Research Article

An Adaptive Gradient Projection Algorithm for Piecewise Convex Optimization and Its Application in Compressed Spectrum Sensing

Tianjing Wang,1 Hang Shen,2 Xiaomei Zhu,2 Guoqing Liu,1 and Hua Jiang1

1School of Physical and Mathematical Science, Nanjing Tech University, Nanjing 211816, China
2College of Computer Science and Technology, Nanjing Tech University, Nanjing 211816, China

Correspondence should be addressed to Tianjing Wang; wangtianjing@njtech.edu.cn

Received 18 November 2017; Revised 20 March 2018; Accepted 2 April 2018; Published 14 May 2018

Copyright © 2018 Tianjing Wang 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.

Signal sparse representation has attracted much attention in a wide range of application fields. A central aim of signal sparse representation is to find a sparse solution with the fewest nonzero entries from an underdetermined linear system, which leads to various optimization problems. In this paper, we propose an Adaptive Gradient Projection (AGP) algorithm to solve the piecewise convex optimization in signal sparse representation. To find a sparser solution, AGP provides an adaptive stepsize to move the iteration solution out of the attraction basin of a suboptimal sparse solution and enter the attraction basin of a sparser solution. Theoretical analyses are used to show its fast convergence property. The experimental results of real-world applications in compressed spectrum sensing show that AGP outperforms the traditional detection algorithms in low signal-to-noise-ratio environments.

1. Introduction

The marked advances in signal processing in recent years have been driven by the emergence of new signal models and their applications. Signal sparse representation is an effective model for solving real-world problems, such as brain signal processing [1], face recognition [2], compressed spectrum sensing [3], and singing voice separation [4].

Given a signal \( y \in \mathbb{R}^m \), signal sparse representation aims to identify the sparsest solution \( x \in \mathbb{R}^n \) from an underdetermined linear system \( y = Ax \), where \( A \in \mathbb{R}^{m \times n} \) is a full row rank matrix. The sparsity of a solution can be measured by \( l_0 \)-norm, which leads to the following optimization problem

\[
\min_x \|x\|_0 \quad \text{s.t.} \quad y = Ax. \tag{1}
\]

Unfortunately, problem (1) is NP-hard [5]. Many methods (e.g., the greedy algorithm [6], the \( l_1 \)-norm minimization [7], the \( l_p \)-norm \((0 < p < 1) \) minimization [8, 9], and the Bayesian method [10]) have been used to find sparse solutions. Because \( l_p \)-norm is the most effective measurement of the sparsity, some researchers are interested in \( l_p \)-norm minimization

\[
\min_x \|x\|_p^p \quad \text{s.t.} \quad y = Ax. \tag{2}
\]

The piecewise convex optimization (2) can be solved using the existing algorithms, including the Focal Underdetermined System Solver (FOCUSS) [11], the Affine Scaling Transformation (AST) method [12], the Iteratively Reweighted \( l_1 \) minimization (IRL1) [13], and the Iteratively Thresholding Method (ITM) [14]. The solutions they obtain, however, may be suboptimal sparse solutions.

In this paper, we propose a novel Adaptive Gradient Projection (AGP) algorithm for the piecewise convex optimization (2). This algorithm moves the iteration solution out of the attraction basin of a suboptimal sparse solution and finds a sparser solution in another attraction basin. The convergence analysis reveals that AGP performs better than
2. Adaptive Gradient Projection Algorithm for Piecewise Convex Optimization

2.1. Description of Affine Scaling Transformation Method. For convenience, problem (2) can be rewritten as

\[
\min_x E^p(x) = \sum_{i=1}^{n} |x(i)|^p \\
\text{s.t.} \quad y = Ax.
\]  
(3)

Note that the objective function is nondifferentiable with zero components. AST uses an affine scaling transformation to solve problem (3). For the \((k+1)\)th iteration, it defines a symmetric scaling matrix \(W_{k+1} = \text{diag}(|x(i)|^{1-p/2})\) and a scaled variable

\[ q = (W_{k+1})^{-1} x, \]  
(4)

Thus, problem (3) in \(x\) is transformed to the problem in \(q\)

\[
\min_q E^p(W_{k+1}q) \\
\text{s.t.} \quad y = A_{k+1}q,
\]  
(5)

where \(A_{k+1} = AW_{k+1}\).

Given a search direction \(l_k = p(I - A_{k+1}^+A_{k+1})q_k\), the new solution \(x_{k+1}\) is

\[
x_{k+1} = W_{k+1}q_{k+1} = x_k - \mu_k W_{k+1}l_k,
\]  
(6)

where \(I \in \mathbb{R}^{n \times n}\) is an identity matrix, \(A_{k+1}^+ = A_{k+1}^T(A_{k+1}A_{k+1}^T)^{-1}\) is a Moore-Penrose pseudoinverse matrix, and \(\mu_k\) is a stepsize.

Using a fixed stepsize \(\mu_k = 1/p\), AST is summarized as

\[
W_{k+1} = \text{diag}(|x_k(i)|^{1-p/2}),
\]  
(7)

\[
q_{k+1} = A_{k+1}^+y, \quad \text{where} \quad A_{k+1} = AW_{k+1},
\]  
(8)

\[
x_{k+1} = W_{k+1}q_{k+1}.
\]  
(9)

The convergence theorem of AST is as follows.

**Theorem 1.** Starting from an initial point \(x_0\), AST generates a sequence \(\{x_k\}_{k=1}^{\infty}\) converging to a sparse solution \(x^*\) of problem (3).

From (9), we see that some small entries of iteration solution converge to zero, because they are sequentially compressed by the scaling elements in \(W_{k+1}\). Thus, a sequence of iteration solutions of AST will converge to a sparse solution \(x^*\), which may be close to \(x_0\). However, this solution may be not the sparsest solution of problem (3). Making the iteration solution enter the attraction basin of other sparse solution is very important to reduce the effect of the initial point. Furthermore, Theorem 1 shows that AST obtains \(x^*\) within an infinite number of iterations, which affects the convergence rate. How to enhance the convergence speed of AST is another problem to be solved.

2.2. Derivation of Adaptive Gradient Projection Algorithm. To solve the above two problems, we first consider the convergence process of AST if an iteration solution has some zero entries.

**Lemma 2.** Given a block matrix \(P = \begin{pmatrix} P_1 & P_2 \\ P_3 & P_4 \end{pmatrix}\), we get

\[
P^{-1} = \begin{pmatrix}
-U^{-1}P_4P_3^{-1} & U^{-1} \\
-P_2^{-1}P_1U^{-1}P_3P_2^{-1} & -P_2^{-1}P_1U^{-1}
\end{pmatrix},
\]  
(10)

where \(U = P_3 - P_2P_2^{-1}P_1\) is a Schur complement.

**Lemma 3.** If \(x_k\) with \(S \) zero entries is a solution of problem (3), then these zero entries will not change in the remaining iterations.

For simplicity, let the front \(S\) entries of \(x_k\) be zero (i.e., \(x_k = (x_k^1, x_k^2)^T = (0, x_k^2)^T\), where \(x_k^1 = (x_k(1), \ldots, x_k(S))^T\), \(x_k^2 = (x_k(S+1), \ldots, x_k(n))^T\). Then, \(x_{k+1}\) can be computed as follows:

\[
x_{k+1} = W_{k+1}x_{k+1} = W_{k+1}A_{k+1}^+Ax_k
\]  
(11)

\[
= W_{k+1} (AW_{k+1})^+ Ax_k,
\]  
(12)

where \(W_{k+1} = \text{diag}(W_{k+1}^1, W_{k+1}^2) = \text{diag}(0, W_{k+1}^2)\). Partitioning \(A\), we calculate \(AW_{k+1}\) in (11) to be

\[
AW_{k+1} = \begin{pmatrix}
A_{11} & A_{12} \\
A_{21} & A_{22}
\end{pmatrix}
\]  
(13)

and its Moore-Penrose pseudoinverse matrix is

\[
(\hat{A}W_{k+1})^+ = (\hat{A}W_{k+1})^T (\hat{A}W_{k+1} (\hat{A}W_{k+1})^T)^{-1},
\]  
(14)

where

\[
(\hat{A}W_{k+1})^{-1} = \begin{pmatrix}
A_{12} (W_{k+1}^2)^2 A_{12} & A_{12} (W_{k+1}^2)^2 A_{22} \\
A_{22} (W_{k+1}^2)^2 A_{12} & A_{22} (W_{k+1}^2)^2 A_{22}
\end{pmatrix}^{-1}.
\]  
(15)
where $U = P$

out of the current attraction basin is a goal of the Adaptive suboptimal sparse solution. Moving the iteration solutions limited by the initial point; that is, the iteration solution is to accelerate convergence by sequentially shrinking the full rank, which has an invertible submatrix. For convenience, assume that $A_{12}(W_{k+1}^2)^T A_{12}^T$ is an invertible submatrix. Otherwise, we partition $(AW_{k+1}(AW_{k+1})^T)^{-1}$ to get an invertible submatrix according to Lemma 2. Defining $P_1 = A_{12}(W_{k+1}^2)^2 A_{12}^T$, $P_2 = A_{12}(W_{k+1}^2)^2 A_{22}^T$, $P_3 = A_{22}(W_{k+1}^2)^2 A_{12}^T$, $P_4 = A_{22}(W_{k+1}^2)^2 A_{22}^T$, we get

\[
(\mathbf{AW}_{k+1}(\mathbf{AW}_{k+1})^T)^{-1} = \begin{pmatrix}
B_{11} & B_{12} \\
B_{21} & B_{22}
\end{pmatrix}, \tag{15}
\]

where $U = P_3 - P_2 P_1^{-1} P_1$, $B_{11} = -U^{-1} P_1 P_2^{-1}$, $B_{22} = -P_2^{-1} P_1 U^{-1} P_1 P_2^{-1}$.

Substituting (12) and (15) into (13), we have

\[
(\mathbf{W}_{k+1})^+ = \begin{pmatrix}
0 & 0 \\
0 & \mathbf{C}_{21}
\end{pmatrix}, \tag{16}
\]

where $\mathbf{C}_{21} = (A_{12} W_{k+1}^2)^T B_{11} + (A_{22} W_{k+1}^2)^T B_{21}$, $C_{22} = (A_{12} W_{k+1}^2)^T B_{12} + (A_{22} W_{k+1}^2)^T B_{22}$.

Equation (11) then becomes

\[
x_{k+1} = W_{k+1} (\mathbf{AW}_{k+1})^+ A x_k = \begin{pmatrix}
0 & 0 \\
0 & W_{k+1}^2
\end{pmatrix} \begin{pmatrix}
(\mathbf{C}_{21}) & 0 \\
0 & \mathbf{C}_{22}
\end{pmatrix}
\begin{pmatrix}
A_{12} x_k^2 \\
A_{22} x_k^2
\end{pmatrix}, \tag{17}
\]

The front $S$ entries of $x_{k+1}$ in (17) are still zero, and the reverse cannot occur in the remaining iterations.

Lemma 3 implies that an unknown sparse solution can be identified in smaller and smaller subspace. It motivates us to accelerate convergence by sequentially shrinking the solving range. Meanwhile, the choice of subspace cannot be limited by the initial point; that is, the iteration solution is able to enter the subspace that does not contain the initial point. AST chooses a fixed stepsize, so it cannot move the iteration solution from one octant to another distant octant. These solutions concentrate in the attraction basin of the suboptimal sparse solution. Moving the iteration solutions out of the current attraction basin is a goal of the Adaptive Gradient Projection algorithm.

To find the search direction at an iteration solution $x$, we define the gradient

\[
g = \nabla_x E^{(p)}(x) = \begin{pmatrix}
\frac{\partial E^{(p)}(x)}{\partial x (1)}, & \ldots, & \frac{\partial E^{(p)}(x)}{\partial x (n)}
\end{pmatrix}^T, \tag{18}
\]

where

\[
\frac{\partial E^{(p)}(x)}{\partial x (i)} = \begin{cases}
p |x(i)|^{p-2} x(i), & \text{if } x(i) \neq 0 \\
delete, & \text{if } x(i) = 0,
\end{cases}
\]

and delete represents that the partial derivative is not computed as $x(i) = 0$, and the corresponding component is deleted from $g$. Definition (18) avoids the nondifferentiable problem of $p$-norm minimization such that the gradient can be calculated in subspace. Then, we provide the algorithm derivation below.

Initially, we reset an initial point $x_0 = A^T y$ as

\[
x_{D_k}(j) = \begin{cases}
x_0(i), & \text{if } x_0(i) \neq 0 \\
delete, & \text{if } x_0(i) = 0,
\end{cases} \tag{20}
\]

where the index set $D_0 = (a_1, \ldots, a_J)$ records the locations of entries as $x_0(i) \neq 0$. For example, if $x_0 = (0.2, -0.1, 0.5, 0, -0.2)^T$, then $D_0 = (1, 2, 3, 5)$ and $x_{D_0} = (0.2, -0.1, 0.5, -0.2)^T$.

For the $(k + 1)$th iteration, let $\overline{W}_{k+1} = A_{D_k} \overline{W}_{k+1}$; the gradient $\overline{g}_k$ and the search direction $\overline{l}_k$ are

\[
\overline{g}_k = \begin{pmatrix}
\frac{\partial E^{(p)}(x_{D_k})}{\partial x_{D_k}(1)}, & \ldots, & \frac{\partial E^{(p)}(x_{D_k})}{\partial x_{D_k}(J_k)}
\end{pmatrix}^T, \tag{21}
\]

\[
\overline{l}_k = (I_k - \overline{A}_{k+1}^* \overline{A}_{k+1}) \overline{g}_k,
\]

where $D_k = (a_1, \ldots, a_{J_k})$ records the locations of entries as $x_{D_k}(j) \neq 0$ ($j = 1, \ldots, J_k$), the column vectors of $A_{D_k}$ are selected from $A$ due to $D_k$, and $I_k \in R^{J_k \times J_k}$ is an identity matrix. In the span space of $A_{D_k}$, the new solution $\overline{x}_{k+1}$ is defined due to (6): $\overline{x}_{k+1} = x_{D_k} - \mu_k \overline{W}_{k+1} \overline{l}_k$, \tag{22}

where $\mu_k$ is a stepsize.

The major challenge in solving problem (3) is to identify the most appropriate subspace where the sparse solution locates. Equation (22) states that $\mu_k$ is important to find an appropriate subspace; thus, a function $E^{(p)}(\overline{x}_{k+1})$ with respect to $\mu$ is defined to investigate the property of the stepsize

\[
F(\mu) = E^{(p)}(x_{\overline{D}_k} - \mu \overline{W}_{k+1} \overline{l}_k). \tag{23}
\]

Figure 1(b) shows that the piecewise convex function $F(\mu)$ has nonunique minimum points that correspond to the zero entries of $\overline{x}_{k+1}$. We can compute these extreme points.

Without loss of generality, let every entry of $\overline{x}_{k+1}$ equal to zero (i.e., $\overline{x}_{k+1}(j) = 0$). Then, the extreme point $\mu_{k,j}$ is computed as

\[
\mu_{k,j} = \frac{x_{D_k}(j)}{\overline{W}_{k+1}(j)}, \tag{24}
\]

where $j = 1, \ldots, J_k$ and $\overline{V}_{k+1}(j) = \overline{W}_{k+1} \overline{l}_k$. An adaptive stepsize is chosen to find the minimal objective function value

\[
\mu_k^* = \arg \min_{\mu_{k,j}} E^{(p)}(x_{D_k} - \mu_{k,j} \overline{W}_{k+1} \overline{l}_k), \tag{25}
\]

and the new solution $\overline{x}_{k+1}$ is written as

\[
\overline{x}_{k+1} = x_{D_k} - \mu_{k,j}^* \overline{W}_{k+1} \overline{l}_k. \tag{26}
\]
The minimum point \((\mu^*_k, E^{(p)}(\bar{x}_{k+1}))\), marked as “*” in Figure 1(b), is determined by comparing three extreme values. Using the adaptive stepsize to obtain an iteration solution is beneficial to accelerate convergence. However, AST cannot obtain a minimum point at the search direction \(W_{k+1}l_k\) in (6), and the corresponding point \((1/p, E^{(p)}(x_{k+1}))\) is marked as “+” in Figure 1(a). Using the fixed stepsize makes the iteration solutions gather in the adjacent region of the initial point. On the other hand, we set \(\mu = \varepsilon\) (\(j = 1, \ldots, J\)), where \(\varepsilon\) is a threshold. AGP then determines multiple zero entries at each iteration so that it can quickly identify subspace where the sparse solution locates.

After some entries of \(\bar{x}_{k+1}\) become zero, the set of their indices \(d_{k+1}\) is deleted from \(D_k\) (i.e., \(D_{k+1} = D_k \setminus d_{k+1}\)). We define \(x_{D_{k+1}} \leftarrow x_{D_{k+1}}(D_{k+1})\) that represents the entries of \(\bar{x}_{k+1}\) located in \(D_{k+1}\) that are assigned to \(x_{D_{k+1}}\). Thus, (26) can be restated as

\[
x_{D_{k+1}} \leftarrow x_{D_k} - \mu_k \overline{W}_{k+1}l_{k+1}.
\]

For example, if \(\bar{x}_{k+1} = (0.3, 1 \times 10^{-6}, 0, 0, -0.1)^T\) and \(\varepsilon = 1 \times 10^{-4}\), then \(D_{k+1} = (1, 5)\) and \(x_{D_{k+1}} = (0.3, -0.1)^T\).

The iterations are performed until the needed support set is determined.

Returning the final solution \(x_{D_k}\) to \(n\)-dimensional space is convenient to obtain the sparse solution \(x^*\). By presetting \(x^* = 0 \in R^n\), we can use the entries of \(x_{D_k}\) that are assigned to \(x^*\). For example, if \(x_{D_k} = (0.3, -0.1)^T\) and \(D_k = (1, 5)\), then \(x^* = (0.3, 0, 0, 0, -0.1)^T\).

As discussed above, AGP is summarized as follows.

**Algorithm 4 (Adaptive Gradient Projection Algorithm).**

**Step 1** (initialization). Given a threshold \(\tau\), compute an initial point \(x_0 = A^T\gamma\) and set an index set \(D_0 = (\alpha_1, \ldots, \alpha_n)\) and the iteration index \(k = 0\).

**Step 2** (iteration)

\[
\textbf{while} \text{ Stopping criterion not met do}
\]

Compute \(\bar{g}_k\) and \(l_k\) due to (21),

Choose an adaptive stepsize by

\[
\mu_k^* = \arg \min_{\mu_k} E^{(p)}(x_{D_k} - \mu_k l_k \overline{W}_{k+1}^T),
\]

Compute a solution \(\bar{x}_{k+1} = x_{D_k} - \mu_k^* \overline{W}_{k+1}^T l_k\),

Update the solution \(x_{D_{k+1}} \leftarrow x_{D_k} - \mu_k^* \overline{W}_{k+1}^T l_k\),

Increase iteration index \(k\).

\[
\textbf{end while} \left| E^{(p)}(x_{D_{k+1}}) - E^{(p)}(x_{D_k}) \right| < \tau
\]

**Step 3** (solution). An optimal sparse solution \(x^* (D_{k+1}) \leftarrow x_{D_{k+1}}\).

2.3. Convergence Analysis. The convergence property of AGP is discussed as follows.

**Lemma 5.** AGP can determine at least one zero entry at each iteration.

Due to (26), there are three cases that AGP determines zero entry at each iteration. The first case is that one entry of \(x_{D_k}\) is set to zero, if \(x_{D_k}\) moves from one octant to the coordinate surface of another distant octant. In the second case, more than one entry of \(x_{D_k}\) become zero when the new iteration solution \(\bar{x}_{k+1}\) exactly locates on the coordinate axis of another distant octant. In the third case, if \(\bar{x}_{k+1}\) locates on the side of the coordinate axis of another distant octant, we set \(\bar{x}_{k+1}(j) = 0\) as \(|\bar{x}_{k+1}(j)| < \varepsilon\) (\(j = 1, \ldots, J_k\)). Therefore, AGP can determine at least one zero entry at each iteration.

**Theorem 6.** Let \(x^* \in R^n\) with \(K\) nonzero entries be a sparse solution of problem (3). The number of iterations of AGP is no more than \(n - K\).
If \( x^* \) has \( K \) nonzero entries, then the locations of \( n - K \) zero entries need to be determined. According to Lemma 5, AGP obtains multiple zero entries at each iteration, and these zero entries do not change in the remaining iterations based on the definition in (20). Therefore, the number of iterations of AGP is no more than \( n - K \).

Remark 7. Theorem 6 shows that AGP obtains a sparse solution within a finite number of iterations, while Theorem 1 shows that AST requires an infinite number of iterations to obtain a sparse solution. In theory, the number of iterations of AGP is less than AST.

Figure 2 gives an example to show the improved convergence performance of AGP compared to AST. There are three sparse solutions \( x_1^*, x_2^*, x_3^* \in \mathbb{R}^3 \), where \( x_3^* \) is the global optimal sparse solution. To clearly display the iteration process of AGP, we return all \( x_{D_k} \) to three dimensional space, which form a sequence \( \{x_k\}_{k=1}^L \), where \( x_k(D_k) \leftarrow x_{D_k} \) represents that the entries of \( x_{D_k} \) located in \( D_k \) which are assigned to \( x_k \).

Starting from \( x_0 \), a sequence \( \{x_k\}_{k=1}^6 \) solved by AST converges to \( x_1^* \) in Figure 2(a). Figure 2(c) shows that all iteration solutions concentrate in the attraction basin of \( x_1^* \) in the contour map. The second iteration solution of AGP in Figure 2(b), however, moves away from \( x_1^* \) and reaches \( x_3^* \). In Figure 2(d), the iteration solution moves out of the attraction basin of \( x_1^* \) and enters the attraction basin of \( x_3^* \), in which the adaptive step size plays an important role. This example verifies that AGP can find a sparser solution than AST by calculating the minimum point of the search direction at each iteration.

2.4. Comparison of Convergence Performance. To compare the convergence performance of AST and AGP, we give an experiment by the following \( l_p \)-norm problem

\[
\min E^{(0.5)}(x) = \sum_{i=1}^{8} |x(i)|^{0.5}
\]

\[
\text{s.t.} \quad \begin{align*}
x_1 + 9x_2 + x_3 + 9x_4 + 2x_5 + x_6 + 6x_7 + 7x_8 &= 11.5 \\
11x_1 + 9x_2 + 4x_3 + 5x_4 + 3x_5 + 10x_6 + x_7 + 4x_8 &= 13 \\
10x_2 + 3x_3 + 10x_4 + x_5 + 6x_6 + 10x_7 + 6x_8 &= 16.2
\end{align*}
\]
3. Application of Adaptive Gradient Projection Algorithm in Compressed Spectrum Sensing

The Compressive Spectrum Sensing (CSS) is considered for this study, because it performs the same tasks as signal sparse representation. In [15, 16], the model of CSS is formulated as follows:

\[ H_0: y = \Phi w, \quad \text{PU absent} \]

\[ H_1: y = \Phi (s + w), \quad \text{PU present} \]  \quad (30)

where \( y \) is a measurement, \( s \) is a Primary User (PU) signal, \( w \) is a Gaussian noise, \( s + w \) is a Secondary User (SU) received signal, and \( \Phi \in \mathbb{R}^{m \times n} \) is a Gaussian random matrix. Assume that \( s \) and \( w \) can be represented on the discrete cosine basis \( \Psi \), (i.e., \( s = \Psi x_s \) and \( w = \Psi x_w \)), where \( x_s \) and \( x_w \) are spectrum coefficients. Let \( A = \Phi \Psi \); the model in (30) can be reformulated as

\[ H_0: y = Ax_w, \quad \text{PU absent} \]

\[ H_1: y = A (x_s + x_w) = Ax_{sw}, \quad \text{PU present} \]  \quad (31)

CSS intends to reconstruct \( x_{sw}^* \) from \( y = Ax_{sw} \), so the reconstruction error \( \varepsilon_r = \| A x_{sw}^* - A x_{sw} \|_2 / \| A x_{sw} \|_2 \) is used to evaluate the reconstruction performance.

Corresponding to \( s \) and \( s + w \) in Figure 3(a), Figure 3(b) shows that \( x_s \) has \( K = 4 \) nonzero entries, while \( x_{sw} \) is...
Table 2: List of the partial iteration solutions solved by AGP.

<table>
<thead>
<tr>
<th>$k$</th>
<th>$x_{D_k}^*$</th>
<th>$D_k$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(0.0030, 0.6181, 0, 0.3752, −0.0115, 0.5101, 0.2972, 0.0410)</td>
<td>(1, 2, 3, 4, 5, 6, 7, 8)</td>
</tr>
<tr>
<td>2</td>
<td>(−0.0005, 0.6860, 0.3288, 0.0017, 0.4870, 0.3128)</td>
<td>(1, 2, 4, 5, 6, 7)</td>
</tr>
<tr>
<td>3</td>
<td>(−0.0070, 1.1946, 0.0229, 0.7034, 0.0010)</td>
<td>(1, 4, 5, 6, 7)</td>
</tr>
<tr>
<td>4</td>
<td>(1.2000, 0.7000)</td>
<td>(4, 6)</td>
</tr>
</tbody>
</table>

Reconstructed denoising signal by AST
Reconstructed denoising signal by AGP

Denoising coefficient by AST
Denoising coefficient by AGP

Figure 4: SU received signals are reconstructed with the denoising spectrum coefficients solved by AST and AGP, respectively. (a) Reconstructed SU received signals based on AST and AGP. (b) Denoising spectrum coefficients solved by AST and AGP.

Distributed throughout the frequency domain. The reconstructed spectrum coefficient is obtained by solving the following problem

$$
\min_x E^{(0.7)}(x_{sw}) = \sum_{i=1}^{64} |x_{sw}(i)|^{0.7}
$$

s.t. $y = Ax_{sw},$  \hspace{1cm} (32)

where $A \in \mathbb{R}^{20 \times 64}$. After 21 iterations, $x_{AST}^*$ solved by AST in Figure 3(d) is unsatisfactory, while $x_{AGP}^*$ solved by AGP converges to $x_{sw}$ after 17 iterations. The corresponding computing times of AST and AGP are 0.0154 s and 0.0227 s, respectively. The reconstructed signals $\Phi x_{AST}^*$ and $\Phi x_{AGP}^*$ are shown in Figure 3(c), and the reconstruction errors are 27.22% and 9.99%, respectively. At the cost of a little computing time, the reconstruction performance of AGP is improved compared to AST in noise environment, so AGP exhibits better performance of noise suppression than AST.

Note that the characteristic of noise suppressing can greatly improve the detection performance of CSS, especially when the number of nonzero entries $K$ is not given in advance. We can reconstruct the SU received signal by choosing some larger nonzero entries of $x_{sw}^*$ with a threshold $\eta$. Setting $\eta = 10$, Figure 4(b) shows the denoising spectrum coefficients are sparser than the coefficients in Figure 3(d). The corresponding reconstructed SU received signals are shown in Figure 4(a). The reconstruction errors reduce to 21.71% and 8.50%. Meanwhile, the variance $\sigma_w^2$ of white noise $w$ reduces to $\sigma_w^2$.

Next, we consider the detection performance using the reconstructed SU signal with denoising coefficient. Let $P_f$ be a false alarm probability and $\lambda$ be a judgment threshold. Then, a binary hypothesis testing problem is used to determine whether the PU is present

$$
P_d = \begin{cases} 0, & H_0 : e(s_d) < \lambda \\ 1, & H_1 : e(s_d) \geq \lambda, \end{cases}
$$

where $e(s_d) = \|s_d\|_2^2$ denotes the energy of the detection signal $s_d = \Psi x_d$. The $N_t$ Monte Carlo experiments are performed to test the detection probability $P_d = N_d/N_t$, where $N_d$ is the number of accurately detecting PU signal. Given $P_f = 0.05$, $N_t = 100$, Figure 5 shows that the Energy Detection (ED) method [17] exhibits high mistaken probabilities in low signal-to-noise-ratio (SNR) environments. However, the detection probabilities of AST, AGP, IRL1, and ITM are greatly enhanced, when the reconstructed SU received signals are solved by the denoising spectrum coefficients. For example, when SNR equals to $-5$ dB, the detection probabilities using AST, AGP, IRL1, and ITM improve by 75.47%, 79.25%, 58.49%, and 47.17% compared with ED. Furthermore, when SNR changes from $-15$ dB to $-1$ dB, AGP shows better detection performance than AST, IRL1, and ITM because of its improved reconstruction performance.
Figure 5: Comparison of the detection performance using ED, ITM, AST, AGP, and IRL1 for different SNRs.

Figure 6: Comparison of the computing time using ED, ITM, AST, AGP, and IRL1 for different SNRs.

Figure 7: Comparison of the detection performance using ED, ITM, AST, AGP, and IRL1 for different sparsity of the spectrum coefficient vector.

Figure 8: Comparison of the computing time using ED, ITM, AST, AGP, and IRL1 for different sparsity of the spectrum coefficient vector.

Figure 6 shows the corresponding computing time of five reconstruction algorithms, in which the time consumption of AGP increases at most 43.66% and 23.39% compared with ITM and AST. Spending a little more computing time, AGP can attain the best result of spectrum sensing especially in low SNR environment.

Note that the sparsity of the spectrum coefficient vector has impact on the reconstruction performance of $l_p$-norm minimization. Given SNR is $-7$ dB, Figure 7 displays that the detection probabilities of AST, AGP, IRL1, and ITM descend when $K$ changes from 2 to 10. This is because the measurement vector $y \in R^{20}$ in (32) is not able to attain the whole information of a SU received signal when $K$ is larger than 6. Therefore, the unsatisfactory reconstruction results of $l_p$-norm minimization greatly affect the detection performance of CSS via AST, IRL1, and ITM, while the performance degradation of CSS via AGP is slower than that of the three reconstruction algorithms. Meanwhile, AST, AGP, IRL1, and ITM cost more computing time to find the sparse solution in Figure 8. The above results reveal that the detection performance of CSS needs to be improved when the number of measurement is insufficient. ED determines the state of PU by measuring the energy of a SU received signal, so the property of the sparsity has little impact on its detection result and computing time.
4. Conclusions

Signal sparse representation has become a fundamental tool that is embedded into various application systems. One of its fundamental problems is finding a sparse coefficient. In this paper, we develop a novel AGP algorithm to solve the $l_p$-norm minimization. Theoretical analysis demonstrates that AGP can find a sparser solution than AST, because it avoids the iterative solutions concentrating in the attraction basin of a suboptimal sparse solution. Applying AGP to compressed spectrum sensing, it can obtain the better detection performance than ED, AST, IRL1, and ITM by spending a little more computing time. Future research will extend AGP to more scenarios.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This research was supported by the National Natural Science Foundation of China (nos. 61501224, 61501223, and 61502230), the Natural Science Foundation of Jiangsu Province (no. BK20150960), and the Natural Science Foundation of the Jiangsu Higher Education Institutions of China (no. 17KJB510024).

References


