

# Sparse solution of underdetermined linear equations via adaptively iterative thresholding

Jinshan Zeng<sup>a,b</sup>, Shaobo Lin<sup>a</sup>, Zongben Xu<sup>a,\*</sup>

<sup>a</sup> School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an 710049, China

<sup>b</sup> Beijing Center for Mathematics and Information Interdisciplinary Sciences (BCMIS), Beijing 100048, China

## ARTICLE INFO

### Article history:

Received 28 May 2013

Received in revised form

24 October 2013

Accepted 28 October 2013

Available online 12 November 2013

### Keywords:

Iterative thresholding algorithm

Global convergence

Underdetermined linear equations

Sparse solution

## ABSTRACT

Finding the sparsset solution of an underdetermined system of linear equations  $y = Ax$  has attracted considerable attention in recent years. Among a large number of algorithms, iterative thresholding algorithms are recognized as one of the most efficient and important classes of algorithms. This is mainly due to their low computational complexities, especially for large scale applications. The aim of this paper is to provide guarantees on the global convergence of a wide class of iterative thresholding algorithms. Since the thresholds of the considered algorithms are set adaptively at each iteration, we call them adaptively iterative thresholding (AIT) algorithms. As the main result, we show that as long as  $A$  satisfies a certain coherence property, AIT algorithms can find the correct support set within finite iterations, and then converge to the original sparse solution exponentially fast once the correct support set has been identified. Meanwhile, we also demonstrate that AIT algorithms are robust to the algorithmic parameters. In addition, it should be pointed out that most of the existing iterative thresholding algorithms such as hard, soft, half and smoothly clipped absolute deviation (SCAD) algorithms are included in the class of AIT algorithms studied in this paper.

© 2013 Published by Elsevier B.V.

## 1. Introduction

Finding the sparsest solution of an underdetermined system of linear equations is an important problem emerged in many applications (especially, in compressed sensing [1,2]). Generally, an underdetermined system of linear equations can be described as

$$y = Ax, \quad (1.1)$$

where  $y \in \mathbf{R}^M$  and  $A \in \mathbf{R}^{M \times N}$  ( $M < N$ ) are known,  $x = (x_1, \dots, x_N)^T \in \mathbf{R}^N$  is unknown. Thus, finding the sparsest solution of the Eq. (1.1) can be mathematically modeled as the

following  $l_0$ -minimization, that is

$$\min_{x \in \mathbf{R}^N} \|x\|_0 \quad \text{s.t. } y = Ax, \quad (1.2)$$

where  $\|x\|_0$  denotes the number of the nonzero components of  $x$  and is formally called the  $l_0$ -norm. However, the problem (1.2) is NP-hard and generally intractable for computing.

Instead, there are mainly two classes of methods, that is, the greedy and relaxed methods for approximately solving the problem (1.2). The basic idea of the greedy method is that a sparse solution is refined iteratively by successively identifying one or more components that yield the greatest improvement in quality [3]. There are many commonly used greedy algorithms such as orthogonal matching pursuit (OMP) [4,5], stagewise OMP (StOMP) [6], regularized OMP (ROMP) [7], compressive sampling matching pursuit (CoSaMP) [8] and subspace pursuit [9]. The greedy algorithms can be quite fast, especially in the

\* Corresponding author. Tel.: +86 2982668005.

E-mail addresses: [jsh.zeng@gmail.com](mailto:jsh.zeng@gmail.com) (J. Zeng), [sblin1983@gmail.com](mailto:sblin1983@gmail.com) (S. Lin), [zbxu@mail.xjtu.edu.cn](mailto:zbxu@mail.xjtu.edu.cn) (Z. Xu).

ultra-sparse case, and also may be very efficient at certain situations for example the dictionary contains a continuum of elements [10]. However, the performance of the greedy algorithms cannot be guaranteed when the signal is not very sparse or the level of the observational noise is relatively high.

The relaxed method converts the combinatorial  $l_0$ -minimization into a more tractable model via replacing the  $l_0$ -norm with a certain nonnegative and continuous function  $P(\cdot)$ , that is

$$\min_{x \in \mathbf{R}^N} P(x) \quad \text{s.t. } y = Ax. \quad (1.3)$$

One of the most important cases is the  $l_1$ -minimization (also known as *Basis Pursuit* (BP) [11]) with  $P(x) = \|x\|_1$ , where  $\|x\|_1 = \sum_{i=1}^N |x_i|$  is called the  $l_1$ -norm. The  $l_1$ -minimization is a convex optimization problem and thus can be efficiently solved. Because of this, the  $l_1$ -minimization gets its popularity and has been accepted as a very useful tool for solution to sparsity problems. Nevertheless, it cannot promote further sparsity when applied to compressed sensing [12–16]. Moreover, many nonconvex functions were proposed as relaxations of the  $l_0$ -norm. Some typical nonconvex examples are the  $l_q$ -norm ( $0 < q < 1$ ) [12–15], smoothly clipped absolute deviation (SCAD) [17] and minimax concave penalty (MCP) [18]. As compared with the  $l_1$ -minimization, the nonconvex relaxed models can usually induce better sparsity, however, they are generally more difficult to be solved.

There are mainly two kinds of algorithms to solve the constrained optimization problem (1.3). The first one is the iteratively reweighted algorithm. Two of the most important iteratively reweighted algorithms are the reweighted  $l_1$ -minimization [16] and iteratively reweighted least squares (IRLS) [19,20] algorithms. One of the main advantages of this kind of algorithms is that they can be used to solve a general model (1.3). However, the computational complexities of these algorithms are usually relatively high. The other one is commonly called the regularization method, which transforms the constrained optimization problem (1.3) into the following unconstrained optimization problem via introducing a regularization parameter:

$$\min_{x \in \mathbf{R}^N} \{\|Ax - y\|_2^2 + \lambda P(x)\}, \quad (1.4)$$

where  $\lambda > 0$  is a regularization parameter. There are many algorithms for solving the regularization model (1.4). Particularly, for some special  $P(x)$  such as the  $l_0$ -norm,  $l_q$ -norms ( $q = 1, 2/3, 1/2$ ), SCAD and MCP, the regularization models (1.4) can permit the thresholding representations and thus yield the corresponding iterative thresholding algorithms [15,21–23]. Intuitively, an iterative thresholding algorithm can be seen as a procedure of Landweber iteration projected by a certain thresholding operator. Compared to the aforementioned algorithms including the greedy, BP and iteratively reweighted algorithms, iterative thresholding algorithms can be implemented fast and have almost the least computational complexity for large scale problems [24–26]. So far, most of the theoretical results of the iterative thresholding algorithms were developed for the regularization model (1.4) with fixed  $\lambda$ . However, it is in general

difficult to determine an appropriate regularization parameter  $\lambda$ , especially when  $P$  is nonconvex.

Alternatively, some adaptive strategies for setting the regularization parameters were proposed for iterative thresholding algorithms. One of the commonly used strategies is to set the regularization parameter adaptively according to a specified sparsity level at each iteration. Once the specified sparsity level is given, the number of nonzero components of vector at each iteration is also determined. In practice, the specified sparsity level is desired to be a good estimation of the true sparsity level. This strategy was first adopted to the iterative hard thresholding algorithm (called hard algorithm for short) in [27], and later the iterative soft [28] (called soft algorithm for short) and half [15] (called half algorithm for short) thresholding algorithms. The convergence of hard algorithm was justified when  $A$  satisfies a certain restricted isometry property (RIP) [27]. Later, Maleki investigated the convergence of both hard and soft algorithms in terms of the coherence [28]. Both in the analysis of [27,28], the specified sparsity levels of AIT algorithms are set to be the true sparsity level of the original sparse solution, however, which is commonly unknown in practice. Therefore, the robustness of AIT algorithms to the specified sparsity levels is very important in practice and worth of investigation. Moreover, besides the hard and soft algorithms, there are many other AIT algorithms such as half, SCAD, MCP algorithms which are widely used in signal processing, variable selection and feature extraction. However, as far as we know, there are lack of the corresponding theoretical guarantees on the global convergence of these algorithms for sparse solution to the underdetermined linear equations. Thus, the theoretical performance of these AIT algorithms should be further studied.

### 1.1. Main contribution

In this paper, we consider the global convergence of a wide class of adaptively iterative thresholding (AIT) algorithms for sparse solution to an underdetermined system of linear equations. The associated thresholding functions satisfy some basic assumptions including odevity, monotonicity and boundedness. The main contribution of this paper can be summarized as follows:

- (1) *Developing the global convergence of an abstract AIT algorithm.* In this paper, under certain assumptions, we developed the global convergence of a wide class of AIT algorithms, which include most of the existing iterative thresholding algorithms emerged in the field of sparse signal processing.
- (2) *Robustness to the algorithmic parameter.* In practice, a good algorithm is commonly desired to be robust to the algorithmic parameters. Distinguished from most of the existing results, the main theorem of this paper (see [Theorem 1](#)) demonstrates that AIT algorithms are indeed robust to the algorithmic parameter.
- (3) *Identifying the support set within finite iterations.* As shown by [Theorem 1](#), AIT algorithms can identify the correct support set within finite iterations. Furthermore, we give

an estimation of the upper bound on the number of iterations required for identifying the correct support set.

- (4) *Linear rate of asymptotic convergence.* We show that the rates of asymptotic convergence of AIT algorithms are linear. More specifically, once the correct support set has been identified, AIT algorithms converge to the original sparse solution exponentially fast. It should be pointed out that the linear rates of asymptotic convergence of AIT algorithms are non-trivial since most of the thresholding operators studied in this paper are expansive. Thus, the classical theoretical results of the Landweber iteration cannot be straightly applicable to these algorithms.
- (5) *Support identified via a sequential recruitment process.* By the proof of [Theorem 1](#), we can observe that the supports are sequentially recruited into the support set in the pre-defined descending order of the values (in magnitude) of the components of the original sparse solution. This property may be very useful for certain applications such as a feature screening problem.

The remainder of this paper is organized as follows. In [Section 2](#), we introduce the adaptively iterative thresholding (AIT) algorithms. In [Section 3](#), we present the main theoretical results of AIT algorithms. In [Section 4](#), we give the proof of the main theorem. In [Section 5](#), we discuss some related work. We conclude the paper in [Section 6](#).

## 2. Adaptively iterative thresholding algorithms

In this section, we first give some notations used in this paper, and then introduce the adaptively iterative thresholding algorithms.

### 2.1. Notion and notation

For any  $x \in \mathbf{R}^N$ ,  $x_i$  represents its  $i$ -th component. Given a positive integer  $k < N$ ,  $|x|_{(k)}$  represents its  $k$ -th largest component of  $x$  in magnitude. For any  $A \in \mathbf{R}^{M \times N}$ ,  $A_i \in \mathbf{R}^M$  denotes its  $i$ -th column, and  $A^T$  represents its transposition. For any index set  $S$ ,  $|S|$  denotes its cardinality and  $S^c$  represents its complementary set. Moreover, we denote by  $A_S$  the submatrix of  $A$  with the columns restricted to  $S$ .

### 2.2. AIT algorithms

The adaptively iterative thresholding algorithm for sparse solution to (1.1) can be generally expressed as the following iterative form:

$$z^{(t+1)} = x^{(t)} - A^T(Ax^{(t)} - y), \tag{2.1}$$

$$x^{(t+1)} = H_{\tau^{(t+1)}}(z^{(t+1)}), \tag{2.2}$$

where

$$H_{\tau^{(t+1)}}(x) = (h_{\tau^{(t+1)}}(x_1), \dots, h_{\tau^{(t+1)}}(x_N))^T \tag{2.3}$$

is a componentwise thresholding operator associated with a thresholding function  $h_{\tau^{(t+1)}}$ ,  $\tau^{(t+1)}$  is the threshold value at  $(t+1)$ -th iteration. More specifically, a thresholding function  $h_{\tau}$  is commonly defined as

$$h_{\tau}(u) = \begin{cases} f_{\tau}(u), & |u| > \tau \\ 0, & \text{otherwise} \end{cases} \tag{2.4}$$

where  $f_{\tau}(u)$  is formally called the defining function for any  $u \in \mathbf{R}$ . We give some basic assumptions of the defining function as follows:

1. *Odevity.*  $f_{\tau}$  is an odd function.
2. *Monotonicity.*  $f_{\tau}(u) \geq f_{\tau}(v)$  for any  $u \geq v \geq 0$ .
3. *Boundedness.* There exists a constant  $0 \leq c \leq 1$  such that  $u - c\tau \leq f_{\tau}(u) \leq u$  for  $u \geq \tau$ .

The odevity and monotonicity are two regular assumptions for the defining function, while the boundedness confines  $h_{\tau}$  to be an appropriate thresholding function. It can be noted that most of the commonly used thresholding functions satisfy these assumptions. We list some typical examples as follows.

**Example 1.** Hard thresholding function for  $L_0$  regularization [\[23\]](#)

$$h_{\tau,0}(u) = \begin{cases} u, & |u| > \tau \\ 0, & \text{otherwise} \end{cases} \tag{2.5}$$

**Example 2.** Half thresholding function for  $L_{1/2}$  regularization [\[15\]](#)

$$h_{\tau,1/2}(u) = \begin{cases} \frac{2}{3}u \left( 1 + \cos \left( \frac{2\pi}{3} - \frac{2}{3} \arccos \left( \frac{\sqrt{2}}{2} \left( \frac{\tau}{|u|} \right)^{3/2} \right) \right) \right), & |u| > \tau \\ 0, & \text{otherwise} \end{cases} \tag{2.6}$$

**Example 3.** 2/3-Thresholding function for  $L_{2/3}$  regularization [\[22\]](#)

$$h_{\tau,2/3}(u) = \begin{cases} \text{sign}(u) \left( \frac{\phi_{\tau}(u) + \sqrt{\frac{2|u|}{\phi_{\tau}(u)} - \phi_{\tau}(u)^2}}{2} \right)^3, & |u| > \tau \\ 0, & \text{otherwise} \end{cases} \tag{2.7}$$

where  $\text{sign}(u)$  denotes the sign function of  $u$  henceforth,

$$\phi_{\tau}(u) = \frac{2^{13/16}}{4\sqrt{3}} \tau^{3/16} \left( \cosh \left( \frac{\theta_{\tau}(u)}{3} \right) \right)^{1/2} \quad \text{with}$$

$$\theta_{\tau}(u) = \text{arccosh} \left( \frac{3\sqrt{3}u^2}{2^{7/4}(2\tau)^{9/8}} \right).$$

**Example 4.** Soft thresholding function for  $L_1$  regularization [\[21\]](#)

$$h_{\tau,1}(u) = \begin{cases} u - \text{sign}(u)\tau, & |u| > \tau \\ 0, & \text{otherwise} \end{cases} \tag{2.8}$$

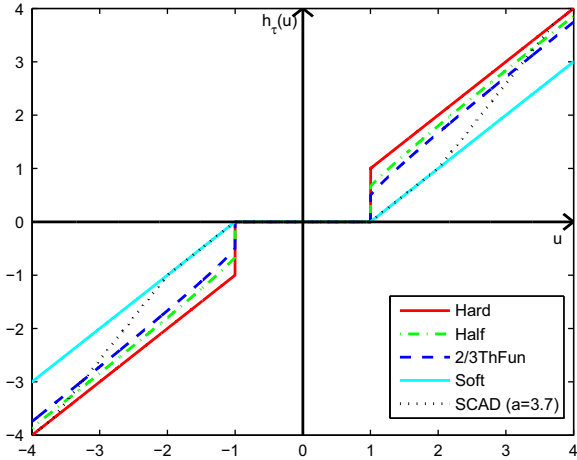


Fig. 1. Typical thresholding functions  $h_\tau(u)$  with  $\tau = 1$ .

**Table 1**  
Boundedness parameters  $c$  for different thresholding functions.

$f_{\tau,*}$	$f_{\tau,0}$	$f_{\tau,1/2}$	$f_{\tau,2/3}$	$f_{\tau,1}$	$f_{\tau,SCAD}$
$c$	0	$\frac{1}{3}$	$\frac{1}{2}$	1	1
$\frac{1}{3+c}$	$\frac{1}{3}$	$\frac{3}{10}$	$\frac{2}{7}$	$\frac{1}{4}$	$\frac{1}{4}$

**Example 5.** SCAD-thresholding function for nonconvex likelihood model ( $a > 2$ ) [17]

$$h_{\tau,SCAD}(u) = \begin{cases} u, & |u| > a\tau \\ \frac{(a-1)u - \text{sign}(u)a\tau}{a-2}, & 2\tau < |u| \leq a\tau \\ u - \text{sign}(u)\tau, & \tau < |u| \leq 2\tau \\ 0, & \text{otherwise} \end{cases} \quad (2.9)$$

The plots of these thresholding functions and their corresponding boundedness parameters  $c$  are shown in Fig. 1 and Table 1, respectively.

It can be observed that the tuning strategies of the threshold value  $\tau^{(t)}$  are crucial for AIT algorithms. In this paper, we consider a heuristic way for setting the threshold value, i.e., the threshold value is set to the  $(k+1)$ -th largest coefficient of  $z$  in magnitude at each iteration, where  $k$  is the unique algorithmic parameter and called the specified sparsity level. Therefore, the adaptively iterative thresholding algorithms can be formulated as Algorithm 1.

**Algorithm 1.** Adaptively iterative thresholding algorithm.

- Step 1. Normalize  $A$  such that  $\|A_j\|_2 = 1$  for  $j = 1, \dots, N$ ;
- Step 2. Choose a specified sparsity level  $k$  and begin with  $x^{(0)} = 0$ ;
- Step 3. Compute  $z^{(t+1)} = x^{(t)} + A^T(y - Ax^{(t)})$ ;
- Step 4. Set  $\tau^{(t+1)} = |z^{(t+1)}|_{[k+1]}$ ;
- Step 5. Update  $x^{(t+1)} = H_{\tau^{(t+1)}}(z^{(t+1)})$ ;
- Step 6. Repeat steps 3–5 until the stop rule being satisfied;

It should be noticed that at  $(t+1)$ -th iteration, the AIT algorithm yields a sparse solution with  $k$  nonzero components by setting  $\tau^{(t+1)} = |z^{(t+1)}|_{[k+1]}$  in step 4 of Algorithm 1. To guarantee the performance of the AIT algorithm, the specified sparsity level is very critical. Assume that the true sparsity level of the original sparse solution is  $k^*$ . On one hand, when  $k \geq k^*$ , the results will get better as  $k$  approaching to  $k^*$ . On the other hand, once  $k < k^*$ , the AIT algorithm fails to find the original sparse solution. Thus,  $k$  should be specified as an upper bound estimation of  $k^*$ .

### 3. Convergence analysis of AIT algorithms

In this section, we provide the convergence analysis of AIT algorithms for sparse solution to (1.1). For simplicity, we assume that the normalization step has been done before the analysis, that is,  $\|A_j\|_2 = 1$  for  $j = 1, \dots, N$ . We use  $x^* = (x_1^*, \dots, x_N^*)^T$  to denote the original sparse solution with  $k^*$  nonzero components. Without loss of generality, we further assume that  $|x_1^*| \geq |x_2^*| \geq \dots \geq |x_{k^*}^*| > 0$  and  $x_j^* = 0$  for  $j > k^*$ . Moreover, we denote by  $I^*$  and  $I^{(t)}$  the support sets of  $x^*$  and  $x^{(t)}$ , respectively. Furthermore, we denote  $I_r = \{1, \dots, r\}$  for  $1 \leq r \leq k^*$  as the set formed by the first  $r$  largest components of  $x^*$  in magnitude. Thus, we have  $I^* = I_{k^*}$ .

To investigate the convergence of AIT algorithms, we introduce the coherence of a matrix  $A$ , which is defined as follows [29]:

$$\mu(A) = \max_{i \neq j} | \langle A_i, A_j \rangle | \quad \text{for } i, j \in \{1, \dots, N\}.$$

The coherence measures the maximal correlation between two different columns of  $A$ . For simplicity, we use  $\mu$  instead of  $\mu(A)$  henceforth if there is no confusion. In [29], it was shown that if  $k^* \leq \frac{1}{2}(1 + 1/\mu)$ , then  $x^*$  is the unique sparsest solution of (1.1). Next, we define the dynamic range of the original sparse solution as

$$Dr = \frac{\min_{i \in I^*} |x_i^*|}{\min_{i \in I^*} |x_i^*|},$$

which measures the diversity of the nonzero components of  $x^*$ . Moreover, we define two positive constants in the following:

$$T_{k^*} = k^* + (k^* - 1) \log_{(1+c)k\mu} \frac{1 - (3+c)k\mu}{(3+c) - (c^2 + 4c + 3 + 2/Dr)k\mu} - \log_{(1+c)k\mu} Dr \quad (3.1)$$

and

$$T_{k^*}^* = k^* + (k^* - 1) \log_{(1+c)k^*\mu} \frac{1 - (3+c)k^*\mu}{(3+c) - (c^2 + 4c + 3 + 2/Dr)k^*\mu} - \log_{(1+c)k^*\mu} Dr. \quad (3.2)$$

With these notations, we present the main result as follows.

**Theorem 1.** Suppose that  $0 < \mu < 1/(3+c)k^*$  and  $k^* \leq k < 1/(3+c)\mu$ . Then there exists a positive integer  $t^* \leq T_{k^*}$  such that when  $t \geq t^*$ , it holds

$$I^* \subset I^{(t)}, \quad (3.3)$$

and

$$\|x^{(t)} - x^*\|_\infty \leq \frac{3+c}{2} \min_{i \in I^*} |x_i^*| \rho^{t-t^*+1} \tag{3.4}$$

with  $\rho = (1+c)k\mu < 1/2$ .

In [Theorem 1](#), we justify the global convergence of AIT algorithms. It shows that as long as  $A$  satisfies a certain coherence property and the specified sparsity level  $k$  is chosen in an appropriate range, AIT algorithms can find the correct support set within finite iterations. Furthermore, once the correct support set has been identified, AIT algorithms converge to the original sparse solution exponentially fast.

As shown by [Theorem 1](#) and [\(3.1\)](#), the upper bound on the number of iterations required for identifying the correct support set is mainly determined by several parameters, i.e.,  $k^*$ ,  $Dr$  and  $k$ . On one hand, according to [\(3.1\)](#),  $T_{k^*}$  is monotonic increasing with respect to both  $k^*$  and  $Dr$ . In other words, if the original sparse solution has more nonzero components and its dynamic range is larger, then more iterations are commonly required to identify the correct support set. These coincide with the common senses. As we all know, it is generally more difficult to find a denser solution. Also, if the dynamic range of the original solution is larger, more efforts are usually required to detect the smallest nonzero component. On the other hand, we can easily verify that  $T_{k^*}$  is monotonically increasing with respect to  $k$ . Therefore, if the specified sparsity level  $k$  is estimated more precisely, then the number of iterations required for finding the correct support set may get fewer. Moreover, according to [\(3.4\)](#), it can be seen that AIT algorithms converge faster with smaller  $\rho$  when  $k$  is closer to  $k^*$ . Thus, in practice,  $k$  is desired to be estimated more precisely in terms of computational efficiency and convergence speed.

As analysed previously, a tighter upper bound estimation of the true sparsity level is more desired for the AIT algorithm in the perspectives of both theory and practice. However, the upper bound is commonly unknown in practice. In applications, we may conduct an empirical study or based on some known priors to yield a reasonable upper bound. Moreover, there are several efficient ways inspired by some theoretical analysis. In [\[30\]](#), it suggested that an upper bound can be estimated by the undersampling-sparsity tradeoff, or “phase-transition curve”. However, it is generally very time-consuming to obtain the “phase-transition curve”. According to [\[31\]](#), it was shown that the coherence satisfies  $\mu \in [\sqrt{(N-M)/M(N-1)}, 1]$ . The lower bound is known as the Welch bound [\[32\]](#). Particularly, when  $N \gg M$ , the lower bound is approximately  $\mu \geq 1/\sqrt{M}$ . Together with [Theorem 1](#), we can suggest  $\mathcal{O}(\sqrt{M})$  as a reasonable upper bound estimation of  $k^*$ .

In the following, we give a corollary to show the special case with  $k = k^*$ .

**Corollary 1.** Suppose that  $0 < \mu < 1/(3+c)k^*$  and  $k = k^*$ . Then there exists a positive integer  $\hat{t}^* \leq T_{k^*}^*$  such that when  $t \geq \hat{t}^*$ , it holds

$$I^* = I^{(t)} \tag{3.5}$$

and

$$\|x^{(t)} - x^*\|_\infty \leq \frac{3+c}{2} \min_{i \in I^*} |x_i^*| \hat{\rho}^{t-\hat{t}^*} + 1 \tag{3.6}$$

with  $\hat{\rho} = (1+c)k^*\mu < 1/2$ .

From [Corollary 1](#), when  $k = k^*$ , the AIT algorithm can recover the support set of  $x^*$  exactly within finite iterations. According to [\(3.2\)](#), it can be observed that if  $k^*\mu$  is not sufficient close to  $1/(3+c)$  and the dynamic range of the original sparse solution is not too large, then the log term about  $k^*\mu$  and  $Dr$  in the second and third terms of [\(3.2\)](#) respectively is relatively small constants. In this case, the number of iterations required for the AIT algorithm is about several times of  $k^*$ . For an instance, assume that  $k^* = 9$ ,  $\mu = \frac{1}{40}$  and  $Dr = 10$ , according to [\(3.2\)](#), the number of iterations required for *hard*, *soft* and *half* algorithms are 20, 42 and 25, which are about 2, 5 and 3 times of  $k^*$ , respectively. Motivated by this observation, we can suggest an efficient halting rule for AIT algorithms through setting the number of maximum iterations according to the true sparsity level.

It can be observed from [Corollary 1](#) that the boundedness parameter  $c$  plays an important role in the guarantees of the convergence of AIT algorithms. The restriction of the matrix  $A$  gets stricter as  $c$  increases. As shown in [Table 1](#), among these AIT algorithms, hard algorithm permits the weakest requirement of  $A$  with  $\mu < 1/3k^*$ , while soft algorithm requires the strictest restriction of  $A$  with  $\mu < 1/4k^*$ . It should be noticed that the restriction on  $\mu$  is relatively loose and can be attained in practice. In fact, it was shown that the coherence  $\mu$  is in the order of  $\sqrt{\log N/M}$  for the random matrix where entries of  $A$  are independently and identically Gaussian distributed [\[33\]](#). This implies that  $k^* = \mathcal{O}(M^{\xi_1})$  might suffice for the AIT algorithm when  $\log N = \mathcal{O}(M^{\xi_2})$  for some positive constants  $\xi_1$  and  $\xi_2$  satisfying  $2\xi_1 + \xi_2 < 1$ .

**Remark 1.** As shown by the proof of [Theorem 1](#) in [Section 4](#), it is interested that the procedure of identifying the correct support set is a sequential recruitment process. That is, the supports are sequentially recruited in a descending order of the values of their coefficients with the larger one being identified not later than the smaller one. This procedure may be very useful to certain applications such as feature screening problem in statistics.

#### 4. Proof of Theorem 1

We denote  $i_{[k+1]}^{(t)} = \arg \min_{i \in \{1,2,\dots,N\}} \{i : |z_i^{(t)}| = |z^{(t)}|_{[k+1]}\}$  and then let  $\Lambda_{[k+1]}^{(t)} = I^{(t)} \cup \{i_{[k+1]}^{(t)}\}$ . To prove [Theorem 1](#), we need the following lemmas. First, we give a lemma to bound the gap between the components of  $x^{(t)}$  and  $z^{(t)}$  at  $t$ -th iteration, which is served as the basis of the other lemmas.

**Lemma 1.** At any  $t$ -th iteration ( $t \geq 1$ ), there exists an  $i_0^{(t)} \in \Lambda_{[k+1]}^{(t)} \setminus I^*$ , such that

$$(i) \text{ for any } i \in I^{(t)}, \tag{4.1}$$

$$|z_i^{(t)} - x_i^{(t)}| \leq c |z_{i_0^{(t)}}^{(t)} - x_{i_0^{(t)}}^*|,$$



where  $c$  is the boundedness parameter of the associated thresholding function;

(ii) for any  $i \notin I^{(t)}$ ,

$$|z_i^{(t)} - x_i^{(t)}| \leq |z_{i_0}^{(t)} - x_{i_0}^{*(t)}|. \quad (4.2)$$

Here, it should be mentioned that  $x_{i_0}^{*(t)} = 0$  and we keep it in (4.1) and (4.2) only for better formats.

**Proof.** (i) For  $i \in I^{(t)}$ , by the definition of the thresholding function  $H_\tau$  and the boundedness assumption of  $f_\tau$ , it holds

$$|z_i^{(t)} - x_i^{(t)}| = |z_i^{(t)} - f_{\tau(t)}(z_i^{(t)})| \leq c\tau^{(t)} = c|z_i^{(t)}|_{|k+1}. \quad (4.3)$$

Since  $i_{[k+1]}^{(t)} \notin I^{(t)}$ , then the cardinality of  $\Lambda_{[k+1]}^{(t)}$  is  $k+1$ . Moreover, by  $|I^*| = k^* < k+1$ , then there exists an index  $i_0^{(t)}$  such that  $i_0^{(t)} \in \Lambda_{[k+1]}^{(t)} \setminus I^*$ . Thus, (4.3) becomes

$$|z_i^{(t)} - x_i^{(t)}| \leq c|z_i^{(t)}|_{|k+1} \leq c|z_{i_0}^{(t)}| = c|z_{i_0}^{(t)} - x_{i_0}^{*(t)}|. \quad (4.4)$$

(ii) Similarly, for any  $i \notin I^{(t)}$ , it holds

$$|z_i^{(t)} - x_i^{(t)}| = |z_i^{(t)}| \leq |z_i^{(t)}|_{|k+1} \leq |z_{i_0}^{(t)} - x_{i_0}^{*(t)}|. \quad (4.5)$$

Thus, we end the proof of this lemma.  $\square$

In the next, we give a lemma to show that the largest component (in magnitude) of  $x^*$  will be detected at the first iteration.

**Lemma 2.** Suppose that  $0 < \mu < 1/(2k^* - 1)$  and  $k^* \leq k < \frac{1}{2}(1 + 1/\mu)$ . Then at the first iteration, it holds

- (i)  $\{1\} \subset I^{(1)}$ ;
- (ii) for any  $j \in I^{(1)}$ ,  $|x_j^{(1)} - x_j^*| \leq ((1+c)(3+c)/2)k\mu|x_1^*|$ .

**Proof.** First, we show that  $\{1\} \subset I^{(1)}$ . On one hand, we observe that

$$|z_1^{(1)}| = \left| x_1^* + \sum_{i \in I^* \setminus \{1\}} \langle A_1, A_i \rangle x_i^* \right| \geq |x_1^*| - \mu \sum_{i=2}^{k^*} |x_i^*| \geq |x_1^*| - (k-1)\mu|x_1^*|.$$

On the other hand, for any  $i \notin I^*$ , it holds

$$|z_i^{(1)}| = \left| \sum_{j=1}^{k^*} \langle A_i, A_j \rangle x_j^* \right| \leq k^* \mu |x_1^*| \leq k\mu |x_1^*|.$$

Since  $k < \frac{1}{2}(1 + 1/\mu)$ , then  $k\mu|x_1^*| < |x_1^*| - (k-1)\mu|x_1^*|$ , which implies that

$$|z_1^{(1)}| > \max_{i \notin I^*} |z_i^{(1)}|.$$

Thus,  $\{1\} \subset I^{(1)}$ .

Next, we give the error bound. For any  $j \in I^{(1)}$ , we observe that

$$|x_j^{(1)} - x_j^*| \leq |x_j^{(1)} - z_j^{(1)}| + |z_j^{(1)} - x_j^*| \leq c|x_{i_0}^{(1)} - z_{i_0}^{(1)}| + |z_j^{(1)} - x_j^*|, \quad (4.6)$$

where the second inequality holds for Lemma 1. Furthermore, for any  $i$ , it holds

$$|z_i^{(1)} - x_i^*| = \left| \sum_{j \in I^* \setminus \{i\}} \langle A_i, A_j \rangle x_j^* \right| \leq k^* \mu |x_1^*| \leq k\mu |x_1^*|. \quad (4.7)$$

Combining (4.6) with (4.7), for any  $j \in I^{(1)}$ , it holds

$$|x_j^{(1)} - x_j^*| \leq (1+c)k\mu|x_1^*| \leq \frac{(1+c)(3+c)}{2} k\mu|x_1^*|.$$

Thus, we end the proof of this lemma.  $\square$

**Lemma 3.** Suppose that  $0 < \mu < 1/(3+c)k^*$  and  $k^* \leq k < 1/(3+c)\mu$ . Moreover, assume that at  $m$ -th iteration,  $I_r \subset I^{(m)}$  ( $0 < r \leq k^*$ ) and for any  $j \in I^{(m)}$ , it holds  $|x_j^{(m)} - x_j^*| \leq ((1+c)(3+c)/2)k\mu|x_r^*|$ . Then at  $(m+s)$ -th iteration ( $s \geq 1$ ), it holds

- (i) for any  $j$ ,
 
$$|z_j^{(m+s)} - x_j^*| \leq \frac{(3+c)}{2} k\mu((1+c)k\mu)^s |x_r^*| + k\mu|x_{r+1}^*| [1 + (1+c)k\mu + \dots + ((1+c)k\mu)^{s-1}];$$
- (ii) for any  $i \in I^{(m+s)}$ ,
 
$$|x_i^{(m+s)} - x_i^*| \leq \frac{(3+c)}{2} ((1+c)k\mu)^{s+1} |x_r^*| + k\mu|x_{r+1}^*| [(1+c)k\mu + \dots + ((1+c)k\mu)^s];$$
- (iii)  $I_r \subset I^{(m+s)}$ .

**Proof.** We prove this lemma by induction. First, when  $s=1$ , for any  $i \in I^{(m+1)}$ , it holds

$$|x_i^{(m+1)} - x_i^*| \leq |x_i^{(m+1)} - z_i^{(m+1)}| + |z_i^{(m+1)} - x_i^*|.$$

By Lemma 1, there exists an  $i_0^{(m+1)} \in \Lambda_{[k+1]}^{(m+1)} \setminus I^*$  such that

$$|x_{i_0}^{(m+1)} - z_{i_0}^{(m+1)}| \leq c|z_{i_0}^{(m+1)} - x_{i_0}^{*(m+1)}|,$$

then it holds

$$|x_i^{(m+1)} - x_i^*| \leq c|z_{i_0}^{(m+1)} - x_{i_0}^{*(m+1)}| + |z_i^{(m+1)} - x_i^*|. \quad (4.8)$$

Moreover, for any  $j$ , it holds

$$\begin{aligned} |z_j^{(m+1)} - x_j^*| &= \left| \sum_{i \in I^{(m)} \cup \{j\}} \langle A_j, A_i \rangle (x_i^* - x_i^{(m)}) \right| \\ &= \left| \sum_{i \in I^{(m)} \setminus \{j\}} \langle A_j, A_i \rangle (x_i^* - x_i^{(m)}) + \sum_{i \in I^* \setminus (I^{(m)} \cup \{j\})} \langle A_j, A_i \rangle x_i^* \right| \\ &\leq k\mu \left( \frac{(1+c)(3+c)}{2} k\mu |x_r^*| \right) + (k^* - r)\mu |x_{r+1}^*| \\ &\leq \frac{(3+c)}{2} k\mu((1+c)k\mu |x_r^*|) + k\mu |x_{r+1}^*|. \end{aligned} \quad (4.9)$$

Combining (4.8) with (4.9), for any  $i \in I^{(m+1)}$ , it holds

$$\begin{aligned} |x_i^{(m+1)} - x_i^*| &\leq (1+c) \left( \frac{(3+c)}{2} k\mu \left( (1+c)k\mu |x_r^*| \right) + k\mu |x_{r+1}^*| \right) \\ &= \frac{(3+c)}{2} ((1+c)k\mu)^2 |x_r^*| + (1+c)k\mu |x_{r+1}^*|. \end{aligned} \quad (4.10)$$

Then we need to prove that  $I_r \subset I^{(m+1)}$ . According to (4.9), for any  $j$ , it holds

$$|z_j^{(m+1)} - x_j^*| \leq \left( 1 + \frac{(3+c)(1+c)}{2} k\mu \right) k\mu |x_r^*|.$$

Since  $k < 1/(3+c)\mu$ , it holds

$$\left(1 + \frac{(3+c)(1+c)}{2}k\mu\right)k\mu < \frac{1}{2}.$$

Then, for any  $j$ , it holds

$$|z_j^{(m+1)} - x_j^*| < \frac{1}{2}|x_r^*|. \tag{4.11}$$

According to (4.11), we observe that, for any  $i \in I_r$ ,

$$|z_i^{(m+1)}| \geq |x_i^*| - |z_i^{(m+1)} - x_i^*| \geq |x_r^*| - \frac{1}{2}|x_r^*| > \frac{1}{2}|x_r^*|. \tag{4.12}$$

While for  $i \notin I^*$ ,

$$|z_i^{(m+1)}| = |z_i^{(m+1)} - x_i^*| < \frac{1}{2}|x_r^*|. \tag{4.13}$$

With (4.12) and (4.13), it follows that  $I_r \subset I^{(m+1)}$ . Therefore, the conclusion holds for  $s=1$ .  $\square$

Second, assume that the conclusion holds for  $s (s \geq 1)$ , then we need to check if it holds for  $s+1$ . The proof is similar to the case  $s=1$  and we omit it here.

**Lemma 4.** Suppose that  $0 < \mu < 1/(3+c)k^*$  and  $k^* \leq k < 1/(3+c)\mu$ . Moreover, assume that at  $m$ -th iteration,  $I_r \subset I^{(m)}$  ( $r < k^*$ ) and for any  $j \in I^{(m)}$ ,  $|x_j^{(m)} - x_j^*| \leq ((1+c)(3+c)/2)k\mu|x_r^*|$ . Then it holds

(i) the index  $\{r+1\}$  will be detected after at most  $l_r$  iterations with

$$l_r = \left\lceil \log_{(1+c)k\mu} \frac{1-(3+c)k\mu}{(3+c)(1-(1+c)k\mu)|x_r^*|/|x_{r+1}^*|-2k\mu} \right\rceil,$$

where the function  $\lceil u \rceil$  denotes the smallest integer not less than  $u$  for any  $u \in \mathbb{R}$ .

(ii) for any  $j \in I^{(m+l_r+1)}$ ,

$$|x_j^{(m+l_r+1)} - x_j^*| < \frac{(1+c)(3+c)}{2}k\mu|x_{r+1}^*|.$$

**Proof.** We first show that the index  $\{r+1\}$  will be detected after at most  $l_r$  iterations, and then give the error bound. According to Lemma 3, at  $(m+l_r)$ -th iteration, for any  $j$ , it holds

$$\begin{aligned} |z_j^{(m+l_r)} - x_j^*| &\leq \frac{(3+c)}{2}((1+c)k\mu)^{l_r}|x_r^*| \\ &\quad + k\mu|x_{r+1}^*| \left(1 + \dots + ((1+c)k\mu)^{l_r-1}\right) \\ &< \frac{(3+c)}{2}((1+c)k\mu)^{l_r}|x_r^*| + k\mu|x_{r+1}^*| \frac{1-((1+c)k\mu)^{l_r}}{1-(1+c)k\mu} \\ &= |x_{r+1}^*| \left( \frac{(3+c)}{2}((1+c)k\mu)^{l_r} \frac{|x_r^*|}{|x_{r+1}^*|} + k\mu \frac{1-((1+c)k\mu)^{l_r}}{1-(1+c)k\mu} \right) \\ &\leq |x_{r+1}^*| \left( \frac{(3+c)}{2}((1+c)k\mu)^{l_r} \frac{|x_r^*|}{|x_{r+1}^*|} + k\mu \frac{1-((1+c)k\mu)^{l_r}}{1-(1+c)k\mu} \right). \end{aligned}$$

Since

$$l_r \geq \log_{(1+c)k\mu} \frac{1-(3+c)k\mu}{(3+c)(1-(1+c)k\mu)|x_r^*|/|x_{r+1}^*|-2k\mu},$$

then

$$\frac{(3+c)}{2}((1+c)k\mu)^{l_r} \frac{|x_r^*|}{|x_{r+1}^*|} + k\mu \frac{1-(1+c)k\mu)^{l_r}}{1-(1+c)k\mu} \leq \frac{1}{2}.$$

Thus, for any  $j$ , it holds

$$|z_j^{(m+l_r)} - x_j^*| < \frac{1}{2}|x_{r+1}^*|. \tag{4.14}$$

By (4.14), on one hand

$$|z_{r+1}^{(m+l_r)}| \geq |x_{r+1}^*| - |z_{r+1}^{(m+l_r)} - x_{r+1}^*| > \frac{1}{2}|x_{r+1}^*|, \tag{4.15}$$

and on the other hand, for any  $j \notin I^*$ ,

$$|z_j^{(m+l_r)}| = |z_j^{(m+l_r)} - x_j^*| < \frac{1}{2}|x_{r+1}^*|. \tag{4.16}$$

With (4.15) and (4.16), it shows that  $\{r+1\}$  will be detected at  $(m+l_r)$ -th iteration, that is,  $\{r+1\} \subset I^{(m+l_r)}$ .

Next, we give the upper bound of the error. For any  $i \in I^{(m+l_r+1)}$ , it holds

$$\begin{aligned} |x_i^{(m+l_r+1)} - x_i^*| &= \left| \sum_{j \in I^{(m+l_r)} \setminus \{i\}} \langle A_i, A_j \rangle (x_j^* - x_j^{(m+l_r)}) \right. \\ &\quad \left. + \sum_{j \in I^* \setminus (I^{(m+l_r)} \cup \{i\})} \langle A_i, A_j \rangle \beta_j^* \right| \\ &\leq \mu \sum_{j \in I^{(m+l_r)} \setminus \{i\}} |x_j^* - x_j^{(m+l_r)}| + (k^* - r - 1)\mu|x_{r+1}^*|. \end{aligned} \tag{4.17}$$

Moreover, for any  $j \in I^{(m+l_r)}$ , it holds

$$|x_j^* - x_j^{(m+l_r)}| \leq |x_j^* - z_j^{(m+l_r)}| + |z_j^{(m+l_r)} - x_j^{(m+l_r)}|. \tag{4.18}$$

According to Lemma 1 and (4.14), then (4.18) becomes

$$|x_j^* - x_j^{(m+l_r)}| < \frac{1}{2}|x_{r+1}^*| + c|z_{i_0}^{(m+l_r)} - x_{i_0}^*| < \frac{1+c}{2}|x_{r+1}^*|. \tag{4.19}$$

Combining (4.17) and (4.19), for any  $i \in I^{(m+l_r+1)}$ , it holds

$$\begin{aligned} |x_i^{(m+l_r+1)} - x_i^*| &\leq \frac{(1+c)}{2}k\mu|x_{r+1}^*| + (k^* - r - 1)\mu|x_{r+1}^*| \\ &= \left( \frac{1+c}{2} + \frac{k^* - r - 1}{k} \right)k\mu|x_{r+1}^*| \\ &\leq \frac{(1+c)(3+c)}{2}k\mu|x_{r+1}^*|. \end{aligned}$$

Therefore, for any  $i \in I^{(m+l_r+1)}$ , it holds

$$|x_i^{(m+l_r+1)} - x_i^*| \leq \frac{(1+c)(3+c)}{2}k\mu|x_{r+1}^*|.$$

Thus, we end the proof of Lemma 4.  $\square$

**Proof of Theorem 1.** With these lemmas, we prove Theorem 1 inductively. For  $i=1$ , by Lemma 2, the largest component (in magnitude) will be detected at the first iteration, that is,  $I_1 = \{1\} \subset I^{(1)}$ . By Lemma 3, once the first largest index is identified, it remains in the support set forever. Furthermore, by Lemma 4, the second largest component will be identified after at most  $l_1$  iterations, i. e.,  $I_2 \subset I^{(t)}$  when  $t \geq 1+l_1$ . In order to obtain the required error bound for the inductive procedure, one more iteration should be implemented. When this procedure is repeated for  $r$  times with  $0 < r \leq k^* - 1$ , it holds  $I_{r+1} \subset I^{(t)}$  when  $t \geq r + \sum_{i=1}^r l_i$ . Furthermore, by Lemma 3, once all

the correct indices are detected, they remain in the support set and the error estimation of the iteration can be obtained. Therefore, there exists an integer constant  $t^* \leq k^* + \sum_{i=1}^{k^*-1} l_i$  such that when  $t \geq t^*$ , it holds  $I^* \subset I^{(t)}$  and the error estimation of the iteration can be achieved. Moreover, by the definition of  $l_i$  in Lemma 4 and the fact that  $|x_i^*|/|x_{i+1}^*| \leq Dr$ , it holds

$$\begin{aligned} l_i &\leq \log_{(1+c)k\mu} \frac{1-(3+c)k\mu}{(3+c)(1-(1+c)k\mu)|x_i^*|/|x_{i+1}^*|-2k\mu} \\ &\leq \log_{(1+c)k\mu} \frac{1-(3+c)k\mu}{(3+c)-(c^2+4c+3+2/Dr)k\mu} \\ &\quad - \log_{(1+c)k\mu} \frac{|x_i^*|}{|x_{i+1}^*|} \end{aligned} \quad (4.20)$$

for  $i = 1, \dots, k^* - 1$ . Therefore,

$$\begin{aligned} k^* + \sum_{i=1}^{k^*-1} l_i &\leq k^* + (k^* - 1) \log_{(1+c)k\mu} \\ &\quad \times \frac{1-(3+c)k\mu}{(3+c)-(c^2+4c+3+2/Dr)k\mu} - \log_{(1+c)k\mu} \frac{|x_1^*|}{|x_{k^*}^*|} = T_{k^*}. \quad \square \end{aligned}$$

Thus, we obtain the proof of Theorem 1.

## 5. Related work

In this section, we first discuss some related work of AIT algorithms, and then give some comparisons with other typical algorithms including BP, OMP, CoSaMP in terms of the sufficient condition for convergence and computational complexity.

(i) *On related work of AIT algorithms.* In [28], Maleki provided some similar results for two special AIT algorithms, i.e., the hard and soft algorithms with  $k = k^*$ . The sufficient conditions for convergence are  $\mu < 1/3.1k^*$  and  $\mu < 1/4.1k^*$  for hard and soft algorithms, respectively. As shown by Corollary 1, our conditions for both algorithms are slightly weaker than Maleki's conditions. Moreover, from Theorem 1, we show the robustness of AIT algorithms to the specified sparsity levels, which is very important in practice. Except the hard and soft algorithms, as far as we know, there are no similar results on the global convergence of other AIT algorithms such as half, SCAD and MCP algorithms for sparse solution to the underdetermined linear equations.

Besides the coherence property, another important property called the restricted isometry property (RIP) is commonly used to characterize the performance of an algorithm for sparse solution to (1.1). The  $s$ -order restricted isometry constant (RIC),  $\delta_s$  of  $A$  is defined as the smallest constant  $0 < \delta < 1$  such that

$$(1 - \delta)\|x\|_2^2 \leq \|Ax\|_2^2 \leq (1 + \delta)\|x\|_2^2, \quad \forall \|x\|_0 \leq s. \quad (5.1)$$

In [34], it was demonstrated that if  $A$  has unit-norm columns and coherence  $\mu$ , then  $A$  has the  $(s, \delta_s)$ -RIP with  $\delta_s \leq (s - 1)\mu$ .

In terms of RIP, Blumensath and Davies justified the performance of the hard algorithm when applied to the signal recovery problem [27]. It was shown that if  $A$  satisfies a certain RIP with  $\delta_{3k^*} < \frac{1}{8\sqrt{2}-1}$ , then the global convergence of the hard algorithm can be guaranteed. Later, this condition

was significantly improved to by Foucart [35], i.e.,  $\delta_{3k^*} < \frac{1}{2}$ . Together with (5.2), we can easily deduce a coherence based sufficient condition of convergence, that is,  $\mu < 1/2(3k^* - 1)$ . As compared with the existing RIP based conditions, it is hard to claim if our conditions are better. Instead, we can give some useful remarks on these conditions. On one hand, the sufficient conditions based on coherence can be in general verified much easier than those based on RIP. On the other hand, the RIP based conditions can be generalized and improved usually easier than those based on coherence.

(ii) *On comparison with other algorithms.* For better comparison, we list the state-of-the-art results on sufficient conditions of some typical algorithms including BP, OMP, CoSaMP, hard, soft, half and other AIT algorithms in Table 2.

From Table 2, in the perspective of coherence, the sufficient conditions of AIT algorithms are slightly stricter than those of BP and OMP algorithms. However, AIT algorithms are generally faster than both algorithms with lower computational complexities, especially for large scale applications. As analyzed in Section 3, the number of iterations required for the convergence of the AIT algorithm is empirically of the same order of the original sparsity level  $k^*$ , that is,  $\mathcal{O}(k^*)$ . At each iteration of the AIT algorithm, only some simple matrix-vector multiplications and a projection on the vector need to be done, and thus the computational complexity per iteration is  $\mathcal{O}(MN)$ . Therefore, the total computational complexity of the AIT algorithm is  $\mathcal{O}(k^*MN)$ . While the total computational complexities of BP and OMP algorithms are generally  $\mathcal{O}(M^2N)$  and  $\max\{\mathcal{O}(k^*MN), \mathcal{O}((k^*)^2(k^* + 1)^2/4)\}$ , respectively. It should be pointed out that the computational complexity of OMP algorithm is related to the commonly used halting rule of OMP algorithm, that is, the number of maximal iterations is set to be the true sparsity level  $k^*$ .

As another important greedy algorithm, CoSaMP algorithm identifies multicomponents (commonly  $2k^*$ ) at each iteration. From Table 2, the RIP based sufficient condition of CoSaMP is  $\delta_{4k^*} < 0.384$  and a deduced coherence based sufficient condition is  $\mu < 0.384/(4k^* - 1)$ . In the perspective of coherence, our conditions for AIT algorithms are better than CoSaMP, though this comparison is not very reasonable. At each iteration of CoSaMP algorithm, some simple matrix-vector multiplications and a least squares problem should be considered. Thus, the computational complexity per iteration of CoSaMP algorithm is generally  $\max\{\mathcal{O}(MN), \mathcal{O}((3k^*)^3)\}$ , which is higher than those of AIT algorithms, especially when  $k^*$  is very large. However, the number of iterations required for CoSaMP algorithm is commonly fewer than those of AIT algorithms, since the speed of convergence of CoSaMP algorithm is exponential while that of AIT algorithms is asymptotically exponential, that is, AIT algorithms converge exponentially fast after certain iterations. Therefore, as claimed in the Introduction, when applied to very sparse case, both OMP and CoSaMP algorithms may be more efficient than AIT algorithms. While AIT algorithms may be better when applied to more general cases.

## 6. Conclusion

In this paper, we provide the convergence analysis of a wide class of adaptively iterative thresholding (AIT)



**Table 2**

Sufficient conditions for different algorithms.

Algorithm	BP	OMP	CoSaMP	Hard	Soft	Half	Other AIT
$\mu$	$\frac{1}{2k^* - 1}$ [28]	$\frac{1}{2k^* - 1}$ [32]	$\frac{0.384}{4k^* - 1}$ <sup>a</sup>	$\frac{1}{3k^*}$	$\frac{1}{4k^*}$	$\frac{3}{10k^*}$	$\frac{1}{(3+c)k^*}$
$(s, \delta_s)$	$(2k^*, 0.465)$ [31]	$(k^* + 1, \frac{1}{3\sqrt{k^*}})$ [33]	$(4k^*, 0.384)$ [34]	$(3k^*, 0.5)$ [34]	–	–	–

‘–’ represents no related theoretical result as far as we know.

<sup>a</sup> A coherence based sufficient condition for CoSaMP derived directly by the fact that  $\delta_{4k^*} < 0.384$  and  $\delta_s \leq (s-1)\mu$ 

algorithms for sparse solution to an underdetermined system of linear equations  $y=Ax$ . We prove that as long as  $A$  satisfies a certain coherence property and the specified sparsity level is set in an appropriate range, AIT algorithms can identify the correct support set within finite steps. Furthermore, we demonstrate that the asymptotic convergence rates of AIT algorithms are linear, that is, once the correct support set has been identified, AIT algorithms converge to the original sparse solution exponentially fast. It is interested that the procedure of finding the correct support set is a sequential recruitment process, i.e., the supports are sequentially recruited into the support set in the descending order of the magnitudes of their coefficients. This property may be very useful to certain applications such as the feature screening problem. It should be noted that most of the commonly used iterative thresholding algorithms (say, hard, soft, half and SCAD algorithms) are included in the class of iterative thresholding algorithms studied in this paper. Besides the hard and soft algorithms, we provide some fundamental guarantees on the performance of the other AIT algorithms for sparse solution to underdetermined linear equations.

## Acknowledgments

This work was supported by the State Key Development Programs for Basic Research of China (973Programs) (Grants nos. 2013CB329404, 2013CB329404), the Key Program of National Natural Science Foundation of China (Grant nos. 11131006), and the National Natural Science Foundations of China (Grants nos. 61075054, 60975036, 11171272). This work was also supported by Southwest China Institute of Electronic Technology. The authors gratefully acknowledge the helpful comments and suggestions of the anonymous reviewers, which have improved the presentation significantly.

## References

- [1] D.L. Donoho, Compressed sensing, *IEEE Trans. Inf. Theory* 52 (4) (2006) 1289–1306.
- [2] E.J. Candes, J. Romberg, T. Tao, Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information, *IEEE Trans. Inf. Theory* 52 (2) (2006) 489–509.
- [3] S. Mallat, Z. Zhang, Matching pursuits with time–frequency dictionaries, *IEEE Trans. Signal Process.* 41 (12) (1993) 3397–3415.
- [4] Y. Pati, R. Rezaifar, P. Krishnaprasad, Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition, in: *Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, 1993.
- [5] J.A. Tropp, A. Gilbert, Signal recovery from random measurements via orthogonal matching pursuit, *IEEE Trans. Inf. Theory* 53 (2007) 4655–4666.
- [6] D.L. Donoho, Y. Tsaig, O. Drori, J.-L. Starck, Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit, *IEEE Trans. Inf. Theory* 58 (2) (2012) 1094–1121.
- [7] D. Needell, R. Vershynin, Signal recovery from incomplete and inaccurate measurements via Regularized Orthogonal Matching Pursuit, *IEEE J. Sel. Top. Signal Process.* 4 (2010) 310–316.
- [8] D. Needell, J.A. Tropp, CoSaMP: iterative signal recovery from incomplete and inaccurate samples, *Appl. Comput. Harmon. Anal.* 26 (3) (2008) 301–321.
- [9] W. Dai, O. Milenkovic, Subspace pursuit for compressive sensing signal reconstruction, *IEEE Trans. Inf. Theory* 55 (5) (2009) 2230–2249.
- [10] J.A. Tropp, S. Wright, Computational methods for sparse solution of linear inverse problems, *Proc. IEEE* 98 (2010) 948–958.
- [11] S.S. Chen, D.L. Donoho, M.A. Saunders, Atomic decomposition by basis pursuit, *SIAM J. Sci. Comput.* 20 (1998) 33–61.
- [12] R. Chartrand, Exact reconstruction of sparse signals via nonconvex minimization, *IEEE Signal Process. Lett.* 14 (10) (2007) 707–710.
- [13] R. Chartrand, V. Staneva, Restricted isometry properties and non-convex compressive sensing, *Inverse Problems* 24 (2008) 1–14.
- [14] Z.B. Xu, H. Zhang, Y. Wang, X.Y. Chang, Y. Liang,  $L_{1/2}$  regularizer, *Sci. China, Ser. F–Inf. Sci.* 53 (2010) 1159–1169.
- [15] Z.B. Xu, X.Y. Chang, F.M. Xu, H. Zhang,  $L_{1/2}$  regularization: a thresholding representation theory and a fast solver, *IEEE Trans. Neural Netw. Learning Syst.* 23 (2012) 1013–1027.
- [16] E.J. Candes, M.B. Wakin, S.P. Boyd, Enhancing sparsity by reweighted  $l_1$  minimization, *J. Fourier Anal. Appl.* 14 (5) (2008) 877–905.
- [17] J. Fan, R. Li, Variable selection via nonconcave penalized likelihood and its oracle properties, *J. Am. Stat. Assoc.* 96 (2001) 1348–1360.
- [18] C.H. Zhang, Nearly unbiased variable selection under minimax concave penalty, *Ann. Stat.* 38 (2) (2010) 894–942.
- [19] I.F. Gorodnitsky, B.D. Rao, Sparse signal reconstruction from limited data using FOCUSS: a re-weighted minimum norm algorithm, *IEEE Trans. Signal Process.* 45 (3) (1997) 600–616.
- [20] I. Daubechies, R. Devore, M. Fornasier, C.S. Gunturk, Iteratively reweighted least squares minimization for sparse recovery, *Commun. Pure Appl. Math.* 63 (2010) 1–38.
- [21] I. Daubechies, M. Defries, C. De Mol, An iterative thresholding algorithm for linear inverse problems with a sparsity constraint, *Commun. Pure Appl. Math.* 57 (2004) 1413–1457.
- [22] W.F. Cao, J. Sun, Z.B. Xu, Fast image deconvolution using closed-form thresholding formulas of  $L_q$  ( $q=1/2, 2/3$ ) regularization, *J. Vis. Commun. Image Represent.* 24 (2013) 31–41.
- [23] T. Blumensath, M.E. Davies, Iterative thresholding for sparse approximation, *J. Fourier Anal. Appl.* 14 (5) (2008) 629–654.
- [24] Y.T. Qian, S. Jia, J. Zhou, A. Robles-Kelly, Hyperspectral unmixing via  $L_{1/2}$  sparsity-constrained nonnegative matrix factorization, *IEEE Trans. Geosci. Remote Sens.* 49 (11) (2011) 4282–4297.
- [25] J.S. Zeng, J. Fang, Z.B. Xu, Sparse SAR imaging based on  $L_{1/2}$  regularization, *Sci. China Inf. Sci.* 55 (2012) 1755–1775.
- [26] J.S. Zeng, Z.B. Xu, B.C. Zhang, W. Hong, Y.R. Wu, Accelerated  $L_{1/2}$  regularization based SAR imaging via BCR and reduced Newton skills, *Signal Process.* 93 (2013) 1831–1844.
- [27] T. Blumensath, M.E. Davies, Iterative hard thresholding for compressed sensing, *Appl. Comput. Harmon. Anal.* 27 (2008) 265–274.
- [28] A. Maleki, Coherence analysis of iterative thresholding algorithms, in: *The Forty-Seventh Annual Allerton Conference, Allerton House, UIUC, IL, USA, 2009.*

- [29] D.L. Donoho, M. Elad, Optimally sparse representation in general (nonorthogonal) dictionaries via  $l_1$  minimization, *Proc. Natl. Acad. Sci.* 100 (5) (2003) 2197–2202.
- [30] A. Maleki, D.L. Donoho, Optimally tuned iterative reconstruction algorithms for compressed sensing, *IEEE J. Sel. Top. Signal Process.* 4 (2) (2010) 330–341.
- [31] R. Gribonval, M. Nielsen, Sparse representations in unions of bases, *IEEE Trans. Inf. Theory* 49 (12) (2003) 3320–3325.
- [32] L.R. Welch, Lower bounds on the maximum cross correlation of signals, *IEEE Trans. Inf. Theory* 20 (3) (1974) 397–399.
- [33] E.J. Candes, Y. Plan, Near-ideal model selection by  $l_1$  minimization, *Ann. Stat.* 37 (2009) 2145–2177.
- [34] T.T. Cai, G. Xu, J. Zhang, On recovery of sparse signals via  $l_1$  minimization, *IEEE Trans. Inf. Theory* 55 (7) (2009) 3388–3397.
- [35] S. Foucart, Sparse recovery algorithms: sufficient conditions in terms of restricted isometry constants, in: M. Neantu, L. Schumaker (Eds.), in: *Proceedings of the 13th International Conference on Approximation Theory*, San Antonio, TX, Springer, 2010.