

Research Article

Efficient LED-SAC Sparse Estimator Using Fast Sequential Adaptive Coordinate-Wise Optimization (LED-2SAC)

T. Yousefi Rezaii,¹ S. Beheshti,² and M. A. Tinati¹

¹ Faculty of Electrical and Computer Engineering, University of Tabriz, 29 Bahman Avenue, Tabriz 51666-15813, Iran

² Department of Electrical and Computer Engineering, Ryerson University, Toronto, ON, Canada M5B 2K3

Correspondence should be addressed to T. Yousefi Rezaii; tohidusefi@gmail.com

Received 7 October 2013; Accepted 29 December 2013; Published 6 February 2014

Academic Editor: Yue Wu

Copyright © 2014 T. Yousefi Rezaii et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Solving the underdetermined system of linear equations is of great interest in signal processing application, particularly when the underlying signal to be estimated is sparse. Recently, a new sparsity encouraging penalty function is introduced as Linearized Exponentially Decaying penalty, LED, which results in the sparsest solution for an underdetermined system of equations subject to the minimization of the least squares loss function. A sequential solution is available for LED-based objective function, which is denoted by LED-SAC algorithm. This solution, which aims to sequentially solve the LED-based objective function, ignores the sparsity of the solution. In this paper, we present a new sparse solution. The new method benefits from the sparsity of the signal both in the optimization criterion (LED) and its solution path, denoted by Sparse SAC (2SAC). The new reconstruction method denoted by LED-2SAC (LED-Sparse SAC) is consequently more efficient and considerably fast compared to the LED-SAC algorithm, in terms of adaptability and convergence rate. In addition, the computational complexity of both LED-SAC and LED-2SAC is shown to be of order $\mathcal{O}(d^2)$, which is better than the other batch solutions like LARS. LARS algorithm has complexity of order $\mathcal{O}(d^3 + nd^2)$, where d is the dimension of the sparse signal and n is the number of observations.

1. Introduction

Compressed sensing (CS) signal processing has gained a lot of popularity due to the lower sampling rate needed for reconstruction of the original signal compared to that of Nyquist lower bound of sampling rate. Consequently, the signal is sampled in a compressed manner via the sampler and there is no need for an additional compression procedure (as it is common in conventional signal processing).

The main challenge in CS signal processing is how to recover the original signal using the few samples from the sampler. Suppose the sparse signal, $\theta^* \in \mathbb{R}^d$, $\|\theta^*\|_0 = K \ll d$, is to be estimated via the following linear regression model:

$$y_i = \mathbf{x}_i^T \theta^* + v_i \quad i = 1, 2, \dots, n, \quad (1)$$

where y_i , $\mathbf{x}_i \in \mathbb{R}^d$, and v_i are the observation, regressor and observation noise at time index i , respectively. v_i is assumed to be additive white Gaussian noise with mean 0 and variance

σ^2 . In the matrix form, the linear regression model (1) will have the following form:

$$\mathbf{y} = \mathbf{X}\theta^* + \mathbf{v}, \quad (2)$$

where $\mathbf{y} = [y_1, \dots, y_n]^T$, $\mathbf{X} = [\mathbf{x}_1^T, \dots, \mathbf{x}_n^T]^T$, and $\mathbf{v} = [v_1, \dots, v_n]^T$ are the observation vector, regression matrix (also called measurement matrix in CS), and noise vector, respectively.

Variable selection and high prediction accuracy are two major issues in sparse signal estimation. Variable selection is necessary for sparsity aware signal estimation. The common approach is to introduce a penalty to the overall objective function which will guarantee sparsity in the estimated signal. In order to ensure high prediction accuracy, a suitable loss function must be introduced. The most common loss function is the L_2 loss function due to its convexity which results in the well-known least squares solution. A general form of the objective functions to be minimized in order to

recover the sparse solution, which is also accepted through this paper, is as follows:

$$\mathcal{J}(\boldsymbol{\theta}^*) = \frac{1}{2n} \|\mathbf{y} - \mathbf{X}\boldsymbol{\theta}^*\|_2^2 + \tau \mathcal{P}(|\boldsymbol{\theta}^*|), \quad (3)$$

where $\mathcal{P}(\cdot)$ is the penalty function and τ is a tuning parameter that balances the prediction accuracy and sparsity.

Depending on the choice of the loss and sparsity encouraging functions and the techniques used for solving the optimization problem, extensive works have been presented in the literature. Some sophisticated and relatively high precision approaches are basis pursuit [1] and greedy [2, 3] reconstruction algorithms which perform batch-based estimation. The other class of batch-based sparse estimation algorithms is based on the L_p norm ($p < 1$) as the sparsity encouraging factor [4–8]. As it is reported, using the L_p norm in the objective function will effectively reduce the number of required measurements compared to that of the L_1 norm reconstruction. However, there is no analytic guarantee.

Adaptive signal reconstruction is of interest in applications where the sparse signal of interest undergoes variations in its support as well as the magnitude of its nonzero entries. Furthermore, in most signal acquisition devices, the observations are obtained sequentially. Thus, the variations in the support of the unknown signal can be sensed by sequentially processing the observations rather than batch processing.

In [9], observations are received in sequence and without any prior assumptions on the signal sparsity; the reconstruction error is computed between observations in order to decide whether enough samples have been obtained. Variational adaptive filters are extensively used to sparse signal reconstruction as in [10–18]. The resulting estimators are sequential but too slow. Furthermore, due to the lack of a direct variable selection stage, like a thresholding rule, exact reconstruction of the zero parameters is impossible.

Recently, the family of Least-Absolute Shrinkage and Selection Operator (Lasso) objective functions has gained a lot of popularity in sparse signal reconstruction context [19]. Lasso in its standard form includes the L_1 penalized least squares error criterion, which continuously shrinks the parameters toward zero. The Least Angle Regression (LARS) is the most famous batch-based algorithm to solve the Lasso problem [20] (a similar solution is also already proposed by Osborne et al. [21]).

It is shown that the standard Lasso objective function leads to biased estimator, which is due to the pure soft thresholding stage used in the estimation procedure [22–25]. SCAD (Smoothly Clipped Absolute Deviation) and adaptive Lasso are two alternatives which are presented in [22, 24], respectively, in order to make the standard Lasso an unbiased estimator. We have presented the LED objective function in [26] as another alternative for Lasso objective function and it is demonstrated that the resulting estimator outperforms the SCAD and adaptive Lasso estimators.

Based on the SCAD and adaptive Lasso objective functions, the TNWL and AdaLasso algorithms are presented in [27, 28, 32], respectively, which are the adaptive and sequential implementation of these objective functions. We have also

developed an adaptive and sequential solution for the LED objective function in [29], namely, LED-SAC, by solving the objective function in a coordinate-wise manner. It is shown that the proposed algorithm satisfies the oracle properties of asymptotic normality and consistency in variable selection and reaches better tracking performance compared to TNWL and AdaLasso.

Although the LED-SAC is a sparse reconstruction algorithm, it doesn't take the advantage of the sparsity of the signal to be estimated in its solution path. In this paper, we first study the complexity of LED-SAC algorithm. Then, a solution path is presented for LED-SAC algorithm, which benefits the sparsity of the signal to be estimated. More specifically, the most effective coordinates of the sparse signal are detected and the update procedure is done merely for those coordinates. Consequently, the resulting estimator, that is LED-2SAC, is more efficient and faster in terms of convergence rate compared to original one and it is shown that it has the same order of complexity as LED-SAC.

2. Summary of LED-SAC Reconstruction Algorithm

The exponentially decaying sparsity encouraging penalty function presented in [26] is as follows:

$$\mathcal{P}_{\tau_n, \gamma}(|\theta^*|) = \begin{cases} \tau_n |\theta^*|, & |\theta^*| \leq \tau_n, \\ \tau_n^2 + \tau_n \gamma (1 - e^{-(|\theta^*| - \tau_n)/\gamma}), & \text{otherwise,} \end{cases} \quad (4)$$

where τ_n and γ are the parameters of the penalty function which define its shape. Considering the penalty function in (4), the penalization rate reduces from a constant value to 0 (unlike standard Lasso), and the transition is smooth and controlled by the parameter γ , contrary to SCAD penalty, which has rough and linear decaying rate.

The overall objective function, which is obtained by augmenting the least squares loss function to the penalty function in (4), is nonconvex and cannot be solved via well-established convex optimization tools. Moreover, solving this nonconvex optimization problem may lead to a local minimum which is not the sparsest solution. In [26], we have obtained a convex approximation for this objective function by locally linearizing it using Taylor series expansion around some consistent estimate of $\boldsymbol{\theta}^*$. The resulting objective function is called LED objective function which is given as

$$\hat{\boldsymbol{\theta}} = \arg \min_{\boldsymbol{\theta}} \frac{1}{2n} \|\mathbf{y} - \mathbf{X}\boldsymbol{\theta}\|_2^2 + \sum_{j=1}^d \mathcal{P}'_{\tau_n, \gamma}(|\theta^{j, \circ}|) |\theta^j|, \quad (5)$$

where $\boldsymbol{\theta}^\circ$ is a consistent estimator of $\boldsymbol{\theta}^*$, such as the ordinary least squares solution (In [29] we have demonstrated the effects of decreasing the number of observations, n , while $\boldsymbol{\theta}^\circ$ is the ordinary least squares solution. The results show that, decreasing the number of observations degrades the performance of the sparse reconstruction algorithm. In this case, the solution of other suitable estimators like ridge regression estimator can be used, which is experimentally shown to

perform better). The properties of asymptotic normality and consistency hold for the LED objective function as long as the two constraints of $\lim_{n \rightarrow \infty} \tau_n \rightarrow 0$ and $\lim_{n \rightarrow \infty} \sqrt{n} \tau_n \rightarrow \infty$ are met, as n goes to infinity and the observation noise has finite variance [26].

In order to solve the approximated objective function in (5), one needs to solve a multivariate optimization problem, which is computationally too expensive to solve especially for sparse signals of higher dimension. In [29], a sequential and adaptive solution path is developed in order to solve the LED objective function in (5), called LED-SAC. The basic idea lies behind the fact that the objective function is convex and separable for each variables of the sparse signal θ . Therefore, the optimization problem is well suited to solve via the coordinate-wise optimization method. The proposed LED-SAC algorithm is capable of tracking the time variations in the support of the underlying sparse signal. The uniqueness of the LED-SAC algorithm is that it uses a novel sparsity-encouraging penalty and solves the overall objective function sequentially. As it is reported in [29], the LED-SAC estimator outperforms the TNWL and AdaLasso estimators in terms of mean squared error as well as tracking capability.

In what follows, we will explore the possibility of improving the convergence rate of the solution path for the LED-SAC algorithm. If so, a solution path is given to solve the objective function in (5), which itself is sparse in the sense that it has the sparsity of the signal to be estimated in mind, while pursuing the solution path. Therefore, the optimization method is itself sparse, which also leads to the sparsest solution. Consequently, the tracking capability of the algorithm will be increased, that is vital for online implementation, particularly in time varying sparse signal scenario.

3. The Proposed LED-Sparse SAC (LED-2SAC) Reconstruction Algorithm

Although the LED-SAC estimator restricts the solution of the under determined system of equations in (5) to the sparsest solution, it ignores the sparsity of the signal to be estimated in the solution path. This is due to the fact that at each iteration, it updates all of the coordinates of the sparse signal in cyclic manner, no matter which one of them belongs to the support of the true signal. This will significantly increase the number of observation needed to reach the desired precision in the reconstructed signal. As the signal of interest is sparse, most of its coordinates are zero and need not to be updated. Furthermore, in time varying sparse signal scenario, detecting and updating the coordinate, which has the most variation in its value, will significantly increase the tracking ability of the estimator.

Therefore, despite all advantages, updating whole coordinate set in a cyclic manner while most of them have zero values will still remain as the main shortcoming of LED-SAC estimator. This kind of blindly processing the coordinates of the signal to be estimated makes the reconstruction algorithm inefficient and rather slow to converge to a solution.

In order to consider the sparsity of the signal in the solution path and implement the LED-SAC more efficiently, one needs to introduce a procedure to detect the most effective coordinate/coordinates at each iteration and perform the update procedure for those coordinates. The objective function in (5) can be decomposed into two parts as follows:

$$\begin{aligned} \mathcal{F}(\theta) &= \overline{\mathcal{F}}(\theta^j) + \text{constant} \\ &= \frac{1}{2} [\mathbf{C}_\alpha(n)]_{jj} \theta^{j^2} \\ &\quad - \frac{1}{n} \beta(n, j) \theta^j + \mathcal{P}'_{\tau_n, \gamma}(|\theta^{j^o}|) |\theta^j| + \text{constant}, \end{aligned} \quad (6)$$

where $\overline{\mathcal{F}}(\theta^j)$ is the portion of $\mathcal{F}(\theta)$ which merely depends on θ^j and the term ‘‘constant’’ is a function of all of the entries of the parameter vector θ except θ^j , so it can be considered as constant with respect to θ^j .

Let us define $\Delta \overline{\mathcal{F}}(\theta^j)$ as the difference between $\overline{\mathcal{F}}(\theta_{(n)}^j)$ and $\overline{\mathcal{F}}(\theta_{(n-1)}^j)$, such that

$$\begin{aligned} \Delta \overline{\mathcal{F}}^j &= \overline{\mathcal{F}}(\theta_{(n)}^j) - \overline{\mathcal{F}}(\theta_{(n-1)}^j) \\ &= \frac{1}{2} [\mathbf{C}_\alpha(n)]_{jj} (\theta_{(n)}^{j^2} - \theta_{(n-1)}^{j^2}) \\ &\quad - \frac{1}{n} \beta(n, j) (\theta_{(n)}^j - \theta_{(n-1)}^j) \\ &\quad + \mathcal{P}'_{\tau_n, \gamma}(|\theta_{(n)}^{j^o}|) (|\theta_{(n)}^j| - |\theta_{(n-1)}^j|), \end{aligned} \quad (7)$$

where $\theta_{(n)}^j$ is the j th entry of the parameter vector θ at time n . One can select the most effective coordinate as the coordinate which makes the most changes to $\Delta \overline{\mathcal{F}}^j$, as [30]:

$$j^* = \arg \max_j \Delta \overline{\mathcal{F}}^j. \quad (8)$$

Selecting the most effective coordinate as in (8) leads to the best tracking performance for the adaptive estimator; however, it is to some extent computationally expensive. This is due to the fact that before obtaining j^* using (8), one has to obtain the estimated values for all of the elements of the sparse vector θ at the current time step, that is $\theta_{(n)}$, which is not a suitable strategy for online implementation of the reconstruction algorithm.

Another alternative which is much more straight forward, and has lower computational burden, is to use the directional derivatives of the objective function $\overline{\mathcal{F}}(\theta_{(n-1)}^j)$ in order to detect the most effective coordinate at each iteration. This approach is initially introduced in [31] for Lasso penalized regressions. In order to proceed in this way, we need to define the forward and backward derivatives of $\overline{\mathcal{F}}(\theta_{(n-1)}^j)$ with respect to $\theta_{(n-1)}^j$ as follows:

$$\begin{aligned} d_{+e^j} \overline{\mathcal{F}}(\theta_{(n-1)}^j) &= \lim_{\varepsilon \rightarrow 0} \frac{\overline{\mathcal{F}}(\theta_{(n-1)}^j + \varepsilon e^j) - \overline{\mathcal{F}}(\theta_{(n-1)}^j)}{\varepsilon} \\ d_{-e^j} \overline{\mathcal{F}}(\theta_{(n-1)}^j) &= \lim_{\varepsilon \rightarrow 0} \frac{\overline{\mathcal{F}}(\theta_{(n-1)}^j - \varepsilon e^j) - \overline{\mathcal{F}}(\theta_{(n-1)}^j)}{\varepsilon}, \end{aligned} \quad (9)$$

where e^j is the coordinate direction, along which $\theta_{(n-1)}^j$ varies. Taking the derivatives, we have

$$\begin{aligned} d_{+e^j} \overline{\mathcal{F}}(\theta_{(n-1)}^j) &= [\mathbf{C}_\alpha(n)]_{jj} \theta_{(n-1)}^j - \frac{1}{n} \beta(n, j) \\ &\quad + \mathcal{P}'_{\tau, \gamma}(|\theta^{j, o}|) \operatorname{sgn}(\theta_{(n-1)}^j), \\ d_{-e^j} \overline{\mathcal{F}}(\theta_{(n-1)}^j) &= -[\mathbf{C}_\alpha(n)]_{jj} \theta_{(n-1)}^j + \frac{1}{n} \beta(n, j) \\ &\quad - \mathcal{P}'_{\tau, \gamma}(|\theta^{j, o}|) \operatorname{sgn}(\theta_{(n-1)}^j). \end{aligned} \quad (10)$$

Considering the directional derivatives of $\overline{\mathcal{F}}(\theta_{(n-1)}^j)$ with respect to $\theta_{(n-1)}^j$ in (10), the most effective coordinate is the one which has the most negative value either for forward or backward derivatives as

$$j^* = \arg \min_j \{d_{+e^j} \overline{\mathcal{F}}(\theta_{(n-1)}^j), d_{-e^j} \overline{\mathcal{F}}(\theta_{(n-1)}^j)\}. \quad (11)$$

It is noteworthy that one can obtain the m most effective coordinates by finding the m coordinates which have the most negative values among d coordinates. This is of interest in the situations in which the signal sparsity, K , is known or there exists an approximate value for that. Furthermore, in the case of unknown sparsity, increasing the value of m will increase the convergence rate of the estimator. However, this will come at the cost of increasing the computational burden.

In the next section, we will give the complexity analysis for both LED-SAC and LED-2SAC algorithms and show that they both have the same order of complexity.

4. Complexity Analysis of LED-SAC and LED-2SAC Algorithms

The solution flow of the LED-SAC algorithm for one of the elements of the sparse signal, that is $\theta_{(n)}^j$, at time index n , is shown in Figure 1 where the shrinkage and update procedures are given in (12), (13), and (14), respectively, as follows:

$$\begin{aligned} \hat{\theta}_{(n)}^j &= \arg \min_{\theta^j} \overline{\mathcal{F}}(\theta^j) \\ &= \frac{1}{[\mathbf{C}_\alpha(n)]_{jj}} \operatorname{shrink} \left[\frac{1}{n} \beta(n, j), \mathcal{P}'_{\tau, \gamma}(|\theta^{j, o}|) \right], \end{aligned} \quad (12)$$

$$\mathbf{C}_\alpha(n) = a \mathbf{C}_\alpha(n-1) + \frac{1}{n} \mathbf{x}_n \mathbf{x}_n^T, \quad (13)$$

$$\beta(n, j) = a \beta(n-1, j) + x_n^j [y_n - \hat{y}_n] + (x_n^j)^2 \hat{\theta}_{(n-1)}^j, \quad (14)$$

where the shrink (\cdot) operator in (12) is defined such that $\operatorname{shrink}(a, b) = (|a| - b)_+ \operatorname{sgn}(a)$, where $(z)_+ = z$ if $z > 0$ and it is 0 otherwise. According to Figure 1, the input parameters are the observation and the corresponding regressor at time index n . In order to obtain $\hat{\theta}_{(n)}^j$, which is the estimate of the j th coordinate at time index n , using the shrinkage procedure in (12) one needs to have the j th simple least

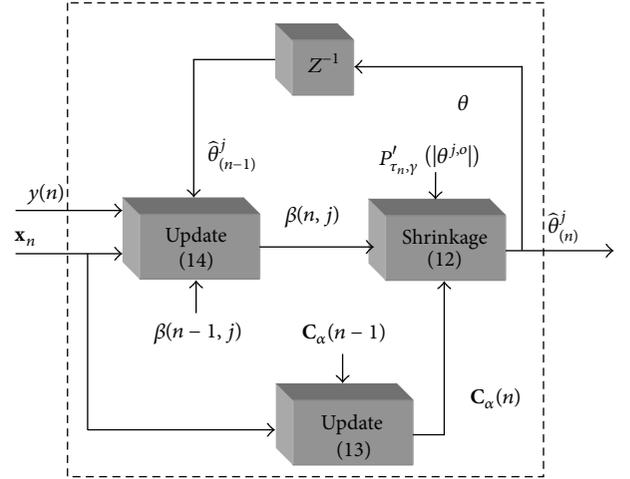


FIGURE 1: Block diagram of the estimation of one of the elements of the underlying sparse signal at the current time index, $\hat{\theta}_{(n)}^j$, for LED-SAC algorithm.

squares coefficient, that is $\beta(n, j)$, as well as $\mathbf{C}_\alpha(n)$, at the current time index. At each iteration, updating $\mathbf{C}_\alpha(n)$, having $\mathbf{C}_\alpha(n-1)$, needs $\mathcal{O}(d^2)$ algebraic operations. Having $\mathbf{C}_\alpha(n)$, obtaining $\beta(n, j)$ via (7), needs $\mathcal{O}(d)$ algebraic operations. Due to the fact that the recursive least squares estimate is used as the consistent estimate of the sparse signal θ_j^o in (14), and considering the complexity of the recursive least squares algorithm, which is of order $\mathcal{O}(d^2)$, the overall complexity of the LED-SAC algorithm at each time step is $\mathcal{O}(d^2)$ whereas the computational complexity of the well-known batch-based algorithm such as LARS, which performs multivariate optimization, is of order $\mathcal{O}(d^3 + nd^2)$. Thus, the LED-SAC algorithm is cheaper in terms of computational burden, especially for signals of higher dimensionality, while it is capable of tracking the variations in the sparse signal contrary to the LARS algorithm.

4.1. Computational Complexity of the LED-2SAC Algorithm. Taking into account the sparse solution path given in Section 3 and the recursive relations for the parameters $\mathbf{C}_\alpha(n)$ and $\beta(n, j)$ in (13) and (14), respectively, the pseudo code for efficient and fast implementation of the LED-SAC algorithm using the directional derivatives is given as LED-2SAC algorithm in Algorithm 1. The implementation is done for m most effective coordinates of the underlying sparse signal.

According to Algorithm 1, the computational complexity of LED-2SAC algorithm is the same as LED-SAC, except for the stage in which the indices of the most effective coordinates are to be estimated. For the case of $m = 1$, one needs to compute the forward and backward derivatives of $\overline{\mathcal{F}}(\theta_{(n-1)}^j)$ via (10), having the estimate of the sparse signal from the previous time step, that is $\theta_{(n-1)}$, and the simple least squares coefficients $\beta(n, j)$ for $j = 1, \dots, d$. Computing $d_{+e^j} \overline{\mathcal{F}}(\theta_{(n-1)}^j)$ and $d_{-e^j} \overline{\mathcal{F}}(\theta_{(n-1)}^j)$ for $j = 1, \dots, d$ requires $\mathcal{O}(d)$ operations,

```

Fast LED-SAC Algorithm
for  $n = 1, 2, \dots, n_{\max}$ 
  Obtain  $y_n$  and the corresponding  $\mathbf{x}_i$ 
  Update  $\mathbf{C}_\alpha(n)$  via (13)
  Update  $\beta(n, j)$  for  $j = 1, \dots, d$  via (14)
  Obtain the index of the  $m$  most effective set of coordinates via (11), as  $\Omega$ 
  Update  $\hat{\theta}_{(n)}^j = \hat{\theta}_{(n-1)}^j, \forall j \notin \Omega$ 
  Update  $\hat{\theta}_{(n)}^j, \forall j \in \Omega$  via (12)
end

```

ALGORITHM 1: The pseudo code of the LED-2SAC algorithm.

which do not affect the complexity order of the original algorithm, LED-SAC, which is of order $\mathcal{O}(d^2)$. Therefore, the presented algorithm, LED-2SAC, reaches higher convergence rate and tracking capability compared to LED-SAC algorithm with just a little increase in the computational burden.

5. Simulation Results

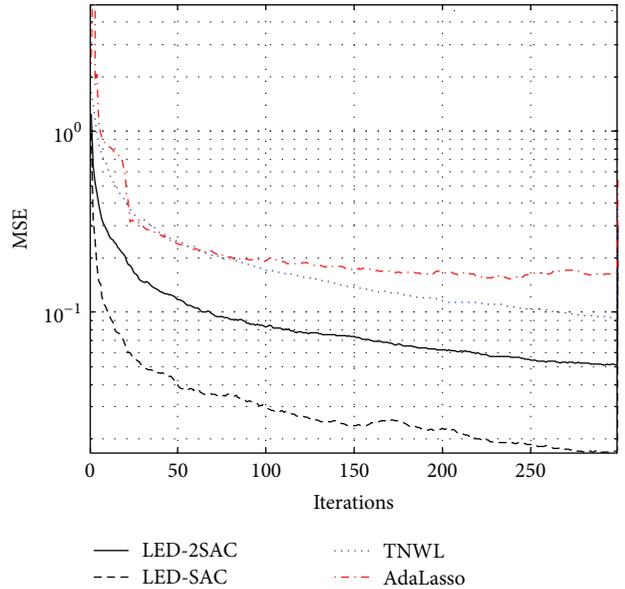
In what follows, the simulation results for the presented LED-2SAC reconstruction algorithm are given and compared to the former version, that is LED-SAC in [29], as well as TNWL reconstruction algorithm in [32] and AdaLasso algorithm presented in [24, 28].

Likewise [26, 29], the data set T is generated according to the model in (1) with $d = 30$ and $v_i \sim \mathcal{N}(0, 1)$. The regressors \mathbf{x}_i are also assumed to be samples from a Gaussian density of the form $\mathbf{x}_i \sim \mathcal{N}(\mathbf{0}_d, \mathbf{I}_d)$. The parameter vector $\boldsymbol{\theta}$ comprises K randomly allocated ± 1 entries, while all the other remaining entries are set to 0. Therefore, the tuning parameters for the parameter $\tau_n = n^{-\delta} \sqrt{\lambda \sigma^2 \log(d)}$ are set to the same values as in [26], which are extracted from cross validation, that is $(\lambda, \delta, \gamma) = (2.5, 2.7, 0.25)$.

In Figure 2, the learning curves of the algorithms are given in terms of the MSE plots. The MSE plots are obtained over 50 repetition of the experiment. The sparsity of the underlying signal to be estimated is set to 3 and the observation noise variance is set to 0.5. Comparing the convergence rate and the steady state error for the presented algorithm and the others reveals the superiority of the LED-2SAC algorithm. As it can be seen, the LED-2SAC algorithm has significantly improved the MSE performance, even compared with its former version, that is LED-SAC. This is due to the efficient implementation of the presented algorithm, such that at each iteration the most effective coordinate is detected and updated first.

In the next experiment, the sparsity of the signal is set to 10, and the results are shown in Figure 3. As it is seen, decreasing the sparsity of the signal has less affected the LED-2SAC algorithm, which is an interesting property for a sparsity aware reconstruction algorithm, since most of the existing methods fail to maintain their performance for lower sparsity levels.

In order to compare the performance of the algorithms in terms of variable selection capability, the percentage of

FIGURE 2: MSE curves over 100 iterations with $d = 30$, $K = 3$, and $\sigma^2 = 0.5$.

exact model selection is given in Figure 4 for two different observation noise powers versus the number of iterations (or number of observations in online implementation). As it is seen, the LED-2SAC algorithm reaches the best model selection rates in different situations. The TNWL and AdaLasso algorithms have almost the same performance in terms of variable selection, as reported in [29], which are overcome by LED-2SAC and LED-SAC algorithms.

The plots of percentage of exact model selection for the algorithms are also given in Figure 5 for two different sparsity levels. Again, in this case, the presented LED-2SAC has retained the superior performance. As it was expected, the performance of the algorithms gets better by increasing the sparsity of the signal and vice versa.

The presented LED-2SAC algorithm is a sequential and adaptive reconstruction algorithm as LED-SAC, TNWL, and AdaLasso. So, we have to demonstrate its ability to track the variations in the support of the sparse signal. In the following, the adaptation capabilities of the algorithms are compared. For this purpose, in an arbitrary iteration (after all of the

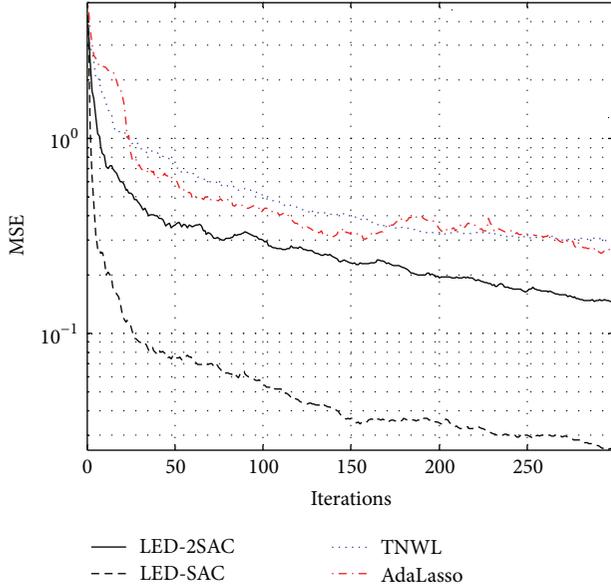


FIGURE 3: MSE curves over 100 iterations with $d = 30$, $K = 10$, and $\sigma^2 = 0.5$.

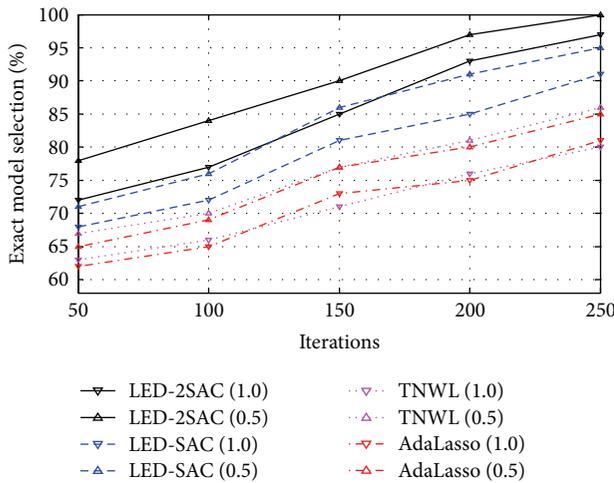


FIGURE 4: Percentage of exact model selection (different noise variances are denoted in the brackets).

algorithms are settled in their steady state performance), the support of the sparse signal is changed in the following fashion: one of the active coordinates is set to zero (inactive) and one of the inactive coordinates is set to 1 (active). The MSE curves are given in Figure 6 for $K = 3$.

As it is shown in Figure 6, the presented LED-2SAC algorithm has significantly improved the tracking capability of the former version, LED-SAC. However, both of the algorithms have outperformed the TNWL and AdaLasso algorithms. Figure 7 shows the same results for the case of $K = 10$. As it was expected from the results of Figures 2 and 3, the LED-2SAC algorithm has maintained its performance as the sparsity of the underlying signal is decreased.

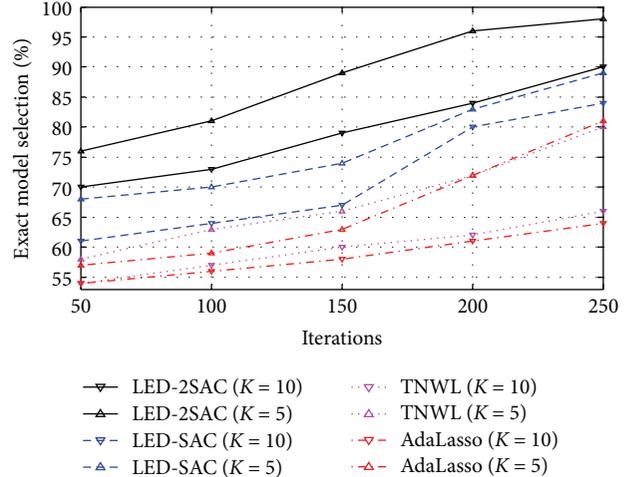


FIGURE 5: Percentage of exact model selection (different sparsity levels are denoted in the brackets).

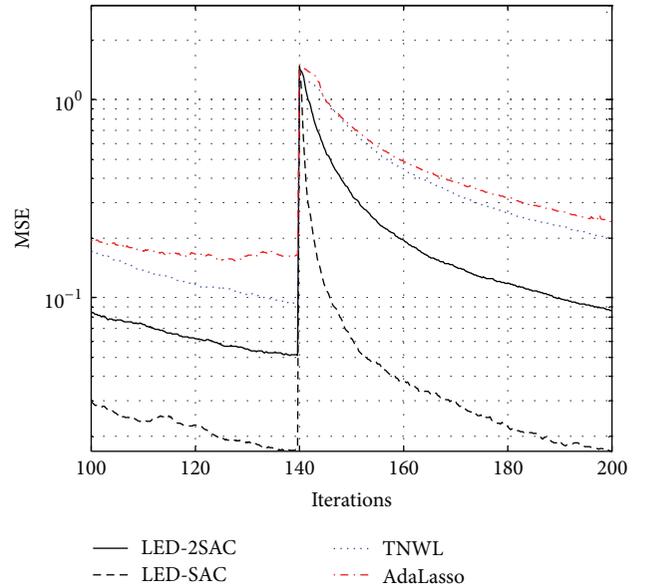


FIGURE 6: MSE when one of the active elements is set to zero and one of the inactive elements is set to one, $d = 30$, $K = 3$, and $\sigma^2 = 0.5$.

Finally, the estimation trajectories for an active coordinate becoming inactive and an inactive coordinate becoming active are shown in Figures 8 and 9, respectively. The LED-2SAC algorithm has outperformed all of the competitors as it was expected from the results of Figures 6 and 7. The interesting thing about Figure 8 is the fact that the LED-2SAC algorithm has almost abruptly detected and updated the inactive element. Although this is not the case for the entire coordinates, it happens more often.

6. Conclusion

In this paper, an efficient solution is proposed to sequentially solve the LED-based objective function that unlike the

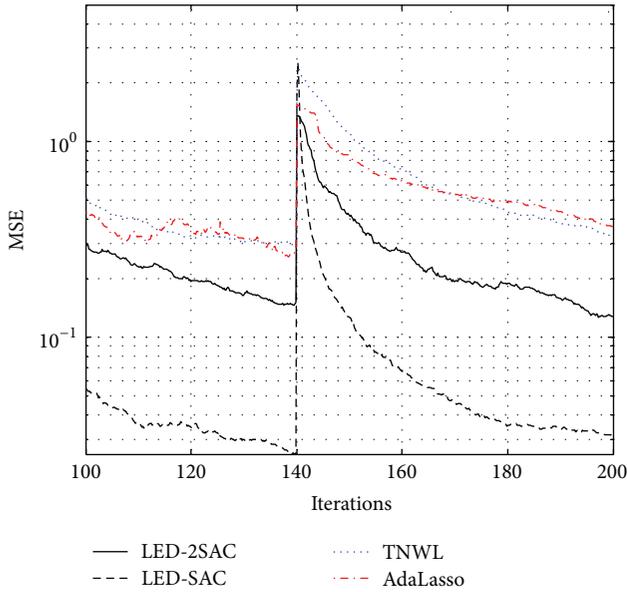


FIGURE 7: MSE when one of the active elements is set to zero and one of the inactive elements is set to one, $d = 30$, $K = 10$, and $\sigma^2 = 0.5$.

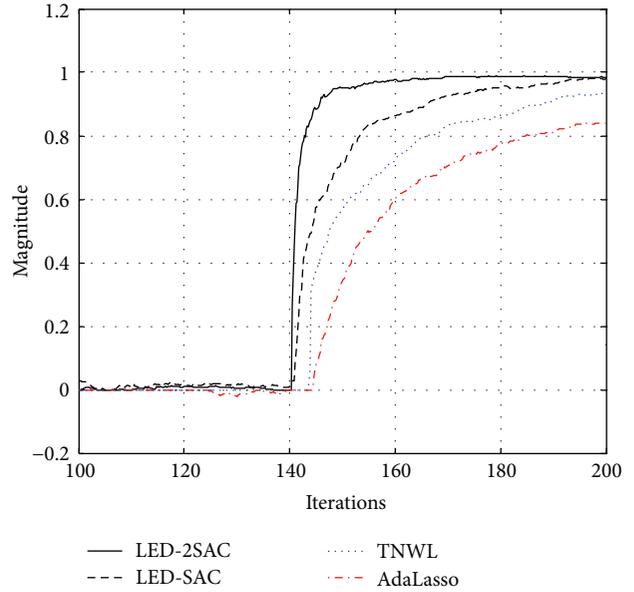


FIGURE 9: Estimation trajectory of the estimate of an inactive parameter becoming active.

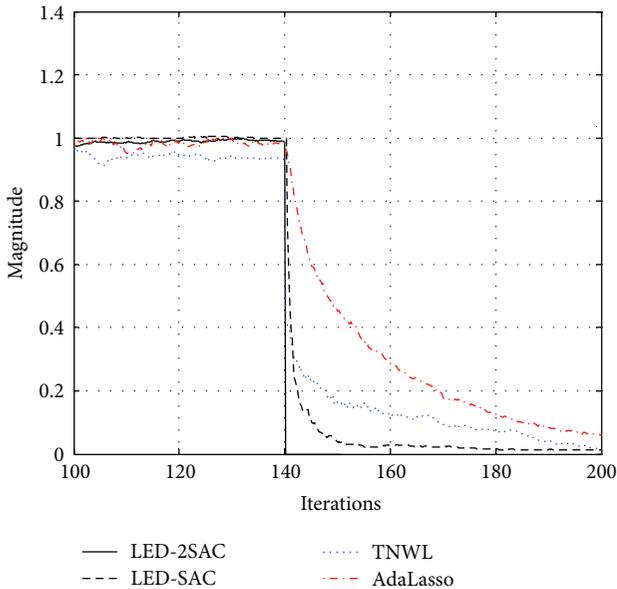


FIGURE 8: Estimation trajectory of the estimate of an active parameter becoming inactive.

existing SAC solution considers the sparsity of the signal to be estimated. Consequently, the proposed algorithm, denoted by LED-2SAC, leads to a significant improvement in the convergence rate and tracking capability of the original LED-SAC algorithm. Moreover, the complexity analysis of LED-SAC and LED-2SAC algorithms shows that both methods have the same order of complexity with additional improved convergence and adaptability behavior in LED-2SAC. Finally, the simulation results are given for the proposed algorithm. Comparing the performance of the presented algorithm with

the original one, as well as two other existing methods, confirms the superiority of the LED-2SAC in terms of convergence rate and adaptation capability.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 33–61, 1998.
- [2] J. A. Tropp, "Greed is good: algorithmic results for sparse approximation," *IEEE Transactions on Information Theory*, vol. 50, no. 10, pp. 2231–2242, 2004.
- [3] Y. Li and S. Osher, "Coordinate descent optimization for L_1 minimization with application to compressed sensing: a greedy algorithm," *Inverse Problems and Imaging*, vol. 3, no. 3, pp. 487–503, 2009.
- [4] M. Çetin, D. M. Malioutov, and A. S. Willsky, "A variational technique for source localization based on a sparse signal reconstruction perspective," in *Proceedings of the IEEE International Conference on Acoustic, Speech, and Signal Processing (ICASSP '02)*, pp. 2965–2968, May 2002.
- [5] E. J. Candès, M. B. Wakin, and S. P. Boyd, "Enhancing sparsity by reweighted L_1 minimization," *The Journal of Fourier Analysis and Applications*, vol. 14, no. 5-6, pp. 877–905, 2008.
- [6] R. Chartrand, "Exact reconstruction of sparse signals via non-convex minimization," *IEEE Signal Processing Letters*, vol. 14, no. 10, pp. 707–710, 2007.
- [7] R. Chartrand and W. Yin, "Iteratively reweighted algorithms for compressive sensing," in *Proceedings of the IEEE International*

- Conference on Acoustics, Speech and Signal Processing (ICASSP '08)*, pp. 3869–3872, Las Vegas, Nev, USA, April 2008.
- [8] C. J. Miosso, R. von Borries, M. Argàez, L. Velazquez, C. Quintero, and C. M. Potes, “Compressive sensing reconstruction with prior information by iteratively reweighted least-squares,” *IEEE Transactions on Signal Processing*, vol. 57, no. 6, pp. 2424–2431, 2009.
 - [9] D. M. Malioutov, S. R. Sanghavi, and A. S. Willsky, “Sequential compressed sensing,” *IEEE Journal on Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 435–444, 2010.
 - [10] J. Benesty and S. L. Gay, “An improved PNLMS algorithm,” in *Proceedings of the IEEE International Conference on Acoustic, Speech and Signal Processing (ICASSP '10)*, pp. 1881–1884, May 2002.
 - [11] D. L. Duttweiler, “Proportionate normalized least-mean-squares adaptation in echo cancelers,” *IEEE Transactions on Speech and Audio Processing*, vol. 8, no. 5, pp. 508–517, 2000.
 - [12] T. Gänslér, S. L. Gay, M. M. Sondhi, and J. Benesty, “Double-talk robust fast converging algorithms for network echo cancellation,” *IEEE Transactions on Speech and Audio Processing*, vol. 8, no. 6, pp. 656–663, 2000.
 - [13] O. Hoshuyama, R. A. Goubran, and A. Sugiyama, “A generalized proportionate variable step-size algorithm for fast changing acoustic environments,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '04)*, pp. 161–164, May 2004.
 - [14] J. Benesty, C. Paleologu, and S. Ciochina, “Proportionate adaptive filters from a basis pursuit perspective,” *IEEE Signal Processing Letters*, vol. 17, no. 12, pp. 985–988, 2010.
 - [15] Y. Chen, Y. Gu, and A. O. Hero III, “Sparse LMS for system identification,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '09)*, pp. 3125–3128, April 2009.
 - [16] Y. Gu, J. Jin, and S. Mei, “ L_0 norm constraint LMS algorithm for sparse system identification,” *IEEE Signal Processing Letters*, vol. 16, no. 9, pp. 774–777, 2009.
 - [17] J. Jin, Y. Gu, and S. Mei, “A stochastic gradient approach on compressive sensing signal reconstruction based on adaptive filtering framework,” *IEEE Journal on Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 409–420, 2010.
 - [18] E. M. Ekşioğlu, “RLS adaptive filtering with sparsity regularization,” in *Proceedings of the 10th International Conference on Information Sciences, Signal Processing and Their Applications (ISSPA '10)*, pp. 550–553, May 2010.
 - [19] R. Tibshirani, “Regression shrinkage and selection via the lasso,” *Journal of the Royal Statistical Society B*, vol. 58, no. 1, pp. 267–288, 1996.
 - [20] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, “Least angle regression,” *The Annals of Statistics*, vol. 32, no. 2, pp. 407–499, 2004.
 - [21] M. R. Osborne, B. Presnell, and B. A. Turlach, “A new approach to variable selection in least squares problems,” *IMA Journal of Numerical Analysis*, vol. 20, no. 3, pp. 389–403, 2000.
 - [22] J. Fan and R. Li, “Variable selection via nonconcave penalized likelihood and its oracle properties,” *Journal of the American Statistical Association*, vol. 96, no. 456, pp. 1348–1360, 2001.
 - [23] M. Yuan and Y. Lin, “On the nonnegative Garotte estimator,” Tech. Rep., Georgia Institute of Technology, School of Industrial and Systems Engineering, 2005.
 - [24] H. Zou, “The adaptive lasso and its oracle properties,” *Journal of the American Statistical Association*, vol. 101, no. 476, pp. 1418–1429, 2006.
 - [25] P. Zhao and B. Yu, “On model selection consistency of Lasso,” *Journal of Machine Learning Research*, vol. 7, pp. 2541–2563, 2006.
 - [26] T. Yousefi Rezaii, M. A. Tinati, and S. Beheshti, “Sparsity aware consistent and high precision variable selection,” *Journal of Signal, Image and Video Processing*, 2012.
 - [27] D. Angelosante and G. B. Giannakis, “RLS-weighted Lasso for adaptive estimation of sparse signals,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '09)*, pp. 3245–3248, April 2009.
 - [28] M. A. Tinati and T. Y. Rezaii, “Adaptive sparsity-aware parameter vector reconstruction with application to compressed sensing,” in *Proceedings of the International Conference on High Performance Computing and Simulation (HPCS '11)*, pp. 350–356, July 2011.
 - [29] T. Y. Rezaii, M. A. Tinati, and S. Beheshti, “Adaptive efficient sparse estimator achieving oracle properties,” *IET Signal Processing*, vol. 7, no. 4, pp. 259–268, 2013.
 - [30] Y. Li and S. Osher, “Coordinate descent optimization for L_1 minimization with application to compressed sensing; a greedy algorithm,” *Inverse Problems and Imaging*, vol. 3, no. 3, pp. 487–503, 2009.
 - [31] T. T. Wu and K. Lange, “Coordinate descent algorithms for lasso penalized regression,” *The Annals of Applied Statistics*, vol. 2, no. 1, pp. 224–244, 2008.
 - [32] D. Angelosante, J. A. Bazerque, and G. B. Giannakis, “Online adaptive estimation of sparse signals: where RLS meets the L_1 -norm,” *IEEE Transactions on Signal Processing*, vol. 58, no. 7, pp. 3436–3447, 2010.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

