Hindawi
Mathematical Problems in Engineering
Volume 2019, Article ID 1978154, 12 pages
https://doi.org/10.1155/2019/1978154

Research Article

A New Smoothed L0 Regularization Approach for Sparse Signal Recovery

Jianhong Xiang, Huihui Yue, Xiangjun Yin, and Linyu Wang

College of Information and Communication Engineering, Harbin Engineering University, Harbin 150001, China

Correspondence should be addressed to Xiangjun Yin; yinxiangjun@hrbeu.edu.cn

Received 10 October 2018; Revised 20 January 2019; Accepted 5 March 2019; Published 17 July 2019

Academic Editor: Wlodzimierz Ogryczak

Sparse signal reconstruction, as the main link of compressive sensing (CS) theory, has attracted extensive attention in recent years. The essence of sparse signal reconstruction is how to recover the original signal accurately and effectively from an underdetermined linear system equation (ULSE). For this problem, we propose a new algorithm called regularization reweighted smoothed L_0 norm minimization algorithm, which is simply called RRSL0 algorithm. Three innovations are made under the framework of this method: (1) a new smoothed function called compound inverse proportional function (CIPF) is proposed; (2) a new reweighted function is proposed; and (3) a mixed conjugate gradient (MCG) method is proposed. In this algorithm, the reweighted function and the new smoothed function are combined as the sparsity promoting objective, and the constraint condition $\|\mathbf{y} - \mathbf{\Phi x}\|_2^2$ is taken as a deviation term. Both of them constitute an unconstrained optimization problem under the *Tikhonov* regularization criterion and the MCG method constructed is used to optimize the problem and realize high-precision reconstruction of sparse signals under noise conditions. Sparse signal recovery experiments on both the simulated and real data show the proposed RRSL0 algorithm performs better than other popular approaches and achieves state-of-the-art performances in signal and image processing.

1. Introduction

CS [1, 2] has been successfully applied in a multitude of scientific fields, ranging from image processing tasks to radar to coding theory, making the potential impact of advancements in theory and practice rather large. In fact, various CS tasks eventually boil down to the sparse signal recovery problem in the following underdetermined linear system:

$$\mathbf{y} = \mathbf{\Phi x} + \mathbf{w}, \tag{1}$$

where $\mathbf{\Phi} \in \mathbb{R}^{m \times n}$ is a sensing matrix (also known as measurement matrix). $\mathbf{x} \in \mathbb{R}^n$ is the sparse vector (or signal) to be solved. $\mathbf{y} \in \mathbb{R}^m$ is the vector (or signal) of measurements. Moreover, the $\mathbf{w} \in \mathbb{R}^m$ denotes the additive noise.

For solving the ULSE in (1), we try to recover the sparse signal $\mathbf{x}$ from given $\{\mathbf{y}, \mathbf{\Phi}\}$. In this case, the sensing matrix $\mathbf{\Phi}$ contains more columns than rows, which means there would be more than one solution that satisfies the constraint $\|\mathbf{\Phi x} - \mathbf{y}\|_2^2 \le \epsilon$ (ϵ is a very small constant). This makes the recovery of sparse signal $\mathbf{x}$ an ill-posed problem. Luckily, since the target signal itself is sparse, the most straightforward method is to use its sparsity to improve the problem. That is, the problem of recovering the target signal $\mathbf{x}$ can be converted into solving the L_0-norm minimization problem as follows:

$$\begin{aligned} (P_0) \quad & \arg\min_{\mathbf{x} \in \mathbb{R}^n} \quad \|\mathbf{x}\|_0, \\ & \text{s.t.} \quad \left\|\mathbf{\Phi x} - \mathbf{y}\right\|_2^2 \le \epsilon. \end{aligned} \tag{2}$$

This rather wonderful transformation is actually supported by outstanding theory [3]. The theory points out that, under the condition of no noise, the sparsest solution is indeed a real signal if $\mathbf{x}$ is sufficiently sparse and $\mathbf{\Phi}$ satisfies the restricted isometry property (RIP) [4]:

$$\left(1 - \delta_K\right) \|\mathbf{x}\|_2^2 \le \|\mathbf{\Phi x}\|_2^2 \le \left(1 + \delta_K\right) \|\mathbf{x}\|_2^2, \tag{3}$$

where K is the sparsity of signal $\mathbf{x}$. In (2), L_0-norm is nonconvex, which leads to a NP-hard problem. In fact, there are two alternative approaches to solve this problem [5, 6]:

(i) Greedy search

(ii) Relaxation method for the P_0

Greedy search requires known sparsity as a constraint and the main methods are the approximate algorithms based on greedy matching pursuit (GMP) algorithms, such as OMP [7], StOMP [8], ROMP [9], CoSaMP [10], GOMP [11], and SP [12] algorithms. The objective function of these algorithms is given by [6]

$$\begin{aligned} \left(P_f\right) \quad & \arg\min_{\mathbf{x}\in\mathbb{R}^n} \quad \frac{1}{2}\left\|\mathbf{\Phi x}-\mathbf{y}\right\|_2^2, \\ & \text{s.t.} \quad \|\mathbf{x}\|_0 \le K. \end{aligned} \tag{4}$$

where K is the sparsity of signal $\mathbf{x}$ as shown in (3).

Based on this, the characteristics of GMP algorithms can be summarized as follows [6]:

(i) Sparsity is used as a prior information

(ii) Least square error is employed as an iterative criterion

The main advantage of GMP algorithms is that it is simple to calculate, but its application range is limited due to its low reconstruction accuracy in the case of noise.

At present, relaxation method for the P_0 is a main method. Relaxation method is mainly divided into two categories: L_1-norm minimization methods and smoothed L_0-norm minimization methods. The representative algorithm of the former is BP algorithm [13], and the latter is SL0 algorithm. Both are optimized using an approximate or equivalent L_0-norm objective function. Therefore, the effect of sparse signal recovery is similar to that of GMP algorithms.

The above sparse signal recovery methods focus more on signal recovery in the absence of noise; they work not well in noise case. However, sparse signal recovery under noise is a very realistic and inevitable problem. Fortunately, the regularization mechanism makes it possible to solve this problem. The regularization mechanism relaxes P_0 into the following unconstrained recovery problem [6]:

$$\left(P_g\right) \quad \arg\min_{\mathbf{x}\in\mathbb{R}^n} \quad \frac{1}{2}\left\|\mathbf{\Phi x}-\mathbf{y}\right\|_2^2 + \lambda \mathrm{g}\left(\mathbf{x}\right), \tag{5}$$

where $\lambda > 0$ is the parameter that balances the trade-off between the deviation term $\|\mathbf{\Phi x}-\mathbf{y}\|_2^2$ and the sparsity regularizer $\mathrm{g}(\mathbf{x})$. The sparse prior information is enforced by the regularizer $\mathrm{g}(\mathbf{x})$. Proper $\mathrm{g}(\mathbf{x})$ is crucial to the successful recovery of sparse signal: it should support sparse solutions and ensure that problem P_g can be effectively solved simultaneously.

For regularization, there are many sparsity regularizers for relaxing the L_0-norm, among which the convex L_1-norm [14, 15] and the nonconvex L_p-norm to the p-th power [16–18] are the main methods:

(i) L_1-norm:

(1) $\|\mathbf{x}\|_1 = \sum_{i=1}^n |x_i|$.

(ii) L_p-norm to the p-th power:

(1) $\|\mathbf{x}\|_p^p = \sum_{i=1}^n |x_i|^p$;
(2) $\|\mathbf{x}\|_p^p \approx \sum_{i=1}^n (x_i^2+\epsilon^2)^{p/2}$;
(3) $\|\mathbf{x}\|_p^p \approx \sum_{i=1}^n (|x_i|+\epsilon)^{p-1}|x_i|$.

In the case of no noise, L_1-norm is equivalent to L_0-norm, and since L_1-norm is the only norm with sparsity and convexity, the sparse solution can be approximated by convex optimization method. However, the equivalence between L_1-norm and L_0-norm is no longer valid in the case of noise, so the effect of using L_1-norm to promote sparsity is not obvious in this case. Compared with L_1-norm, the nonconvex L_p- norm to the p-th power is closer to the L_0-norm, so L_p-norm minimization performs better than L_1-norm minimization in recovering sparse signal. Based on the idea of approximation, a Gauss function is employed to approximate the L_0-norm in [19]. Moreover, [20] uses the Gauss function as a sparsity regularizer. The approximation of L_0-norm by Gauss function can be expressed as follows [6]:

$$\begin{aligned} \|\mathbf{x}\|_0 &\approx \lim_{\sigma\longrightarrow 0} F_\sigma\left(\mathbf{x}\right) = \lim_{\sigma\longrightarrow 0}\sum_{i=1}^n f_\sigma\left(x_i\right), \\ f_\sigma\left(x_i\right) &= 1-\exp\left(-\frac{x_i^2}{2\sigma^2}\right). \end{aligned} \tag{6}$$

According to this equation, it is obvious that

$$f_\sigma\left(x_i\right) \approx \begin{cases} 1 & \text{if } x_i \gg \sigma \\ 0 & \text{if } x_i \ll \sigma. \end{cases} \tag{7}$$

When σ is a small enough positive value, the sparse regularizer almost equivalent to L_0-norm, so it can promote sparsity. Furthermore, the Gauss function can be optimized by optimization methods because it is differentiable and smoothed. Based on this idea, a hyperbolic tangent (tanh) function is proposed in [21]:

$$\|\mathbf{x}\|_0 \approx \lim_{\sigma\longrightarrow 0}\sum_{i=1}^n \frac{\exp\left(x_i^2/2\sigma^2\right)-\exp\left(-x_i^2/2\sigma^2\right)}{\exp\left(x_i^2/2\sigma^2\right)+\exp\left(-x_i^2/2\sigma^2\right)}. \tag{8}$$

The tanh function proposed in [21] is a smoothed function, and it performs better in approximating L_0-norm than the Gauss function in [20]. Based on this, the tanh function has better performance in sparse signal recovery. Indeed, we confirmed this view by a large number of simulation experiments.

In this paper, we propose a CIPF function as new sparsity regularizer and show their effectiveness and advantages over other popular regularizers in promoting sparse solutions with both theoretical analyses and experimental evaluations.

This paper is organized as follows. Section 2 introduces the main contributions of the proposed RRSL0 algorithm. In Section 3, the steps of the RRSL0 algorithm and the selection of related parameters are described in detail. Section 4 provides the experimental results to evaluate the performance of the proposed method. The paper is finally concluded in Section 5.

2. Main Contributions of the Proposed RRSL0 Algorithm

In this paper, based on the P_g in (5), we propose a new objective function, which is given by

$$\arg\min_{\mathbf{x}\in\mathbb{R}^n} \quad \frac{1}{2}\left\|\mathbf{\Phi x}-\mathbf{y}\right\|_2^2 + \lambda \mathbf{W} H_\sigma\left(\mathbf{x}\right). \tag{9}$$

From the equation, we can see that there is an obvious difference compared with other normalized objective functions. The obvious difference is objective regularizer. In this paper, as an objective regularizer, we not only propose a smoothed function approximating L_0 norm but also propose a reweighted function to promote sparsity. This section focuses on the relevant contents of reweighted function $\mathbf{W} = [w_1, w_2, \ldots, w_n]^T$ and $H_\sigma(\mathbf{x})$.

2.1. New Smoothed Function: CIPF. According to [22], some properties of the smoothed functions are summarized in the following.

Property. Let $f : \mathbb{R} \longrightarrow [-\infty, +\infty]$ and define $f_\sigma(x) \approx f_\sigma(x/\sigma)$ for any $\sigma > 0$. The function f has the Property, if

(a) f is real analytic on (x_0, ∞) for some x_0;

(b) $\forall x \geq 0$, $f''(x) \geq -\epsilon_0$, where $\epsilon_0 > 0$ is some constant;

(c) f is convex on $\mathbb{R}$;

(d) $f(x) = 0 \longleftrightarrow x = 0$;

(e) $\lim_{x \longrightarrow +\infty} f(x) = 1$.

It follows immediately from *Property* that $\{f_\sigma(x)\}$ converges to L_0-norm as $\sigma \longrightarrow 0^+$; i.e.,

$$\lim_{\sigma \longrightarrow +\infty} f_\sigma(x) = \begin{cases} 0 & \text{if } x = 0 \\ 1 & \text{otherwise.} \end{cases} \tag{10}$$

Based on the *Property*, this paper proposes a new smoothed function model called CIPF, which satisfies the *Property* and better approximates L_0-norm. The smoothed function model is given as

$$f_\sigma(x) = \frac{cx^2}{cx^2 + \sigma^2}. \tag{11}$$

In (11), regulatory factor c is a large constant (because it satisfies $\sigma^2/c \longrightarrow 0$ for any σ). By experiments, the factor c is determined to be 10, which is the good result of the simulation. σ represents a smoothed factor, and when it is smaller, it will make the proposed model closer to the L_0-norm. Obviously,

$$\lim_{\sigma \longrightarrow 0} f_\sigma(x) = \begin{cases} 0, & x = 0 \\ 1, & x \neq 0 \end{cases} \tag{12}$$

or approximately

$$f_\sigma(x) \approx \begin{cases} 0, & |x| \ll \sigma \\ 1, & |x| \gg \sigma \end{cases} \tag{13}$$

is satisfied. Let

$$H_\sigma(\mathbf{x}) = \sum_{i=1}^{n} f_\sigma(x_i) = \sum_{i=1}^{n} \frac{cx_i^2}{cx_i^2 + \sigma^2} \tag{14}$$

where $H_\sigma(\mathbf{x}) \approx \|\mathbf{x}\|_0$ for small values of σ, and the approximation tends to equality when $\sigma \longrightarrow 0$.

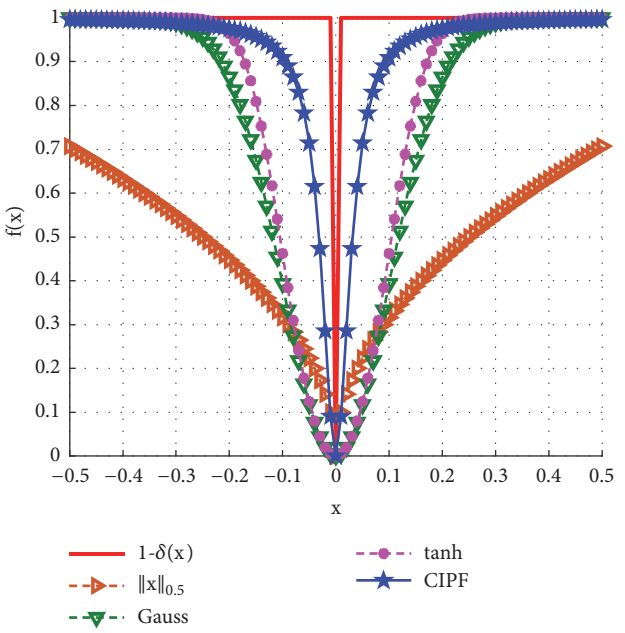


Figure 1: Different function used in the literature to approximate the L_0-norm and some of them are plotted in this figure and display $L_{0.5}$-norm for comparison.

Figure 1 shows the situation of CIPF model approximates the L_0-norm. Obviously, the CIPF model makes a better approximation.

In conclusion, the merits of CIPF model can be summarized as follows:

(i) It closely approximates L_0-norm;

(ii) It is simpler than Gauss and tanh function model.

These merits make it possible to reduce the computational complexity on the premise of ensuring the sparse signal reconstruction accuracy, which is of practical significance for sparse signal reconstruction.

2.2. New Reweighted Function. Candès et al. [23] proposed the reweighted L_1-norm minimization method, which employs the reweighted norm to enhance the sparsity of the solution. Moreover, they provided an analytical result of the improvement in the sparsity recovery by incorporating reweighted function to the objective function. Pant et al. [24] applied another reweighted smoothed L_0-norm minimization method, which uses a similar reweighted function to improve sparsity. The reweighted function can be summarized as follows:

(i) Candès et al.:

$$w_i = \begin{cases} \dfrac{1}{|x_i|} & x_i \neq 0 \\ \infty & x_i = 0; \end{cases} \tag{15}$$

(ii) Pant et al.: $w_i = 1/(|x_i| + \zeta)$, ζ is a small enough positive constant.

From the two reweighted functions, we can find a phenomenon: a large signal entry x_i is reweighted with a small value w_i; on the contrary, a small signal entry x_i is reweighted with a large value w_i. By analysis, the large w_i force the solution $\mathbf{x}$ to concentrate on the indices where w_i is small, and by construction these correspond precisely to the indices where $\mathbf{x}$ is nonzero.

Combined with the above idea, we propose a new reweighted function, which is given by

$$w_i = \frac{1}{e^{(x_i)^2}}, \quad \text{s.t. } i = 1, 2, \ldots, n. \tag{16}$$

As for Candès et al., when signal entry is zero or close to zero, the value of w_i will be very large, which is not suitable for computation by computer. Although Pant et al. noticed the problem and improved the reweighted function to avoid this problem, the constant ζ depends on experience. Actually, the proposed reweighted function can avoid the two problems; therefore, the proposed reweighted function can play a better effect.

2.3. Representation of Solution Set in Nullspace Measurement Matrix. As for $\mathbf{y} = \mathbf{\Phi x}$, it is well known that all of its solutions can be parameterized as

$$\mathbf{x} = \mathbf{x}_s + \mathbf{V}\boldsymbol{\xi} \tag{17}$$

where $\mathbf{x}_s$ denotes particular solution of system equation, which can be described as $\mathbf{x}_s = \mathbf{\Phi}^T(\mathbf{\Phi}\mathbf{\Phi}^T)^{-1}\mathbf{y}$. $\mathbf{V} \in \mathbb{R}^{n\times(n-m)}$ is a matrix, and its column vectors $\mathbf{v}_i$, $i = 1, 2, \ldots, n-m$, are made by orthonormal basis of nullspace measurement matrix $\mathbf{\Phi}$. $\mathbf{V}$ can be evaluated by singular-value decomposition or, more efficiently, QR decomposition of $\mathbf{\Phi}$ and $\boldsymbol{\xi} \in \mathbb{R}^{n-m}$. Adopting the solution can diminish the constraint of the optimization and the dimensions of space of solution are reduced from n to $n - m$, thereby reducing the complexity of computation.

3. New Algorithm for CS: RRSL0

3.1. RRSL0 Algorithm and Its Steps. As explained above, the proposed objective function can be eventually described as

$$\arg\min_{\boldsymbol{\xi}\in\mathbb{R}^{n-m}} \quad \frac{1}{2}\left\|\mathbf{y} - \mathbf{\Phi}\left(\mathbf{x_s} + \mathbf{V}\boldsymbol{\xi}\right)\right\|_2^2 + \lambda\mathbf{W}^T H_\sigma\left(\mathbf{x}_s + \mathbf{V}\boldsymbol{\xi}\right), \tag{18}$$

where $H_\sigma(\mathbf{x}_s + \mathbf{V}\boldsymbol{\xi}) = (c\cdot(\mathbf{x}_s + \mathbf{V}\boldsymbol{\xi})^2)/(c\cdot(\mathbf{x}_s + \mathbf{V}\boldsymbol{\xi})^2 + \sigma^2)$ is a differentiable smoothed accumulated function. Reweighted function $\mathbf{W} = 1/e^{(\mathbf{x}_s+\mathbf{V}\boldsymbol{\xi})^2}$. Let $\mathbf{x}_s = [x_s(1), x_s(2), \ldots, x_s(n)]^T$, $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_{n-m}]^T$; therefore, we have

$$\mathbf{G} = \frac{\partial H_\sigma\left(\mathbf{x}_s + \mathbf{V}\boldsymbol{\xi}\right)}{\partial\boldsymbol{\xi}} = \mathbf{V}^T\frac{2c\sigma^2\left(\mathbf{x}_s + \mathbf{V}\boldsymbol{\xi}\right)}{\left(c\left(\mathbf{x}_s + \mathbf{V}\boldsymbol{\xi}\right)^2 + \sigma^2\right)^2} \tag{19}$$

$$\mathbf{U} = \frac{\partial\mathbf{G}}{\partial\boldsymbol{\xi}} = \mathbf{V}^T\frac{2c\sigma^4 - 6c^2\sigma^2\left(\mathbf{x}_s + \mathbf{V}\boldsymbol{\xi}\right)}{\left(c\left(\mathbf{x}_s + \mathbf{V}\boldsymbol{\xi}\right)^2 + \sigma^2\right)^3}\mathbf{V}. \tag{20}$$

According to (19), the gradient of the proposed objective function is written as

$$\mathbf{g} = \mathbf{V}^T\left(\mathbf{\Phi}^T\left(\mathbf{y} - \mathbf{\Phi}\left(\mathbf{x}_s + \mathbf{V}\boldsymbol{\xi}\right)\right) + \lambda\left(\frac{1}{e^{(\mathbf{x}_s+\mathbf{V}\boldsymbol{\xi})^2}}\right)^T\frac{2c\sigma^2\left(\mathbf{x}_s + \mathbf{V}\boldsymbol{\xi}\right)}{\left(c\left(\mathbf{x}_s + \mathbf{V}\boldsymbol{\xi}\right)^2 + \sigma^2\right)^2}\right). \tag{21}$$

Combining (20) and (21), the second derivative is expressed as

$$\mathbf{H} = \frac{\partial\mathbf{g}}{\partial\boldsymbol{\xi}} = \mathbf{V}^T\left(\mathbf{\Phi}^T\mathbf{\Phi} + \lambda\mathbf{W}^T\mathbf{U}\right)\mathbf{V}. \tag{22}$$

Solving problem of ULSE is to solve the optimization problem in (18). As for this problem, there are many methods, such split Bregman methods [25–27], FISTA [28], alternating direction methods [29], and gradient descent [30]. In order to reduce the computational complexity, this paper adopts conjugate gradient (CG) method to optimize the proposed objective function. There are two classical methods in CG: FR conjugate gradient (FR-CG) method [31] and PRP conjugate gradient (PRP-CG) method [32]. This paper combines FR-CG with PRP-CG, forming a mixed CG algorithm; here we called it MCG method. Based on this, the MCG method is employed to solve the optimization problem in this paper.

The problem firstly can be solved by using a sequential σ − *continuation* strategy as detailed in the next paragraph.

Given σ a small target value $\sigma_{\min}$, and a sufficiently large initial value $\sigma_{\max}$, after referring to the annealing mechanism in simulated annealing [33], this paper proposes a monotonically decreasing sequence $\{\sigma_t \mid t = 2, 3, \ldots, T\}$, which is generated as

$$\sigma_t = \sigma_{\max}a^{-\gamma(t-1)}, \quad \text{s.t. } t = 1, 2, 3, \ldots, T. \tag{23}$$

where $\gamma = \log_a(\sigma_{\max}/\sigma_{\min})/(T-1)$, a can be a constant that is larger than 1, and T is the maximum of iterations. Using such a monotonically decreasing sequence can avoid the case of too small σ leading to local optimum.

According to CG algorithm, the solution $\mathbf{x}$ is updated as

$$\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} + \alpha^{(t)}\mathbf{d}^{(t)}, \tag{24}$$

where the parameter $\mathbf{d}^{(t)}$ can be given by

$$\mathbf{d}^{(t)} = \begin{cases} -\mathbf{g}^{(t)}, & t = 1 \\ -\mathbf{g}^{(t)} + \beta^{(t-1)}\mathbf{d}^{(t-1)}, & t \geq 2 \end{cases} \tag{25}$$

the parameter $\beta^{(t-1)}$ is given as

$$\beta^{(t-1)} = \begin{cases} \beta_{PRP}^{(t-1)}, & 0 < \beta_{PRP}^{(t-1)} < \beta_{FR}^{(t-1)}, \\ \beta_{FR}^{(t-1)}, & else. \end{cases} \tag{26}$$

where $\beta_{FR}^{(t-1)} = \|\mathbf{g}_t\|_2^2/\|\mathbf{g}_{t-1}\|_2^2$ and $\beta_{PRP}^{(t-1)} = \mathbf{g}_t^T(\mathbf{g}_t - \mathbf{g}_{t-1})/\|\mathbf{g}_{t-1}\|_2^2$, and the parameter $\alpha^{(t)}$ is updated as

$$\alpha^{(t)} = \frac{\left\|\mathbf{g}^{(t)}\right\|_2^2}{\mathbf{d}^{(t)T}\mathbf{H}^{(t)}\mathbf{d}^{(t)}} \tag{27}$$

Initialization:$\mathbf{\Phi}, \mathbf{x}, \beta, \sigma_{\max}, \sigma_{\min}, T$ and λ
Step 1: Set $\boldsymbol{\xi}^{(0)} = 0, \mathbf{W}^{(0)} = \mathbf{I}_n, \sigma_1 = \sigma_{\max}, t = 0$;
Step 2: QR decomposition $\mathbf{\Phi}^{\mathbf{T}} = \mathbf{QR}$, the $\mathbf{V}$ equals to the $n - m$ columns of $\mathbf{Q}$, so the initial value $\mathbf{x}^{(0)} = \mathbf{x}_s + \mathbf{V}\boldsymbol{\xi}^{(0)}$;
Step 3: For $t = 1, 2, \ldots, T$
(1) Set $\sigma = \sigma_t$, compute σ_t by equation (23);
(2) Set $s = 0$, and iterative termination threshold e;
(3) While Res $> e$
(a) Compute $\boldsymbol{\xi}^{(s+1)}$ by equations (24), (25), (26), (27), (28);
(b) Compute Res $= \|\alpha^{(s)}\mathbf{d}^{(s)}\|_2^2$;
(4) Let $\boldsymbol{\xi}^{(t)} = \boldsymbol{\xi}^{(s+1)}$, and compute $\mathbf{x}^{(t)} = \mathbf{x}_s + \mathbf{V}\boldsymbol{\xi}^{(t)}$;
Step 4: Output $\mathbf{x} = \mathbf{x}^{(t)}$.

Algorithm 1: Regularized reweighted smoothed L_0-norm minimization (RRSL0) algorithm using MCG method.

From above equations, we can see that $\alpha^{(t)}$ is positive if $\mathbf{H}^{(t)}$ is positive definite (PD). As shown in (22), $\mathbf{\Phi}^T\mathbf{\Phi}$ is PD, and $\mathbf{W}$ is PD, so $\mathbf{H}^{(t)}$ is PD if $\mathbf{U}^{(t)}$ is PD. To get the PD of $\mathbf{U}^{(t)}$, this paper makes the following corrections to the diagonal elements of matrix $\mathbf{U}^{(t)}$:

$$u_i = \begin{cases} u_i & u_i > \eta, \\ \eta & u_i \leqslant \eta. \end{cases} \tag{28}$$

where η is a small positive constant with a value of 10^{-5}. By this processing in (28), the matrix $\mathbf{U}^{(t)}$ is PD, $\mathbf{H}^{(t)}$ is PD as well, and hence the optimization direction of each component of $\mathbf{x}$ keeps consistency.

The above is the optimization process analysis of MCG method. For MCG method, the direction of each iteration of this method is the combination of the last iteration direction and the negative gradient direction, which can overcome the sawtooth phenomenon caused by the steepest descent method.

According to the explanation above, we can conclude the steps of proposed RRSL0 algorithm, which is given in Algorithm 1. As for σ, it can be shown that function $H_\sigma(\mathbf{x})$ remains convex in the region where the largest magnitude of the component of $\mathbf{x} = \mathbf{x}_s + \mathbf{V}\boldsymbol{\xi}$ is less than σ. As the algorithm starts at the original value $\boldsymbol{\xi}^{(0)} = \mathbf{0}$, the above choice of σ_1 ensures that the optimization starts in a convex region. This greatly facilitates the convergence of the RRSL0 algorithm.

3.2. Selection of Parameters. The selection of parameters λ and σ will affect the performance of the RRSL0 algorithm; thus this paper discusses the selection of the two above parameters in this section.

3.2.1. Selection of Parameter λ. The choice of parameter λ is closely related to the noise level. If the noise level is known, the *discrepancy principle* (DP) can be used to determine the parameter λ. However, in practice, the noise level cannot be measured accurately. In this case, it is necessary to use *data – driven principle* to select parameter λ, such as *BIC principle*. Ref [34], etc. prove the rationality of *BIC principle* under appropriate conditions, so this paper uses this *principle* to determine λ. The *BIC principle* is given by

$$BIC\{\lambda_k\} := \frac{1}{2}\left\|\mathbf{\Phi}\mathbf{x} - \mathbf{y}\right\|_2^2 + \frac{\ln n}{m}\mathbf{W}F_\sigma(\mathbf{x}). \tag{29}$$

Then we determine λ by $\arg\min_{\lambda_k} BIC\{\lambda_k\}$.

3.2.2. Selection of Parameter σ. According to (23), descending sequence of σ is generated by $\sigma_t = \sigma_{\max}(\sigma_{\min}/\sigma_{\max})^{(t-1)/(T-1)}$ (it is obtained through simplification of (23)). Parameter $\sigma_{\min}$ and parameter $\sigma_{\max}$ should be appropriately selected. The selection of $\sigma_{\min}$ and $\sigma_{\max}$ is discussed below.

For initial σ value $\sigma_{\max}$, here let $\widetilde{x} = \max\{|\mathbf{x}|\}$; in order to make the algorithm converge quickly, let parameter $\sigma_{\max}$ satisfies

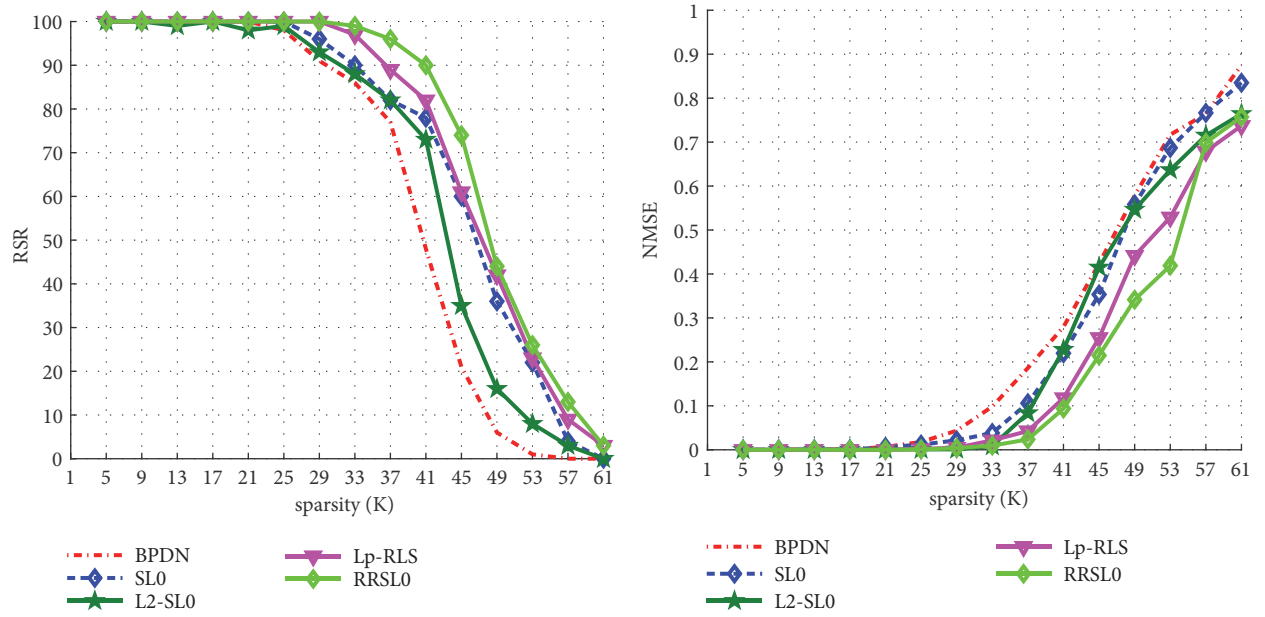
$$\begin{aligned} H_\sigma(\widetilde{x}) = \frac{c\widetilde{x}^2}{c\widetilde{x}^2 + \sigma_{\max}^2} \leq b \Longrightarrow \\ \sigma_{\max} \geq \left(\sqrt{\frac{1-b}{b}c}\right)\widetilde{x}. \end{aligned} \tag{30}$$

From the equation, we can see constant b satisfies $(1 - b)/b \geq 0$; thus $0 \leq b \leq 1$; here we define constant b as 0.5. Hence $\sigma_{\max} = \sqrt{c}\max\{|\mathbf{x}|\}$.

As for final value $\sigma_{\min}$, when $\sigma_{\min} \longrightarrow 0$, $H_{\sigma_{\min}}(\mathbf{x}) \longrightarrow \|\mathbf{x}\|_0$. That is, the smaller $\sigma_{\min}$, the more $H_{\sigma_{\min}}(\mathbf{x})$ can reflect the sparsity of signal $\mathbf{x}$, but at the same time it is also more sensitive to noise; therefore, the value $\sigma_{\min}$ should not be too small. Combining [19], we choose $\sigma_{\min} = 0.01$.

4. Numerical Simulation and Analysis

The numerical simulation platform is MATLAB 2017b, which is installed on the WINDOWS 10, 64-bit operating system. The CPU of simulation computer is Intel (R) Core (TM) i5-3230M, and the frequency is 2.6 GHz. In this section, the performance of RRSL0 algorithm is verified by signal and image recovery in the noiseless case and noise case.



(a) RSR of recovery signal

(b) NMSE of recovery signal

Figure 2: Signal recovery in noiseless case. RSR and NMSE of recovery signal are displayed at intervals of 5 with sparsity from 1 to 61. The comparison algorithms are BPDN, SL0, L2-SL0, Lp-RLS, and RRSL0. The experimental results were obtained by 100 independent repeated experiments.

Table 1: Signal CRT analysis for BPDN, SL0, L2-SL0, L_p-RLS, and proposed RRSL0 with signal length changes according to sequence [170, 220, 270, 320, 370, 420, 470, 520] while 100 runs.

Signal length (n)	CPU running time (seconds)				
	BPDN	SL0	L2-SL0	Lp-RLS	RRSL0
170	0.1958	0.0572	0.0909	0.0636	0.0628
220	0.2899	0.1385	0.2298	0.1503	0.1425
270	0.4953	0.2285	0.4257	0.3050	0.2908
320	0.7675	0.3196	0.6385	0.5121	0.5088
370	1.0586	0.4559	0.9260	0.9023	0.8924
420	1.4767	0.6133	1.1327	1.0906	1.0170
470	1.9414	0.7956	1.4782	1.4178	1.3438
520	2.6194	1.0383	2.0894	1.9097	1.8818

Here some state-of-the-art algorithms are selected for comparison. The parameters are selected to obtain best performance for each algorithm: for BPDN algorithm [35], $\lambda = \sigma_N\sqrt{2\log(n)}$; for SL0 algorithm, $\delta_{\max} = 2\max\{|\mathbf{x}|\}$, $\delta_{\min} = 0.01$, and scale factors are set as $L = 5$, $\rho = 0.8$; for L2-SL0 algorithm, $\delta_{\max} = 4\max\{\mathbf{x}\}$, $\delta_{\min} = 0.01$, $L = 10$, and $\rho = 0.8$; for L_p-RLS algorithm, $T = 80$, $p_1 = 1$, $p_T = 0.1$, $\epsilon_1 = 1$, $\epsilon_T = 0.01$, and $E_t = 10^{-25}$; and for RRSL0 algorithm, smoothed factor $\sigma_{\max} = \sqrt{c}\max\{|\mathbf{x}|\}$, $\sigma_{\min} = 0.01$, and iterations $T = 80$. All experiments are based on the 100 trials.

4.1. Comparison of Reconstruction Performance. For signal recovery under no noise conditions, we evaluate performance of algorithms by *reconstruction success rate* (RSR), *normlized mean square error* (NMSE), and *cpu running time* (CRT). RSR is defined as the ratio of the number of successful experiments to the total number of experiments, and if $\|x - \hat{x}\|_2/\|x\|_2 < 10^{-3}$, the recovery is considered to be a success. NMSE is defined as $\|x - \hat{x}\|_2/\|x\|_2$. CRT is measured with *tic* and *toc*. The case of signal recovery under noise is considered by NMSE and *signal to noise ratio* (SNR). SNR is defined as $20\log(\|x - \hat{x}\|_2/\|x\|_2)$.

4.1.1. Signal Recovery in Noiseless Case. In this section, the SL0, BPDN, L2-SL0, and Lp-RLS are used for comparison. Here we fix $n = 256$ and $m = 100$ and the sparsity $K = 4q + 1$, $q = 1, 2, \ldots, 15$ for Figure 2 and let $n = [170, 220, 270, 320, 370, 420, 470, 520]$, $m = n/2$, $K = n/5$ for Table 1. For every experiment, we generate a pair of $\{\mathbf{x}, \mathbf{\Phi}\}$: $\mathbf{\Phi}$ is a $m \times n$ random Gauss matrix with normalized rows; the *nonzero* entries of the sparse signal $\mathbf{x} \in \mathbb{R}^n$

are *i.i.d.* generated according to the Gaussian distribution $\mathcal{N}(0, 1)$.

Figure 2(a) shows the RSR of all algorithms. When K is large enough and continues to increase, the RSR of all algorithms tends to decrease until they drop to *zero*. In fact, the speed of changes in the RSR reflects the *robustness* of the algorithm. Specifically, the slower the RSR changes, the better the *robustness* of the algorithm. Obviously, the RRSL0 algorithm has a more slowly changing trend than other comparison algorithms. By comparison, SL0 is superior to BPDN and L2-SL0, but inferior to L_p-RLS and RRSL0. BPDN, L2-SL0, L_p-RLS, and RRSL0 are the unconstrained optimization algorithms. When recovering signal in noiseless case, the deviation term $\|\mathbf{\Phi x} - \mathbf{y}\|_2^2$ is equivalent to the error between original signal and recovered signal, which interferes with the *robustness* of the algorithm. Luckily, the L_p-RLS adopts FR-CG algorithm to optimize, which improves the stability of the optimized solution. The proposed RRSL0 algorithm applies MCG algorithm to enhance the reconstruction performance.

The NMSE of all algorithms is shown in Figure 2(b). NMSE reflects the signal reconstruction accuracy, and the lower the NMSE, the higher the accuracy of the reconstructed signal. As shown in this figure, all algorithms perform well when the sparsity is lower than 29. NMSE of BDPN increases fastest, followed by L2-SL0, SL0, L_p-RLS, and proposed RRSL0. When sparsity is increasing, L2-SL0 does not perform well, while L_p-RLS and RRSL0 perform well. This result indirectly verified that MCG and FR-CG method are superior to steepest descent method in SL0 and L2-SL0; furthermore, the term $\|\mathbf{\Phi x} - \mathbf{y}\|_2^2$ has a relatively large influence on the recovery results. In addition, the experiment also verified that RRSL0 performs better than L_p-RLS and other algorithms.

Table 1 shows the CRT of all algorithms. The n changes according to a given sequence [170, 220, 270, 320, 370, 420, 470, 520]. From the table, for any n, SL0 has the shortest computation time, followed by RRSL0, L_p-RLS, and L2-SL0, and BPDN takes the longest computation time. BPDN algorithm is generally implemented by quadratic programming method, and the computational complexity of this method is very high, thus resulting in a large increase in the overall computation time of the algorithm. Furthermore, In L2-SL0, the step factor needs to be obtained by a one-dimensional linear search method, while L_p-RLS and RRSL0 do not. Compared with L_p-RLS, RRSL0 is more prominent in the decrease of computation time. In order to verify the difference in computation time between the two algorithms, the convergence comparison of the two algorithms is shown in Figure 3. From this figure, we can draw that the convergence rate of RRSL0 is much faster than L_p-RLS.

4.1.2. Signal Recovery in Noise Case. In this section, we discuss signal recovery performance in noise case. We add noise $\mathbf{w}$ to the measurement vector $\mathbf{y}$; moreover, $\mathbf{w}$ is randomly formed and follows the Gauss distribution of $\delta_N \times \mathcal{N}(0, 1)$. In order to analyze the antinoise performance of the RRSL0 algorithm more closely to the real situation, we constructed a certain signal as an experimental object in the experiments in this section. The certain signal is given by

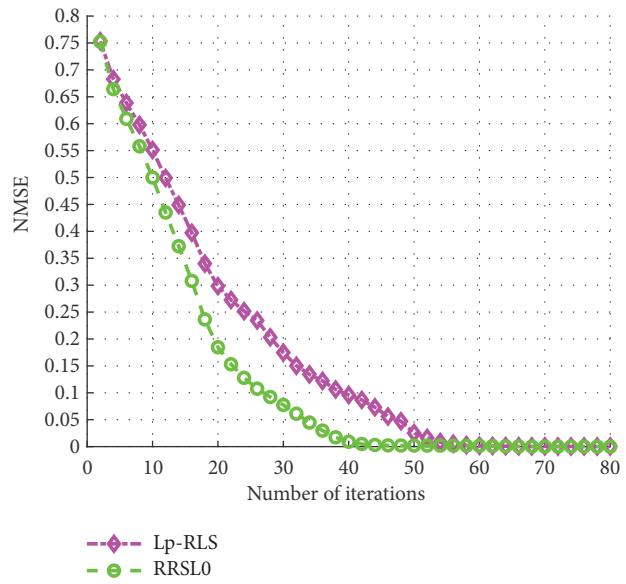


Figure 3: Convergence verification for L_p-RLS and proposed RRSL0 with signal length $n = 256$, $m = 100$, and $K = 20$ with 100 runs.

$$\begin{aligned}\mathcal{X} = {} & \alpha_1 \sin\left(2\pi f_1 T_s \mathbf{t}\right) + \beta_1 \cos\left(2\pi f_2 T_s \mathbf{t}\right) \\ & + \alpha_2 \sin\left(2\pi f_3 T_s \mathbf{t}\right) + \beta_2 \cos\left(2\pi f_4 T_s \mathbf{t}\right)\end{aligned} \tag{31}$$

where $\alpha_1 = 0.2$, $\alpha_2 = 0.1$, $\beta_1 = 0.3$, and $\beta_2 = 0.4$. $f_1 = 50$Hz, $f_2 = 100$Hz, $f_3 = 200$Hz, and $f_4 = 300$Hz. Here $\mathbf{t}$ is a sequence with $\mathbf{t} = [1, 2, 3, \ldots, n]$, T_s is sampling interval valued $1/f_s$. f_s is sampling frequency valued 800Hz. The object that needs to be reconstructed can be expressed as

$$\begin{aligned}\mathcal{Y} &= \boldsymbol{\psi}\boldsymbol{\varphi}\mathbb{X} + \{\delta_N \times \mathcal{N}(0, 1)\}_{m\times 1} \\ &= \mathbf{\Phi}\mathbb{X} + \{\delta_N \times \mathcal{N}(0, 1)\}_{m\times 1}.\end{aligned} \tag{32}$$

where $\mathcal{X} = \boldsymbol{\varphi}\mathbb{X}$, $\boldsymbol{\varphi}^T$ is the *Fourier transform* basis (here called sparse basis matrix) given by *Fourier*$\{\mathbf{I}_{n\times n}\}$, $\mathbf{I}_{n\times n}$ is a unit matrix, and $\mathbb{X} \in \mathbb{R}^n$ is a sparse signal in frequency domain obtained by the *Fourier transform* of $\mathcal{X}$. Moreover, $\boldsymbol{\psi} \in \mathbb{R}^{m\times n}$ is a random matrix generated by Gauss distribution, sensing matrix $\mathbf{\Phi} = \boldsymbol{\psi}\boldsymbol{\varphi}$. $\mathcal{Y} \in \mathbb{R}^m$. Here, let $n = 128$ and $m = 64$. Therefore, the sparse signal $\mathbb{X}$ can be recovered from given $\{\mathcal{Y}, \mathbf{\Phi}\}$ by CS recovery methods, and then the original signal $\mathcal{X}$ can be obtained by $\mathcal{X} = \boldsymbol{\varphi}\mathbb{X}$.

Figure 4 shows the signal recovery effect by different algorithms under noise intensity $\delta_N = 0.2$. Meanwhile, time-frequency characteristics are shown in Figure 5. Obviously, BPDN and SL0 perform not well while L2-SL0, L_p-RLS, and proposed RRSL0 perform quite well. This verifies that the regularization mechanism has good antinoise effect. The performance of each algorithm under different noise intensities is shown in Table 2. When $\delta_N = 0$, the SL0 outperforms other algorithms, but with the increase of δ_N, the effect of SL0 is getting worse and worse. This result further illustrates that the traditional constrained sparse recovery algorithm does not have the performance of antinoise. For BPDN, L2-SL0, L_p-RLS, and RRSL0, they all applied regularization mechanism,

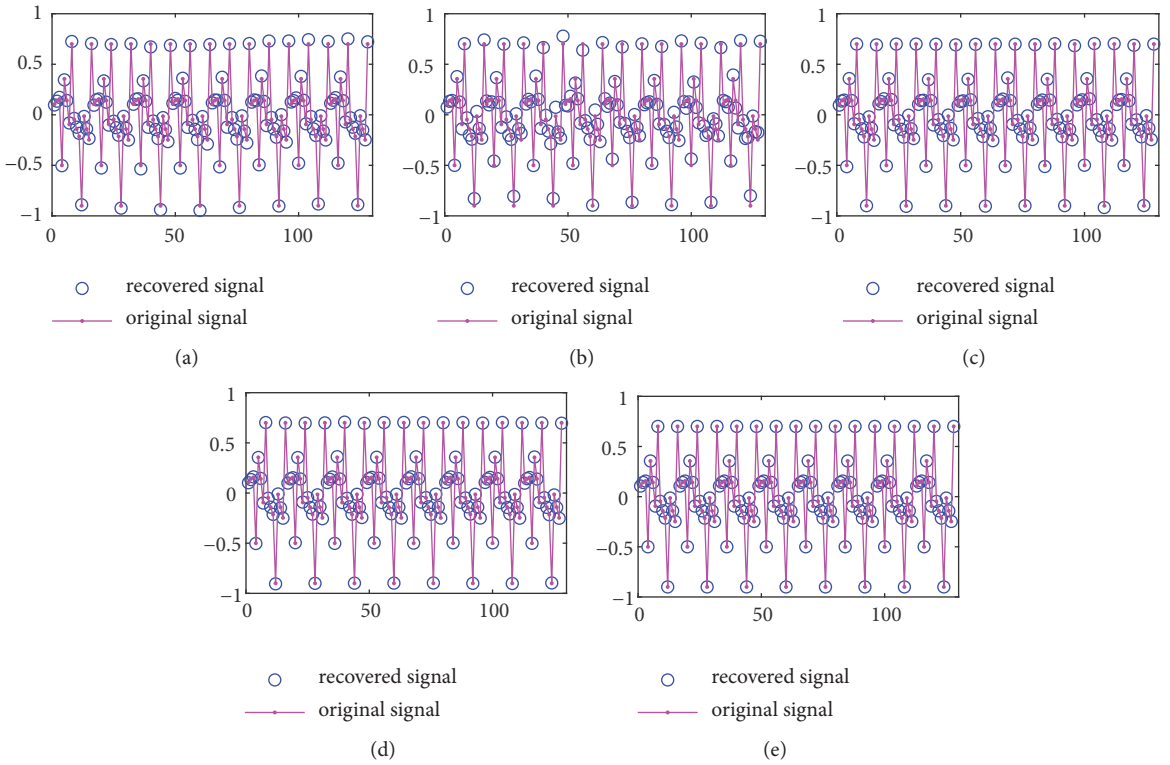


FIGURE 4: Signal recovery effect by different algorithms when noise intensity $\delta_N = 0.2$. (a) signal recovery by the BPDN algorithm; (b) signal recovery by the SL0 algorithm; (c) signal recovery by the L2-SL0 algorithm; (d) signal recovery by the L_p-RLS algorithm; (e) signal recovery by the RRSL0 algorithm.

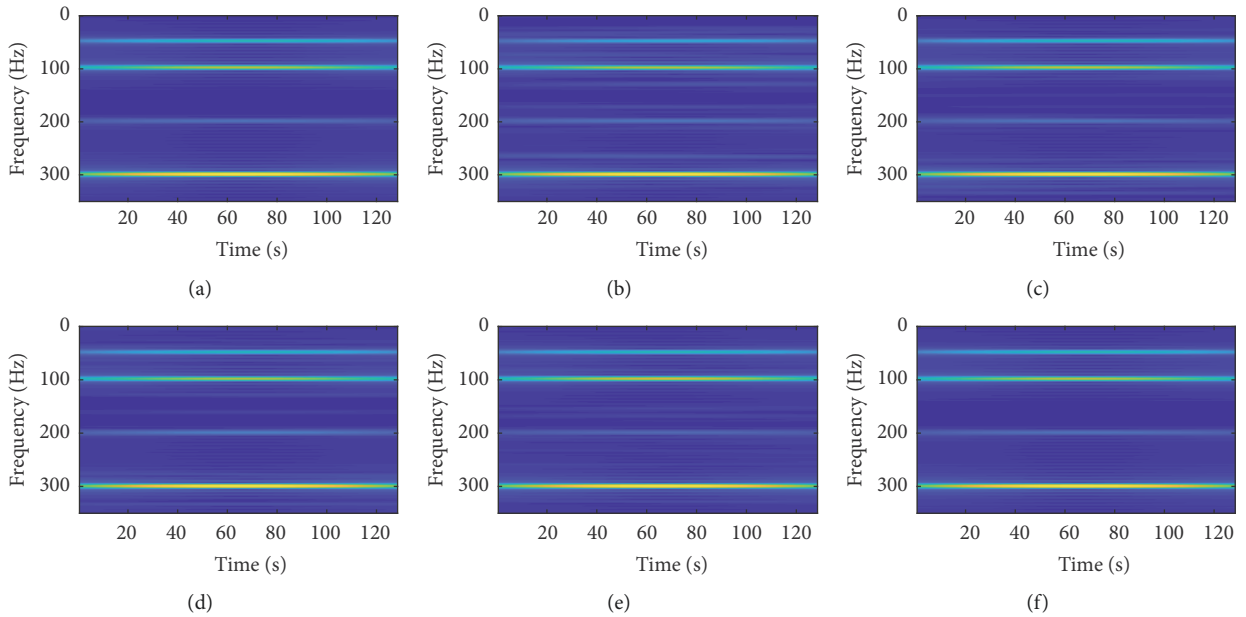


FIGURE 5: Reconstruction time-frequency characteristics of different algorithms when noise intensity $\delta_N = 0.2$. (a) time-frequency characteristics of the original signal; (b) BPDN; (c) SL0; (d) L2-SL0; (e) L_p-RLS; (f) RRSL0.

Table 2: NMSE analysis for BPDN, SL0, L2-SL0, L_p-RLS, and proposed RRSL0 with noise intensity changes according to sequence δ_N = [0, 0.01, 0.05, 0.1, 0.2, 0.5] while 100 runs.

Noise intensity (δ_N)	NMSE of signal recovery				
	BPDN	L0	L2-SL0	Lp-RLS	RRSL0
0	2.553e-05	1.311e-07	4.563e-07	1.563e-07	2.338e-07
0.01	4.659e-04	3.054e-04	2.352e-04	7.930e-05	6.032e-05
0.05	1.607e-03	4.079e-03	9.012e-04	2.384e-04	1.150e-04
0.1	7.633e-03	8.424e-03	3.309e-03	7.793e-04	5.522e-04
0.2	1.381e-02	2.090e-02	6.091e-03	1.130e-03	8.892e-04
0.5	3.063e-02	6.908e-02	1.083e-02	5.743e-03	3.041e-03

(a) Original Boat (b) Original Barbara

Figure 6: Original images.

and they are indeed superior to SL0 in noise case. Therefore, the proposed RRSL0 in this paper has the best antinoise performance.

4.2. Comparison of Image Recovery Performance. Real images are considered to be approximately sparse under some proper basis, such as the Discrete Cosine Transform (DCT) basis and Discrete Wavelet Transform (DWT) basis. Here we choose DWT basis to recover these images. We compare the recovery performances based on the 2 real images in Figure 6: Boat and Barbara, and the size of these images is 256×256. The compression ratio (CR, it is defined as m/n) is 0.5, and noise δ_N equals 0.01. We still choose SL0, BPDN, L2-SL0, and L_p-RLS to make comparisons. For image recovery, the object of image processing is given by

$$\mathbf{Y} = \mathbf{\Phi X} + \mathbf{b} \tag{33}$$

Here $\mathbf{Y}$, $\mathbf{X}$, and $\mathbf{b}$ are matrices; among these, $\mathbf{Y}, \mathbf{b} \in \mathbb{R}^{m\times n}$, $\mathbf{X} \in \mathbb{R}^{n\times n}$. In order to meet the basic requirements of CS, we make the following processing:

$$\mathbf{Y} = [\mathbf{Y}_1, \mathbf{Y}_2, \ldots, \mathbf{Y}_n] \tag{34}$$

where $\mathbf{Y}_i$, $i = 1, 2, \ldots, n$, is the column vector of $\mathbf{Y}$. Therefore we can process images by CS.

For performance of image recovery, we valuate it by *Peak Signal to Noise Ratio* (PSNR) and *Structural Similarity Index* (SSIM). PSNR is defined as

$$\text{PSNR} = 10 \log\left(\frac{255^2}{\text{MSE}}\right) \tag{35}$$

where $\text{MSE} = \|x - \hat{x}\|_2^2$, and SSIM is defined as

$$\text{SSIM}(p, q) = \frac{(2\mu_p + \mu_q + c_1)(2\sigma_{pq} + c_2)}{(\mu_p^2 + \mu_q^2 + c_1)(\sigma_p^2 + \sigma_q^2 + c_2)}. \tag{36}$$

Among this, μ_p is the mean of image p, μ_q is the mean of image q, σ_p is the variance of image p, σ_q is the variance of image q, and σ_{pq} is covariance between image p and image q. Parameters $c_1 = z_1 L$ and $c_2 = z_2 L$, where $z_1 = 0.01$, $z_2 = 0.03$, and L is dynamic range of pixel values. Range of SSIM is $[-1, 1]$; when these two images are the same, the SSIM equals to 1.

Figure 6 shows the original images. Figure 7 shows the recovery effect of *Boat* and *Barbara* with noise intensity $\delta_N = 0.01$ and $\delta_N = 0.1$. In the *Boat*, the recovered images by SL0 and BPDN have the obvious water ripples while recovered images by other algorithms have no such water ripples. Similarly, for *Barbara*, recovered images by SL0 and BPDN are blurred compared with recovered images by other algorithms. L2-SL0, L_p-RLS, and RRSL0 algorithms are also effective for noisy image recovery. For L2-SL0, L_p-RLS, and RRSL0 algorithms, their recovery effects are very similar. In order to further analyze the advantages and disadvantages of these algorithms, we analyze PSNR and SSIM of the images recovered by these algorithms and results are shown in Tables 3–6. By observation and analysis, L_p-RLS performs better than L2-SL0 on the whole (except SSIM when $\delta_N = 0.05$, because the robustness of L_p-RLS is not strong); at the same time, RRSL0 outperforms L_p-RLS. Hence, the RRSL0 proposed by this paper is superior to other selected algorithms in image processing.

(a) Recovered Boat with noise intensity $\delta_N = 0.01$

(b) Recovered Barbara with noise intensity $\delta_N = 0.01$

(c) Recovered Boat with noise intensity $\delta_N = 0.1$

(d) Recovered Barbara with noise intensity $\delta_N = 0.1$

Figure 7: Images recovery effect by BPDN, SL0, L2-SL0, L_p-RLS, and RRSL0 algorithms under different noise intensities. In subfigures (a), (b), (c), and (d), from left to right are images recovered by BPDN, SL0, L2-SL0, L_p-RLS, and RRSL0 algorithms.

Table 3: PSNR analysis of recovered Boat image by BPDN, SL0, L2-SL0, L_p-RLS, and RRSL0 with noise intensity changes according to sequence $\delta_N = [0, 0.01, 0.05, 0.1, 0.2, 0.5]$.

Noise intensity (δ_N)	PSNR of Boat image recovery				
	BPDN	SL0	L2-SL0	L_p-RLS	RRSL0
0	28.4928	31.5901	31.7175	32.2332	33.3495
0.01	27.9775	29.8589	30.2423	30.9742	31.5447
0.05	24.0531	27.1226	27.7101	28.2422	29.2505
0.1	20.3769	21.6485	23.9331	25.8639	26.4745
0.2	16.3044	17.9303	20.6744	23.0823	23.7415
0.5	9.2283	10.7863	13.4863	14.4974	16.3646

Table 4: SSIM analysis of recovered Boat image by BPDN, SL0, L2-SL0, L_p-RLS, and RRSL0 with noise intensity changes according to sequence $\delta_N = [0, 0.01, 0.05, 0.1, 0.2, 0.5]$.

Noise intensity (δ_N)	SSIM of Boat image recovery				
	BPDN	SL0	L2-SL0	L_p-RLS	RRSL0
0	0.9865	0.9901	0.9902	0.9914	0.9934
0.01	0.9764	0.9852	0.9865	0.9887	0.9898
0.05	0.9418	0.9714	0.9756	0.9789	0.9827
0.1	0.8827	0.9083	0.9403	0.9631	0.9669
0.2	0.7452	0.8077	0.8868	0.9284	0.9396
0.5	0.3571	0.4427	0.5966	0.6492	0.7351

Table 5: PSNR analysis of recovered Barbara image by BPDN, SL0, L2-SL0, L_p-RLS, and RRSL0 with noise intensity changes according to sequence $\delta_N = [0, 0.01, 0.05, 0.1, 0.2, 0.5]$.

Noise intensity (δ_N)	PSNR of Barbara image recovery				
	BPDN	SL0	L2-SL0	L_p-RLS	RRSL0
0	28.6613	29.5761	31.7921	32.3566	33.4074
0.01	27.7887	30.4746	30.9875	31.7365	32.6767
0.05	26.0316	26.1561	26.3147	27.3533	29.8613
0.1	20.7076	21.6914	23.8343	26.9408	27.5167
0.2	16.6564	16.6721	19.1297	19.2624	19.7801
0.5	9.1316	9.3525	13.1838	15.7972	16.9070

Table 6: SSIM analysis of recovered Barbara image by BPDN, SL0, L2-SL0, L_p-RLS, and RRSL0 with noise intensity changes according to sequence $\delta_N = [0, 0.01, 0.05, 0.1, 0.2, 0.5]$.

Noise intensity (δ_N)	SSIM of Barbara image recovery				
	BPDN	SL0	L2-SL0	L_p-RLS	RRSL0
0	0.9834	0.9920	0.9926	0.9936	0.9984
0.01	0.9817	0.9904	0.9914	0.9927	0.9942
0.05	0.9797	0.9744	0.9753	0.9739	0.9888
0.1	0.9161	0.9314	0.9544	0.9728	0.9810
0.2	0.8165	0.8170	0.9051	0.9142	0.9278
0.5	0.3863	0.3992	0.6309	0.7712	0.8190

5. Conclusions

In this paper, we propose the RRSL0 algorithm to recover sparse signal from given $\{\mathbf{y}, \mathbf{\Phi}\}$ in noise case. For the RRSL0 algorithm, it is constructed under the framework of *Tikhonov* regularization, in which the constraint condition $\|\mathbf{y} - \mathbf{\Phi}\mathbf{x}\|_2^2 \leq \epsilon$ is the deviation term, and the regularization term is replaced by a reweighted smoothed function $\mathbf{W}^T H_\sigma(\mathbf{x})$. As a key part of the RRSL0 algorithm, the reweighted smoothed function is to promote sparsity and provide guarantee for robust and accurate signal recovery. Furthermore, We consider the value of λ and the initial value $\sigma_{\max}$ and final value $\sigma_{\min}$. Combined with previous literatures, we used α-method to determine the value of λ and deduced the method of initial value and final value of σ. Sparse signal recovery experiments on both the simulated signal and real images show the proposed RRSL0 algorithm performs better than the L_1 or L_p regularization methods and classical L_0 regularization methods. In addition, we would also like to apply the proposed algorithm to other CS applications such as the RPCA [36, 37] and SAR imaging [38].

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare no conflicts of interest.

Authors' Contributions

All authors have made great contributions to the work. Jianhong Xiang, Huihui Yue, and Xiangjun Yin conceived and designed the experiments; Jianhong Xiang and Huihui Yue performed the experiments and analyzed the data; Xiangjun Yin and Linyu Wang gave insightful suggestions for the work; Jianhong Xiang, Huihui Yue, and Xiangjun Yin wrote the paper.

Acknowledgments

This paper is supported by the National Key Laboratory of Communication Anti-jamming Technology.

References

[1] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[2] E. J. Candes and M. B. Wakin, "An introduction to compressive sampling: a sensing/sampling paradigm that goes against the common knowledge in data acquisition," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21–30, 2008.

[3] E. J. Candès, "The restricted isometry property and its implications for compressed sensing," *Comptes Rendus Mathematique*, vol. 346, no. 9-10, pp. 589–592, 2008.

[4] E. J. Candes and T. Tao, "Decoding by linear programming," *IEEE Transactions on Information Theory*, vol. 51, no. 12, pp. 4203–4215, 2005.

[5] S. Huang and T. D. Tran, "Sparse signal recovery via generalized entropy functions minimization," *IEEE Transactions on Signal Processing*, vol. 67, no. 5, 2017.

[6] L. Wang, X. Yin, H. Yue, and J. Xiang, "A regularized weighted smoothed L0 norm minimization method for underdetermined blind source separation," *Sensors*, vol. 18, no. 12, p. 4260, 2018.

[7] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655–4666, 2007.

[8] D. L. Donoho, Y. Tsaig, I. Drori, and J. L. Starck, "Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 2, pp. 1094–1121, 2012.

[9] D. Needell and R. Vershynin, "Signal recovery from incomplete and inaccurate measurements via regularized orthogonal matching pursuit," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 310–316, 2010.

[10] D. Needell and J. A. Tropp, "CoSaMP: iterative signal recovery from incomplete and inaccurate samples," *Communications of the ACM*, vol. 53, no. 12, pp. 93–100, 2010.

[11] J. Wang, S. Kwon, and B. Shim, "Generalized orthogonal matching pursuit," *IEEE Transactions on Signal Processing*, vol. 60, no. 12, pp. 6202–6216, 2012.

[12] W. Dai and O. Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *IEEE Transactions on Information Theory*, vol. 55, no. 5, pp. 2230–2249, 2009.

[13] P. S. Huggins and S. W. Zucker, "Greedy basis pursuit," *IEEE Transactions on Signal Processing*, vol. 55, no. 7, part 2, pp. 3760–3772, 2007.

[14] X. X. Zhu and R. Bamler, "Tomographic SAR inversion by L1-norm regularization-the compressive sensing approach," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 10, pp. 3839–3846, 2010.

[15] T. Long, W. Jiao, and G. He, "RPC estimation via l1-norm-regularized least squares (L1LS)," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 8, pp. 4554–4567, 2015.

[16] S. Foucart and M.-J. Lai, "Sparsest solutions of underdetermined linear systems via ℓq-minimization for $0 < q \leq 1$," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 395–407, 2009.

[17] J. K. Pant, W. S. Lu, and A. Antoniou, "New improved algorithms for compressive sensing based on lp norm," *IEEE Transactions on Circuits and Systems II Express Briefs*, vol. 3, pp. 198–202, 2014.

[18] D. Wipf and S. Nagarajan, "Iterative reweighted ℓ 1and ℓ 2 methods for finding sparse solutions," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 317–329, 2010.

[19] H. Mohimani, M. Babaie-Zadeh, and C. Jutten, "A fast approach for overcomplete sparse decomposition based on smoothed l^0 norm," *IEEE Transactions on Signal Processing*, vol. 57, no. 1, pp. 289–301, 2009.

[20] X. Ye, W.-P. Zhu, A. Zhang, and J. Yan, "Sparse channel estimation of MIMO-OFDM systems with unconstrained smoothed l0-norm-regularized least squares compressed sensing," *EURASIP Journal on Wireless Communications and Networking*, vol. 2013, no. 1, p. 282, 2013.

[21] R. Zhao, W. Lin, H. Li, and S. Hu, "Reconstruction algorithm for compressive sensing based on smoothed l 0 norm and revised newton method," *Journal of Computer-Aided Design and Computer Graphics*, vol. 24, no. 4, pp. 478–484, 2012.

[22] C. Zhang, D. Hao, C. Hou, and X. Yin, "A new approach for sparse signal recovery in compressed sensing based on minimizing composite trigonometric function," *IEEE Access*, vol. 6, pp. 44894–44904, 2018.

[23] E. J. Candes, M. B. Wakin, and S. P. Boyd, "Enhancing sparsity by reweighted l1 minimization," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5-6, pp. 877–905, 2008.

[24] J. K. Pant, W.-S. Lu, and A. Antoniou, "Reconstruction of sparse signals by minimizing a re-weighted approximate l0-norm in the null space of the measurement matrix," in *Proceedings of the 53rd IEEE International Midwest Symposium on Circuits and Systems, MWSCAS 2010*, pp. 430–433, USA, August 2010.

[25] T. Goldstein and S. Osher, "The split Bregman methods for L1 regularized problems," *Society for Industrial and Applied Mathematics*, 2009.

[26] D. S. Smith, J. C. Gore, T. E. Yankeelov, and W. E. Brian, "Real-time compressive sensing mri reconstruction using gpu computing and split bregman methods," *International Journal of Biomedical Imaging*, vol. 2012, Article ID 864827, 6 pages, 2012.

[27] H. Nien and J. A. Fessler, "A convergence proof of the split bregman method for regularized least-squares problems," *Mathematics*, 2014.

[28] Y. Liu, Z. Zhan, J. Cai, D. Guo, Z. Chen, and X. Qu, "Projected iterative soft-thresholding algorithm for tight frames in compressed sensing magnetic resonance imaging," *IEEE Transactions on Medical Imaging*, vol. 35, no. 9, pp. 2130–2140, 2016.

[29] J. Yang and X. Yuan, "Linearized augmented lagrangian and alternating direction methods for nuclear norm minimization," *Mathematics of Computation*, vol. 82, no. 281, pp. 301–329, 2013.

[30] A. Antoniou and W. S. Lu, *Practical Optimization: Algorithms and Engineering Applications*, Springer, New York, NY, USA, 2007.

[31] C. Wang and M. Li, "Convergence property of the Fletcher-Reeves conjugate gradient method with errors," *Journal of Industrial and Management Optimization*, vol. 1, no. 2, pp. 193–200, 2005.

[32] Z. Wan, C. Hu, and Z. Yang, "A spectral PRP conjugate gradient methods for nonconvex optimization problem based on modified line search," *Discrete and Continuous Dynamical Systems - Series B*, vol. 16, no. 4, pp. 1157–1169, 2011.

[33] D. Bertsimas and J. Tsitsiklis, "Simulated annealing," *Statistical Science*, vol. 8, no. 1, pp. 10–15, 1993.

[34] H. Zhang, Y. L. Jiao, Y. Y. Shi et al., "Sparse recovery decay signals based on L0 regularization via PDASC," *Chinese Science: Information Science*, In press (in Chinese).

[35] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Review*, vol. 43, no. 1, pp. 129–159, 2001.

[36] E. J. Candes, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *Journal of the ACM*, vol. 58, no. 3, article no. 11, 2011.

[37] L. I. Shan-Shan, L. Chen, Y. X. Zhang, and Y. T. Yuan, "Fuzzy edge detection algorithm based on RPCA," *Computer Science*, 2018.

[38] J. Ni, Q. Zhang, Y. Luo, and L. Sun, "Compressed sensing SAR imaging based on centralized sparse representation," *IEEE Sensors Journal*, vol. 18, no. 12, pp. 4920–4932, 2018.