, L$, when $S = 20$, 50, 100, and 200. The results are averaged over 100 independent runs for each $S$. We use up-to-date Matlab implementations of the compared methods on a PC with an i7 2.7GHz processor. The implementations of ALG and HOM use the matrix inversion lemma to speed up the computations of the homotopy step direction

[1] `http://users.ece.gatech.edu/~sasif/homotopy/`
[2] `http://www.cs.ubc.ca/~mpf/spgl1`

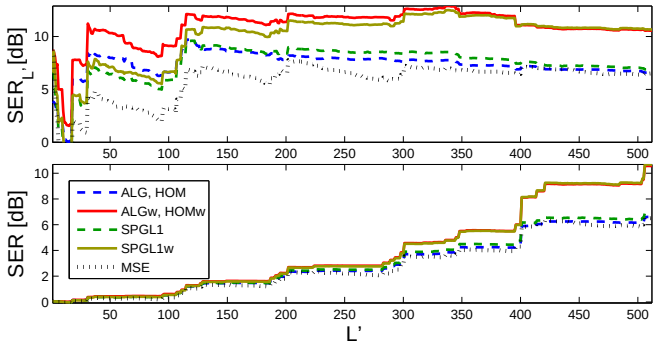


**Fig. 2**. Comparison of SER and $\text{SER}_{L'}$ for the recovery of the sparse impulse response in Fig. 1.

| | Number of nonzero coefficients in $g$ | | | |
|---|---|---|---|---|
| | $S$=20 | $S$=50 | $S$=100 | $S$=200 |
| **ALG** | 1058 / 1.4 | 1927 / 2.1 | 2748 / 3.2 | 3580 / 5.2 |
| **ALGw** | 1059 / 1.4 | 2036 / 2.1 | 3029 / 3.5 | 4165 / 6.5 |
| **HOM** | 12749 / 6.7 | 31347 / 17.3 | 48776 / 30.4 | 69342 / 48.2 |
| **HOMw** | 12211 / 6.3 | 30661 / 16.5 | 50300 / 31.2 | 77682 / 56.2 |
| **SPGL1** | 6092 / 6.3 | 11222 / 8.5 | 15100 / 10.4 | 21164 / 13.3 |
| **SPGL1w** | 8182 / 7.2 | 22370 / 13.5 | 35730 / 19.9 | 51134 / 27.4 |

**Table 1**. Average number of iterations vs. average computational time [secs] to compute $\mathbf{x}^{L'}$, for all $L' = 1, \ldots, L$.

vector $\mathbf{\Delta x}$ [8]. To compute $\mathbf{x}^{L'}$ by SPGL1, the algorithm is initialized with $[\mathbf{x}^{L'-1}; 0]$, which is faster than starting the method from scratch.

The proposed method requires significantly shorter computational time as well as fewer iterations than the compared methods. The computational time grows with $S$. With the weighting matrix $\mathbf{W}_2$, the complexity of ALG and SPGL1 is slightly higher, because the solutions for $\mathbf{W}_2$ are "less sparse". On the other hand, HOM is faster with $\mathbf{W}_2$.

### 3.2. Audio Source Separation Application: Sparse Target-Cancellation Filter Estimation

In audio source separation, an important tool is the target-cancellation filter (CF), which is a multi-channel filter that cancels the target signal but lets other signals pass. Its output provides a noise reference signal, which is useful in various methods and beamformers to extract the target signal from its noisy recording; see, e.g., [14, 15, 16].

For a static source, the CF could be computed from a noise-free recording of the target. We consider such left and right channel recordings, respectively, modeled as $x_{\text{L}} = s + y_{\text{L}}$ and $x_{\text{R}} = g * s + y_{\text{R}}$, where $s$ is the target signal observed on the left microphone, $y_{\text{L}}$ and $y_{\text{R}}$ are noise signals (noise+interferences), and $g$ is the relative impulse response between the channels. $g$ could be estimated from the noise-free recording, and then the CF could be defined as such that its output is $h * x_{\text{L}} - x_{\text{R}}$ where $h$ is the estimate of $g$ [17, 18].

The MSE estimation of $g$ leads to the task (21) where $u = x_{\text{L}}$ and $v = x_{\text{R}}$. Although $g$ is not sparse in general, its sparse estimates are also worth considering, for instance, to denoise the estimate and/or to reduce the number of parameters for easier interpretability of the estimate; see, e.g., [19, 20].

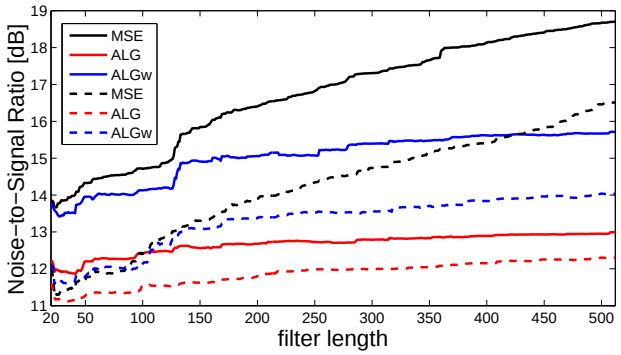


**Fig. 3**. Cancellation performance of CFs in terms of the NSR (the higher the NSR, the better the cancellation) for a speaker that is 90 cm (solid lines) and 175 cm (dashed lines) distant from microphones. Better NSRs are achieved for the speaker distance of 90 cm than for 175 cm, because the effective length of $g$ typically grows with the distance.

We compare CFs computed by MSE, ALG ($\tau = 0.05$) and ALGw for a target speaker that is 90 and 175 cm distant, respectively, from two microphones; the real-world recordings are taken from [21]. In ALGw, the coefficients weighted by $\tau/100$ are selected based on the MSE estimate of $g$: their absolute values in the MSE estimate are higher than $0.05$.

Fig. 3 shows the Noise-to-Signal Ratio (NSR) in the output of the CFs when an interfering speaker is present at approximately the same distance. MSE achieves, naturally, the best NSR; nevertheless, the NSRs values achieved by ALGw and ALG are close to that of MSE while the number of nonzero coefficients in resulting filters is $\leq 100$. Viewing the dependency of the NSR on the filter length enables us to find the trade-off between the filter length, the number of nonzero coefficients and the cancellation performance. Examples of resulting filters are shown in Fig. 4.

## 4. CONCLUSIONS

The proposed algorithm is a fast method to compute solutions of the weighted $\ell_1$ optimization problem for all model orders $1, \ldots, N$.

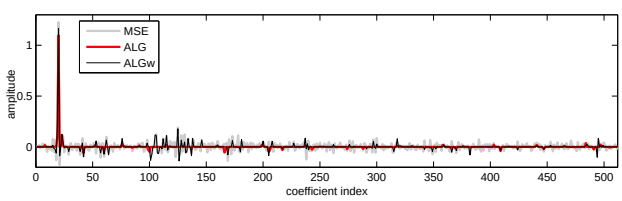


**Fig. 4**. Estimated $g$ for $L' = 512$ for the speaker distance 175 cm.

## 5. REFERENCES


[1] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Royal. Statist. Soc. B.,* vol. 58, pp. 267–288, 1996.

[2] E. J. Candès , M. B. Wakin , S. P. Boyd, "Enhancing sparsity by reweighted $\ell_1$ minimization," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5, pp. 877–905, 2008.

[3] D. L. Donoho , Y. Tsaig, "Fast solution of l1-norm minimization problems when the solution may be sparse," *IEEE Trans. Inf. Theory*, vol. 54, no. 11, pp. 4789–4812, Nov. 2008.

[4] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.

[5] E. van den Berg and M. P. Friedlander, "Probing the Pareto frontier for basis pursuit solutions," *SIAM J. on Scientific Computing*, vol. 31, no. 2, pp.890–912, Nov. 2008.

[6] M. R. Osborne, B. Presnell, and B.A. Turlach, "A new approach to variable selection in least squares problems," *IMA J. Numer. Anal.*, vol. 20, no. 3, pp. 389–403, 2000.

[7] B. Efron, T. Hastie, I. Johnstone, R. Tibshirani, "Least angle regression, " *Annals of Statistics*, vol. 32, no. 2, pp. 407–499, 2004.

[8] M. S. Asif and J. Romberg, "Fast and accurate algorithms for re-weighted L1-norm minimization," *submitted to IEEE Trans. on Signal Process.*, arXiv:1208.0651, July 2012.

[9] M. S. Asif and J. Romberg, "Dynamic updating for $\ell_1$ minimization," *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 2, pp. 421–434, 2010.

[10] P. J. Garrigues and E. L. Ghaoui, "An homotopy algorithm for the Lasso with online observations," *Neural Inf. Process. Syst. (NIPS)*, vol. 21, 2008.

[11] D. M. Malioutov, R. S. Sujay, and A. S. Willsky, "Sequential compressed sensing," *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 2, pp. 435–444, April 2010.

[12] Y. Chen and A. O. Hero III, "Recursive $\ell_{1,\infty}$ Group Lasso," *IEEE Trans. Signal Process.*, vol. 60, No. 8, Aug. 2012.

[13] N. Levinson, "The Wiener RMS error criterion in filter design and prediction," *J. Math. Phys.*, vol. 25, pp. 261–278, 1947.

[14] S. Gannot, D. Burshtein, and E. Weinstein, "Signal enhancement using beamforming and nonstationarity with applications to speech," *IEEE Trans. Signal Process.*, vol. 49, no. 8, pp. 1614–1626, Aug. 2001.

[15] A. Krueger, E. Warsitz, and R. Haeb-Umbach, "Speech enhancement with a GSC-like structure employing eigenvector-based transfer function ratios estimation," *IEEE Audio, Speech, Language Process.*, vol. 19, no. 1, Jan. 2011.

[16] S. Doclo and M. Moonen, "GSVD-based optimal filtering for single and multimicrophone speech enhancement," *IEEE Trans. Signal Process.*, vol. 50, no. 9, pp. 2230–2244, Sep. 2002.

[17] Y. Lin, J. Chen, Y. Kim and D. Lee, "Blind channel identification for speech dereverberation using $\ell_1$ norm sparse learning," *Advances in Neural Information Processing Systems 20, Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems*, MIT Press, Vancouver, British Columbia, Canada, December 3-6, 2007.

[18] Z. Koldovský, P. Tichavský, D. Botka, "Noise Reduction in Dual-Microphone Mobile Phones Using A Bank of Pre-Measured Target-Cancellation Filters," *Proc. of ICASSP 2013*, pp. 679–683, Vancouver, Canada, May 2013.

[19] K. Kowalczyk, E. A. P. Habets, W. Kellermann, P. A. Naylor, "Blind System Identification Using Sparse Learning for TDOA Estimation of Room Reflections," *IEEE Signal Processing Letters*, vol. 20, no. 7, pp. 653–656, July 2013.

[20] J. Málek and Z. Koldovský, "Sparse Target Cancellation Filters with Application to Semi-Blind Noise Extraction," *Proc. of ICASSP 2014 (this conference)*, Florence, Italy, May 2014.

[21] Dataset of the SiSEC 2010 evaluation campaign [online], `http://sisec2010.wiki.irisa.fr/tiki-index.php?page=Robust+blind+linear+separation+of+short+two-sources-two-microphones+recordings`