Research Article

Approximately Normalized Iterative Hard Thresholding for Nonlinear Compressive Sensing

Xunzhi Zhu

School of Science, Shandong University of Technology, Zibo 255000, China

Correspondence should be addressed to Xunzhi Zhu; zhxzh1997@126.com

Received 1 April 2016; Revised 4 July 2016; Accepted 20 July 2016

Academic Editor: Cornel Ioana

Copyright © 2016 Xunzhi Zhu. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The nonlinear compressive sensing (NCS) is an extension of classical compressive sensing (CS) and the iterative hard thresholding (IHT) algorithm is a popular greedy-type method for solving CS. The normalized iterative hard thresholding (NIHT) is a modification of IHT and is more effective than IHT. In this paper, we propose an approximately normalized iterative hard thresholding (ANIHT) algorithm for NCS by using the approximate optimal stepsize combining with Armijo stepsize rule preiteration. Under the condition similar to restricted isometry property (RIP), we analyze the condition that can identify the iterative support sets in a finite number of iterations. Numerical experiments show the good performance of the new algorithm for the NCS.

1. Introduction

Compressed sensing (CS) [1, 2] deals with the problem of recovering sparse signals from underdetermined linear measurements. In recent years, it has attracted considerable attention in areas of signal processing, electrical engineering, computer science, and applied mathematics; see [3, 4]. However, many real-life applications in physics and biomedical sciences carry some strongly nonlinear structure, so that the linear model is not suited anymore, even as an approximation. It is of utmost interest to investigate compressive sensing with nonlinear measurements, which is called nonlinear compressive sensing (NCS) [5, 6]. So, it is necessary to consider the following NCS: to find a vector $x \in \mathbb{R}^n$ from the observations $b \in \mathbb{R}^m$ given by

$$
\Phi(x) + \eta = b, \quad \text{s.t.} \quad \|x\|_0 \leq s,
$$

where $\Phi(x) = (\phi_1(x), \ldots, \phi_m(x))^\top : \mathbb{R}^n \to \mathbb{R}^m$ is nonlinear, $\eta$ is some noise term and $s < m < n$, and $\|x\|_0$ is the $l_0$-norm of $x$, which refers to the number of nonzero elements in the vector $x$. The optimization problem associated with (1) is

$$
\min f(x) = \frac{1}{2} \|\Phi(x) - b\|^2, \quad \text{s.t.} \quad \|x\|_0 \leq s,
$$

where $\|\cdot\|$ is $l_2$-norm and $r(x) \triangleq \Phi(x) - b$ is the residual function. Let $\Phi(x)$ be continuously differentiable and let $J(x) \in \mathbb{R}^{m \times n}$ denote its Jacobian matrix. Then $\forall f(x) = J(x)^\top r(x)$. Clearly, if $\Phi(x)$ is linear function, model (1) and the optimization problem (2) reduce to classical CS,

$$
Ax + e = b, \quad \text{s.t.} \quad \|x\|_0 \leq s,
$$

and problem

$$
\min \frac{1}{2} \|Ax - b\|^2, \quad \text{s.t.} \quad \|x\|_0 \leq s,
$$

respectively, where $A \in \mathbb{R}^{m \times n}$.

Greedy methods have already proven useful and efficient to tackle (4) [7]. A variety of greedy methods have been proposed to solve (4), such as matching pursuit (MP) [8], orthogonal MP (OMP) [9], compressive sampling matching pursuit (CoSaMP) [10], subspace pursuit (SP) [11], hard thresholding pursuit (HTP) [12], and conjugate gradient iterative hard thresholding (CGIHT) [13].

There is another greedy method, iterative hard thresholding (IHT) algorithm for problem (4), which was proposed by
Blumensath and Davies in [14, 15]. When matrix $A$ is row full-rank and the spectral norm $\|A\|_2 < 1$, IHT converges to a local minimum of (4) [14]. In [16], the authors showed that the numerical studies of IHT are not very promising and the algorithm often fails to converge when the conditions fail. Then they gave normalized IHT (NIHT) with an adaptive stepsize and line search and proved that it converges to a local minimum if $A$ is row full-rank and $s$-regular, where $s$-regular means that any $s$ columns of $A$ are linear independent [17]. Caris and Thompson [18] showed NIHT converges to local minimum if matrix $A$ is $2s$-regular. Blumensath [5] showed that IHT can recover signals from NCS under conditions similar to those required in CS.

Inspired by these works, we propose an approximately NIHT (ANIHT) algorithm to solve the NCS problem (2). Since problem (2) is in general a nonconvex programming, we can only expect to find out the stationary point rather than the local minimizer, which is different from NIHT for CS. Then we show that the accumulation point of the ANIHT algorithm is the stationary point. With the nondegeneracy and strict complementarity of stationary point, the support sets of the sequence are identified in a finite number of iterations. At last, we simulate several experiments to demonstrate the effectiveness of the algorithm.

This paper is organized as follows. Section 2 gives the ANIHT algorithm for (2) and proves its convergence properties. Numerical results are given in Section 3. The last section makes some concluding remarks.

### 2. Algorithm

In this section, we will present the approximately normalized iterative hard thresholding (ANIHT) algorithm for (2) and then analyze its convergence properties. Denote

$$S \triangleq \{x \in \mathbb{R}^n \mid \|x\|_0 \leq s\}. \quad (5)$$

For $x \in \mathbb{R}^n$ and set $A \subseteq \mathbb{R}^n$, projector onto $A$ is $P_A(x) \triangleq \arg \min_{y \in A} \|x - y\|$. Note that the projection onto sparse set $S$, written as $P_S(\cdot)$, sets all but $s$ largest absolute value components of $x$ to zero. The definition of $L$-stationarity was proposed in [17] based on the notion of fixed-point equation.

**Definition 1.** The point $x^* \in S$ is called the stationary point of problem (2), if it holds that

$$x^* \in P_S(x^* - \alpha \nabla f(x^*)). \quad (6)$$

Note that $x^* \in S$ is the stationary point of problem (2) if and only if [17]

$$\left| \nabla_i f(x^*) \right| \leq \begin{cases} 0, & i \in \Gamma^* \\ \frac{1}{\alpha} M_i(x^*), & i \notin \Gamma^* \end{cases}, \quad (7)$$

where $M_i(x)$ denotes the $i$th largest element in absolute value of $x \in S$ and $\Gamma^* = \{i \in \{1, \ldots, n\} \mid x_i^* \neq 0\}$.

In NIHT for CS [16], to guarantee the objective function a sufficient decrease per iteration, the authors added a stepsize strategy based on restricted isometry property (RIP) [1]. In ANIHT for NCS (2), we use the approximately optimal stepsize to accelerate the convergence and use Armijo-type stepsize and make a sufficient decrease of the objective function directly without RIP. Here $(\cdot)_\Gamma$ is the subvector (submatrix) obtained by discarding all but the elements (columns) in $\Gamma$. The framework of ANIHT is described as follows.

**Step 1.** Initialize $x^0, \alpha_0 > 0, \sigma > 0, 0 < \beta < 1$.

**Step 2.** Let $\Gamma^k = \text{supp}(x^k)$ and compute

$$\bar{x}^{k+1} = P_S(x^k - \alpha_k^0 J (x^k)^T r(x^k)), \quad (8)$$

where

$$\alpha_k^0 = \min \left\{ \alpha_0, \frac{\| J (x^k)^T r(x^k) \|_2^2}{\| J (x^k)^T r(x^k) \|_1^2} \right\}. \quad (9)$$

**Step 3.** If $\text{supp}(\bar{x}^{k+1}) = \Gamma^k$, then $x^{k+1} = \bar{x}^{k+1}$, $\Gamma^{k+1} = \text{supp}(x^{k+1})$, and $\alpha_k = \alpha_k^0$; else compute

$$x^{k+1} = P_S(x^k - \alpha_k J (x^k)^T r(x^k)) \quad (10)$$

and $\Gamma^{k+1} = \text{supp}(x^{k+1})$, where $\alpha_k = \alpha_k^0 \beta^m$ and $m_k$ is the smallest positive integer $m$ such that

$$\| r(x^k (\alpha_k^0 \beta^m)) \|_2^2 \leq \| r(x^k) \|_2^2 - \sigma \| x^k (\alpha_k^0 \beta^m) - x^k \|_2^2, \quad (11)$$

and $x^k(\alpha) = P_S(x^k - \alpha J (x^k)^T r(x^k))$.

**Step 4.** If the stopping criterion is met, then stop.

**Remark 2.** We now briefly illustrate the algorithm:

(i) In Step 2, a proposal point $x^{k+1}$ is calculated, whether accepting it depends on the relationship between its support set and the previous point’s; see Step 3. In addition, to accelerate the rate of convergence, we utilize the approximately optimal initial stepsize to compute $x^{k+1}$. The linear approximate of $\Phi(x^k - \alpha J (x^k)^T r(x^k))$ at $x^k$ is

$$\Phi \left( x^k - \alpha J (x^k)^T r(x^k) \right) \approx \Phi(x^k) - \alpha J (x^k)^T r(x^k). \quad (12)$$
\[ \Gamma_k \text{ is the support of the best } s \text{ term approximation to } b \text{ at the current iteration. So we obtain that} \]
\[ \arg\min \left\{ \Phi \left( x^k \right) - \alpha \left( J(x^k) \right)^\top \left( I(x^k)^\top r(x^k) \right)_{\Gamma_k} - b \right\}, \alpha > 0 \right\} = \frac{\left\| \left( J(x^k)^\top r(x^k) \right)_{\Gamma_k} \right\|^2}{\left\| J(x^k)^\top r(x^k) \right\|_{\Gamma_k}}. \quad (13) \]

This stepsize is in accordance with the optimal stepsize in NIHT for CS in [16]. Furthermore, by Assumption 4, when \( \alpha \) is relatively small, the error introduced in the linear approximation is small; then the objective function decreases if the support set is not changed.

(ii) Armijo-type stepsize rule in Step 3 makes the choice of stepsize and support set adaptively and sufficient decrease of the objective function meanwhile per iteration. It is well defined by Lemma 6.

The following assumptions are chosen to ensure the descent property (14) of the objective function (1/2)||r(x)||².

**Assumption 3.** There exists a constant \( U_{2s} > 0 \) such that \( \| J(x)(x - y) \| \leq U_{2s} \| x - y \| \) for \( |\text{supp}(x)\cup\text{supp}(y)| \leq 2s \).

We also need the assumption that the Jacobian \( J(\cdot) \) of residual \( r(\cdot) \) is restricted Lipschitz continuous on \( \mathbb{R}^n \).

**Assumption 4.** There exists a constant \( \mathcal{J}_{2s} > 0 \) such that \( \| J(x) - J(y) \| \leq \mathcal{J}_{2s} \| x - y \| \) for \( |\text{supp}(x)\cup\text{supp}(y)| \leq 2s \).

**Lemma 5.** Suppose that Assumptions 3 and 4 hold. If there exists \( \rho > 0 \) for \( x \) and the iterative point \( x^k \) of ANIHT satisfying \( |\text{supp}(x)\cup\text{supp}(x^k)| \leq 2s \) and \( \| x - x^k \| \leq 2\rho \), then
\[ \| r(x) \|^2 \leq \| r(x^k) \|^2 + 2 \left< J(x^k)^\top r(x^k), x - x^k \right> \]
\[ + \bar{L}_{2s} \| x - x^k \|^2, \]

where \( \bar{L}_{2s} \equiv U_{2s}^2 + \mathcal{J}_{2s} \| r(x^0) \| + \rho U_{2s} \mathcal{J}_{2s} \) and \( x^0 \) is the initial point of ANIHT.

**Proof.** We first show
\[ \left< J(x)^\top r(x) - J(x^k)^\top r(x^k), x - x^k \right> \leq \bar{L}_{2s} \| x - x^k \|^2. \quad (15) \]

Since ANIHT algorithm generates monotonically decreasing function values, then \( \| r(x^k) \| \leq \| r(x^0) \| \) for all \( k \). Direct calculation yields that
\[ \left< J(x)^\top r(x) - J(x^k)^\top r(x^k), x - x^k \right> = \left< J(x)^\top r(x) - J(x^k)^\top r(x^k) + J(x)^\top r(x^k) - J(x^k)^\top r(x^k), x - x^k \right> \]
\[ = \left< J(x)^\top r(x) - J(x^k)^\top r(x^k), x - x^k \right> + \left< J(x^k)^\top r(x^k), x - x^k \right> \]
\[ \leq 2 \int_0^1 \left< J(x^k + t(x - x^k))^\top r(x^k + t(x - x^k)), x - x^k \right> dt \leq 2 \int_0^1 t \bar{L}_{2s} \| x \]
\[
- x^k \|^2 dt = \bar{L}_{2s} \| x - x^k \|^2,
\]

which completes the proof. □

**Lemma 6.** Suppose that Assumptions 3 and 4 hold and \( x^k \) is the iterative point of ANIHT with \( 0 < \alpha_0 < 1/\bar{L}_{2s} \). Then
\[ \|r(x^k(\alpha))\|^2 \leq \begin{cases} \|r(x^k)\|^2 - \left(\frac{1}{\alpha_0} - \tilde{L}_{2s}\right)\|x^k(\alpha) - x^k\|^2, & \text{if } 0 < \alpha \leq \alpha_0, \\ \|r(x^k)\|^2 - \sigma\|x^k(\alpha) - x^k\|^2, & \text{if } 0 < \alpha \leq \frac{1}{L_{2s} + \sigma}. \end{cases} \] (18)

Therefore, \( \alpha_k \) is well defined.

**Proof.** According to the computation in Step 2, we have

\[ x^k(\alpha) \in \arg \min \left\{ \|x - x^k + \alpha J(x^k)^T r(x^k)\|^2, \|x\|_0 \leq s \right\}, \] (19)

which implies that

\[ \|x^k(\alpha) - x^k + \alpha J(x^k)^T r(x^k)\|^2 \leq \|\alpha J(x^k)^T r(x^k)\|^2, \] (20)

that is,

\[ \|x^k(\alpha) - x^k\|^2 \leq -2\alpha \left\langle J(x^k)^T r(x^k), x^k(\alpha) - x^k \right\rangle. \] (21)

If \( \text{supp}(\tilde{x}^{k+1}) = \text{supp}(x^k) \), then the above inequality and the monotonicity of \( \{r(x^k)\} \) yield that

\[ \|x^k(\alpha) - x^k\|^2 \leq -2\alpha \left\langle J(x^k)^T r(x^k), x^k(\alpha) - x^k \right\rangle \leq 2\alpha \|r(x^k)\| \cdot \|J(x^k)(x^k(\alpha) - x^k)\| \leq 2\alpha_0 \|r(x^0)\| U_{2s} \|x^k(\alpha) - x^k\|. \] (22)

Otherwise, by the Armijo-type stepsize rule and the monotonicity of \( \{r(x^k)\} \), we have

\[ \|x^k(\alpha) - x^k\|^2 \leq \frac{1}{\sigma} \left( \|r(x^0)\|^2 - \|r(x^k(\alpha))\|^2 \right) \leq \frac{1}{\sigma} \|r(x^0)\|^2. \] (23)

Then \( \|x^k(\alpha) - x^k\| \) can be smaller than \( \max\{2\alpha_0 U_{2s}, 1/\sqrt{\sigma}\} \cdot \|r(x^0)\| \).

By Lemma 5 and (21), we get that

\[ \|r(x^k(\alpha))\|^2 \leq \|r(x^k)\|^2 + 2 \left\langle J(x^k)^T r(x^k), x^k(\alpha) - x^k \right\rangle \]

where \( \alpha_k \geq \min\{\alpha_0, 1/U_{2s}\} \).

If \( \text{supp}(\tilde{x}^{k+1}) = \text{supp}(x^k) \), \( \alpha_k \leq \alpha_0 < 1/L_{2s} \), then \( 1/\alpha - \tilde{L}_{2s} > 0 \). Otherwise, by letting \( 1/\alpha - \tilde{L}_{2s} \geq \sigma \), we can obtain the desired result by the definition of \( \alpha \).

\[ \sqrt{1/\alpha - \tilde{L}_{2s}} \leq \tilde{L}_{2s} + \sigma \]

Therefore, \( \alpha_k \) is well defined.

**Proof.** According to the computation in Step 2, we have

\[ x^k(\alpha) \in \arg \min \left\{ \|x - x^k + \alpha J(x^k)^T r(x^k)\|^2, \|x\|_0 \leq s \right\}, \] (19)

which implies that

\[ \|x^k(\alpha) - x^k + \alpha J(x^k)^T r(x^k)\|^2 \leq \|\alpha J(x^k)^T r(x^k)\|^2, \] (20)

that is,

\[ \|x^k(\alpha) - x^k\|^2 \leq -2\alpha \left\langle J(x^k)^T r(x^k), x^k(\alpha) - x^k \right\rangle. \] (21)

If \( \text{supp}(\tilde{x}^{k+1}) = \text{supp}(x^k) \), then the above inequality and the monotonicity of \( \{r(x^k)\} \) yield that

\[ \|x^k(\alpha) - x^k\|^2 \leq -2\alpha \left\langle J(x^k)^T r(x^k), x^k(\alpha) - x^k \right\rangle \leq 2\alpha \|r(x^k)\| \cdot \|J(x^k)(x^k(\alpha) - x^k)\| \leq 2\alpha_0 \|r(x^0)\| U_{2s} \|x^k(\alpha) - x^k\|. \] (22)

Otherwise, by the Armijo-type stepsize rule and the monotonicity of \( \{r(x^k)\} \), we have

\[ \|x^k(\alpha) - x^k\|^2 \leq \frac{1}{\sigma} \left( \|r(x^0)\|^2 - \|r(x^k(\alpha))\|^2 \right) \leq \frac{1}{\sigma} \|r(x^0)\|^2. \] (23)

Then \( \|x^k(\alpha) - x^k\| \) can be smaller than \( \max\{2\alpha_0 U_{2s}, 1/\sqrt{\sigma}\} \cdot \|r(x^0)\| \).

By Lemma 5 and (21), we get that

\[ \|r(x^k(\alpha))\|^2 \leq \|r(x^k)\|^2 + 2 \left\langle J(x^k)^T r(x^k), x^k(\alpha) - x^k \right\rangle \]

where \( \alpha_k \geq \min\{\alpha_0, 1/U_{2s}\} \).

If \( \text{supp}(\tilde{x}^{k+1}) = \text{supp}(x^k) \), \( \alpha_k \leq \alpha_0 < 1/L_{2s} \), then \( 1/\alpha - \tilde{L}_{2s} > 0 \). Otherwise, by letting \( 1/\alpha - \tilde{L}_{2s} \geq \sigma \), we can obtain the desired result by the definition of \( \alpha \).

\[ \sqrt{1/\alpha - \tilde{L}_{2s}} \leq \tilde{L}_{2s} + \sigma \]
Therefore, \( \alpha_k \) is bounded from below by a positive constant \( \min \{ \alpha_0, \beta/(\tilde{L}^2s + \sigma) \} \). We can conclude that
\[
\lim_{k \to \infty} \frac{x^{k+1} - x^k}{\alpha_k} = 0. \quad (28)
\]

(ii) Suppose that \( x^* \) is an accumulation point of the sequence \( \{x^k\} \); then there exists a subsequence \( \{x^{k_j}\} \) converging to \( x^* \) and \( \lim_{j \to \infty} x^{k_j+1} = x^* \) by (i). For the simplification of notation, denote \( \alpha_k \) as \( \alpha \) by its boundedness and \( \Gamma^* = \text{supp}(x^*) \). Based on
\[
x^{k+1} = P_S\left( x^k - \alpha (f(x^k)^T r(x^k)) \right) \quad (29)
\]
in Step 2, we consider two cases.

Case 1 \((i \in \Gamma^*)\). The convergence of \( \{x^k\} \) and \( \{x^{k_j}\} \) guarantees that, for some \( n_1 > 0 \), \( x^{k_j+1} > 0 \) for all \( j > n_1 \). The definition of the projection on \( S \) shows that
\[
x^{k_j+1} = x^{k_j} - \alpha (f(x^{k_j})^T r(x^{k_j}))_i. \quad (30)
\]
Taking \( j \to \infty \), we have \((f(x^*)^T r(x^*))_i > 0 \).

Case 2 \((i \notin \Gamma^*)\). If there exists an \( n_2 > 0 \) such that, for all \( j > n_2 \), \( x^{k_j+1} > 0 \), the projection implies that
\[
\left| x^{k_j} - \alpha (f(x^{k_j})^T r(x^{k_j}))_i \right| \leq M_s(x^{k_j+1}). \quad (31)
\]
Letting \( j \to \infty \) and exploiting the continuity of the function \( M_s \), we obtain that
\[
\left| \alpha (f(x^*)^T r(x^*))_i \right| \leq M_s(x^*). \quad (32)
\]
On the other hand, if there exists an infinite number of indices of \( k_j \) for \( x^{k_j+1} > 0 \), as the same proof in Case 1, it follows that \((f(x^*)^T r(x^*))_i = 0 \). Since \( \alpha_k \) is bounded from below by a positive constant, we have
\[
\left| (f(x^*)^T r(x^*))_i \right| = 0, \quad \text{if } i \in \Gamma^*, \quad (33)
\]
\[
\leq \frac{1}{\alpha} M_s(x^*), \quad \text{if } i \notin \Gamma^*,
\]
which means \( x^* \) is a stationary point of (2).

We are now ready to show that under suitable conditions the support set of a point is identified in a finite number of iterations. We can easily verify that if \( \|x^*\|_0 = s \), then the support set of \( x^* \) in a sufficiently small neighborhood of \( x^* \) is identified. For \( \|x^*\|_0 < s \), we introduce the concept of strict complementarity to identify the support set.

**Definition 8.** The point \( x^* \in S \) is called nondegenerate if \( \|x^*\|_0 = s \). The condition
\[
\left\{ (f(x^*)^T r(x^*))_i \right\} = \begin{cases} 0, & \text{if } i \in \Gamma^* \\ \leq \frac{1}{\alpha} M_s(x^*), & \text{if } i \notin \Gamma^*, \end{cases} \quad (34)
\]
is called strict complementarity condition of (2), where \( \Gamma^* = \text{supp}(x^*) \).

**Theorem 9.** For any sequence \( \{x^k\} \) converging to \( x^* \), we have the following:

(i) if \( x^* \) is nondegenerate point, then
\[
\text{supp}(x^k) = \text{supp}(x^*) \quad (35)
\]
for all \( k \) sufficiently large and
\[
\lim_{k \to \infty} \left\| (f(x^k)^T r(x^k))_i \right\| = 0, \quad (36)
\]
where \( I_z^k = \text{supp}(x^k) \) and \( I_z^* = \text{supp}(x^*) \).

(ii) if \( x^* \) satisfies strict complementarity, then (35) holds if and only if (36) holds.

**Proof.** (i) Suppose that \( x^* \) is a nondegenerate point. We have (35) when \( z^k \) is sufficiently close to \( z^* \). By (35) and the continuity of \( f(\cdot)^T r(\cdot) \), we can easily get that
\[
\lim_{k \to \infty} \left\| (f(x^k)^T r(x^k))_i \right\| = \lim_{k \to \infty} \left\| (f(z^k)^T r(z^k))_i \right\| = 0. \quad (37)
\]

(ii) Suppose that \( z^* \) satisfies strict complementarity condition. The "only if" part can be obtained by the similar proof as in (i). Let (36) hold. Since \( z^k \to z^* \), \( I_z^k \subseteq \Gamma_z^* \). Assume that there is an infinite subsequence \( K \) and an index \( j \) such that \( j \notin \Gamma_z^* \) and \( j \in I_z^k \) for all \( k \in K \). By (36),
\[
\left| (f(z^k)^T r(z^k))_j \right| = \lim_{k \to \infty} \left| (f(z^k)^T r(z^k))_j \right| = 0. \quad (38)
\]
While the strict complementarity condition (34) implies that \((f(z^*)^T r(z^*))_j \neq 0 \). This contradiction proves that \( \Gamma_z^* = I_z^* \).

**3. Numerical Experiments**

In this part, sensor localization problem and phase retrieval problem will be stimulated. In both examples, the stop criteria will be set as \( \| (f(x^k)^T r(x^k))_i \| \leq e_2 \), where \( e_2 \) is pretty small in different cases or the maximum iterative times being equal to 5000.

**Sensor localization problem** can be described as follows: given \( M \) known anchors \( p^1, p^2, \ldots, p^M \in \mathbb{R}^n \), the purpose is to find a sensor \( x \in \mathbb{R}^n \) satisfying
\[
\| x - p^i \|^2 + \eta_i = b_i, \quad i = 1, \ldots, M, \quad (39)
\]
\[
\| x \|_0 \leq s,
\]
where \( \eta_i, i = 1, \ldots, M \), is the noise (which obeys the normal distribution with zero expectation and \( \sigma^2 \) variance here). The problem of finding an \( x \in \mathbb{R}^n \) satisfying above equalities is the same as finding an optimal solution to the optimization problem (4) with \( f(x) = \sum_{i=1}^M (\| x - p^i \|^2 - b_i^2) \). We first
compare the ANIHT on an example with $M = 80$, $N = 120$, and $s = 1, 2, \ldots, 10$. Each component of the $M$ vectors $p^1, \ldots, p^M$ was randomly and independently generated from a standard normal distribution. Then the true vector $x_{\text{orig}}$ and $b$ are designed as following MATLAB codes:

$$x_{\text{orig}} = \text{zeros}(n, 1);$$

$$T = \text{randperm}(n);$$

$$x_{\text{orig}}(T(1: s)) = 10 \ast \text{rand}(s, 1);$$

$$b_i = \|x_{\text{orig}} - p^i\|^2 + \sigma_0 \ast \text{randn}(1, 1).$$

(40)

For each value of $s$ ($s = 1, 2, \ldots, 10$), we ran ANIHT algorithm from 100 different and randomly generated initial data sets. The numbers of runs of 100 in which the methods found the “correct” solution are given in Table 1. Here, the “correct” solution $x$ ($\|x\|_0 \leq s$) means that $f(x) \leq f(x_{\text{orig}})$, or $x$ and $x_{\text{orig}}$ are very close, say

$$\frac{\|x - x_{\text{orig}}\|}{\|x\|} < 10^{-3}. \quad (41)$$

As can be clearly seen by the results in the table, the ANIHT performs well in terms of the success probability. For more details, when the “true” solutions are quite sparse compared to the dimension $n$, ANIHT can almost recover all the “true” solutions, while the performance is becoming worse as $s$ rises.

Then we run ANIHT algorithm in a higher dimensional data set, where $N = 2M$, $s = \lfloor 0.01N \rfloor$, and $M = 100, 200, \ldots, 2000$. For each data set, we run 40 times and record the average results (in which the unsuccessful recoveries are expelled). Figure 1 shows the performance of ANIHT when addressing this problem.

Phase retrieval is to recover a signal from the magnitude of its Fourier transform, or of any other linear transform. Due to the loss of Fourier phase information, the problem is generally ill-posed. The phase retrieval problem can be described as follows: given $M$ known measurement vectors $p^1, p^2, \ldots, p^M \in \mathbb{R}^n$, the purpose is to reconstruct a signal $x \in \mathbb{R}^n$ satisfying

$$\left| \langle x, p^i \rangle \right|^2 + \eta_i = b_i, \quad i = 1, \ldots, M,$$

(42)

$$\|x\|_0 \leq s,$$

where $p^i$ is the $i$th column of the general matrix or the discrete Fourier transform (DFT) and $\eta_i, i = 1, \ldots, M$, is the noise (which obeys the normal distribution with zero expectation and $\sigma_0^2$ variance here). Also this problem is equivalent to recover an optimal solution to the optimization problem (4) with

$$f(x) = \sum_{i=1}^{M} |\langle x, A_i x - b_i \rangle|^2,$$

where $A_i = p^i p^{i\top}, i = 1, \ldots, M$.

There are some other methods for sparse phase retrieval and the codes are available. So we can compare our ANIHT algorithm with them. We first compare the ANIHT algorithm with the partial sparse-simplex method (PSS) and greedy sparse-simplex (GSS) method in [17] with $M = 80$, $N = 120$, and $s = 2, 3, \ldots, 9$ which is identical to those in [17]. The true vector $x_{\text{orig}}$ and the measurement vectors $p^1, p^2, \ldots, p^M$ are generated as that produced in sensor localization problem; $b$ is designed as following MATLAB codes:

$$b_i = \left| \langle x_{\text{orig}}, p^i \rangle \right|^2 + \sigma_0 \ast \text{randn}(1, 1). \quad (43)$$

For each value of $s$ ($s = 2, 3, \ldots, 9$), we ran ANIHT algorithm from 100 different and randomly generated initial data sets.
The numbers of runs of 100 in which the methods found the “correct” solution are given in Table 2. As can be clearly seen by the results in the table, the ANIHT outperforms PSS and GSS in terms of the success probability. What is more, the data in the row with \( \sigma_0 = 0 \) are higher than those with \( \sigma_0 = 0.01 \) and \( \sigma_0 = 0.1 \).

We also compare our ANIHT algorithm with GESPAR in [19] to recover a signal from the magnitude of its Fourier transform. Namely, it is to find a real-valued discrete time signal \( x \in \mathbb{R}^N \) from its magnitude-squared value of an \( N \) point discrete Fourier transform (DFT):

\[
b_j = \left| \sum_{k=1}^{N} x_k e^{-2\pi i (j-1)(k-1)/N} \right|^2, \quad j = 1, \ldots, N.
\]

We denote by \( F \) the DFT matrix; then elements \( F_{jk} = e^{-2\pi i (j-1)(k-1)/N} \) and \( b = |Fx|^2 \), where \( |\cdot|^2 \) denotes the element-wise absolute-squared value. We get \( b \) by the pseudo MATLAB codes:

\[
b = \text{abs} \left( \text{fft} \left( x_{\text{orig}} \right) \right) .^2 + \sigma_0 \ast \text{randn} \left( N, 1 \right).
\]

To see the accuracy of the solutions and the speed of these two methods, we run the two methods for \( n \) increasing from 512 to 3072 and keeping \( N = 2n, s = 1\%n \). We also test them under noiseless and two noise levels, \( \sigma_0 = 0.01 \) and \( \sigma_0 = 0.1 \). From Table 3, we can see that ANIHT outperforms GESPAR in terms of both average CPU time and average relative error for large \( n \) (\( n \geq 2048 \)).

4. Conclusion

Nonlinear CS (NCS) not only is of academic interest but also might be important in many real-world applications when the measurements cannot be designed to be perfectly linear. In this paper, we have proposed an ANIHT algorithm for NCS and studied its convergence. We have showed that any accumulation point of the algorithm is the stationary point. The support set of the sequence can be identified with the assumption of nondegeneracy and strict complementarity of stationary point. The numerical experiments show that ANIHT algorithm is effective for NCS. In the future, we will further consider other methods for nonlinear least square problem to improve the rate of convergence, such as L-M method or cubic regularization methods [20].

Competing Interests

The author declared that there are no competing interests in their submitted paper.

Acknowledgments

This research was supported by National Natural Science Foundation of China (11271233) and Shandong Province Natural Science Foundation (ZR2012AM016).

References


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