MINIMIZATION OF $\ell_{1-2}$ FOR COMPRESSED SENSING*

PENGHANG YIN, YIFEI LOU, QI HE, AND JACK XIN †

Abstract. We study minimization of the difference of $\ell_1$ and $\ell_2$ norms as a non-convex and Lipschitz continuous metric for solving constrained and unconstrained compressed sensing problems. We establish exact (stable) sparse recovery results under a restricted isometry property (RIP) condition for the constrained problem, and a full-rank theorem of the sensing matrix restricted to the support of the sparse solution. We present an iterative method for $\ell_{1-2}$ minimization based on the difference of convex functions algorithm (DCA), and prove that it converges to a stationary point satisfying first order optimality condition. We propose a sparsity oriented simulated annealing (SA) procedure with non-Gaussian random perturbation and prove the almost sure convergence of the combined algorithm (DCASA) to a global minimum. Computation examples on success rates of sparse solution recovery show that if the sensing matrix is ill-conditioned (non RIP satisfying), then our method is better than existing non-convex compressed sensing solvers in the literature. Likewise in the magnetic resonance imaging (MRI) phantom image recovery problem, $\ell_{1-2}$ succeeds with 8 projections. Irrespective of the conditioning of the sensing matrix, $\ell_{1-2}$ is better than $\ell_1$ in both the sparse signal and the MRI phantom image recovery problems.

Key words. $\ell_{1-2}$ minimization, compressed sensing, non-convex, difference of convex functions algorithm, simulate annealing

AMS subject classifications. 90C26, 65K10, 49M29


1. Introduction. Compressed sensing (CS) has been a rapidly growing field of research in signal processing and mathematics stimulated by the foundational papers [8, 6, 22, 23] and related Bregman iteration methods [57, 33]. A fundamental issue in CS is to recover an $n$-dimensional vector $\bar{x}$ from $m \ll n$ measurements (the projection of $\bar{x}$ onto $m$ $n$-dimensional vectors), or in matrix form given $b = A\bar{x}$, where $A$ is the so-called $m \times n$ sensing (measurements) matrix. One can also view $\bar{x}$ as coefficients of a sparse linear representation of data $b$ in terms of redundant columns of matrix $A$ known as dictionary elements.

The conditioning of $A$ is related to its restricted isometry property (RIP) as well as the coherence (maximum of pairwise mutual angles) of the column vectors of $A$. Breakthrough results in CS have established when $A$ is drawn from a Gaussian matrix ensemble or random row sampling without replacement from an orthogonal matrix (Fourier matrix), then $A$ is well-conditioned in the sense that if $\bar{x}$ is $s$-sparse ($s$ is much less than $n$), $m = O(s \log n)$ measurements suffice to recover $\bar{x}$ (the sparsest solution) with an overwhelming probability by $\ell_1$ minimization or the basis pursuit (BP) problem [6, 15]:

$$\ell_1$$

$$\min_{\bar{x}} \|x\|_1 \text{ subject to } Ax = b.$$ 

In the above formulation, $\ell_1$ norm works as the convex relaxation of $\ell_0$ that counts the nonzeros. Such a matrix $A$ has incoherent column vectors. On the other hand, if columns of $A$ are coherent enough, such as those arising in discretization of continuum imaging problems (radar and medical imaging) when the grid spacing is below the Rayleigh threshold [25], $\ell_1$ minimization may not give the sparsest solution [25, 55].

*The work was partially supported by NSF grants DMS-0928427 and DMS-1222507.
†Department of Mathematics, UC Irvine, Irvine, CA 92697, USA. (penghany@uci.edu, ylou1@math.uci.edu, qhe2@uci.edu, jxin@math.uci.edu)
In the past decade, great efforts have been devoted to explore efficient and stable algorithms for solving BP problem and its associated $\ell_1$-regularized problem (also called lasso [47]):

\begin{equation}
\min_x \frac{1}{2} \|Ax - b\|_2^2 + \lambda \|x\|_1,
\end{equation}

where $\lambda > 0$ is a free parameter. Bregman iterative method, now known to be equivalent to the augmented Lagrangian method, was proposed to solve the BP problem by Yin et al. [57]. There are many state-of-the-art algorithms available for lasso problem (1.2), such as the split Bregman [33], being equivalent to ADMM [5, 27], FPC [34], FISTA [2] among others [49, 54, 58, 50].

Non-convex (concave) functions, such as $\ell_p$ (quasi-)norm ($p < 1$) [12, 13] and log-det functional [10], have also been proposed as alternatives to $\ell_0$. Such non-Lipschitz continuous metrics usually require additional smoothing in minimization to avoid division by zero and to enhance sparsity. Besides, a general class of penalty functions satisfying unbiasedness, sparsity and continuity can be found in [24, 41]. While non-convex metrics are generally more challenging to minimize, they have advantages over the convex $\ell_1$ norm. Simply put, non-convex CS enables one to reconstruct the sparse signal of interest from substantially fewer measurements.

On the computational side, researchers have observed that under certain conditions on the sensing matrix $A$ (e.g. when columns of $A$ are sufficiently randomized), several non-convex CS solvers do produce solutions of better quality [10, 14, 38, 19, 26], even though none of them theoretically guarantees convergence to a global minimum. Algorithms that directly tackle the $\ell_0$ minimization include Compressive Sampling Matching Pursuit (CoSaMP) [43] which is a greedy method amongst variants of orthogonal matching pursuit (OMP) [48], iterative hard thresholding (IHT) algorithm [4, 3] and penalty decomposition method [42], whereas iteratively reweighted least squares (IRLS) [19, 14, 38] and iteratively reweighted $\ell_1$ (IRL1) [17, 61, 10, 26] can be applied to minimize non-convex proxies of $\ell_0$. Particularly for minimization of $\ell_p$ norm, empirical studies [14, 53] show that whenever $p \in [1/2, 1)$, the smaller the $p$, the sparser the solutions by minimizing $\ell_p$ norm. On the other hand, for $p \in (0, 1/2)$, the performance has no significant improvement as $p$ decreases. Xu et al. thus proposed an iterative half thresholding algorithm [52, 53, 59] specifically for solving $\ell_{1/2}$ regularization.

In this paper, we study minimization of the non-convex yet Lipschitz continuous metric $\ell_{1-2}$, for sparse signal recovery and compare it with various CS solvers. $\ell_{1-2}$ was first addressed in [28] by Esser et al. in the context of nonnegative least squares problems and group sparsity with applications to spectroscopic imaging. A contour plot of $\ell_{1-2}$ metric can be seen in Fig. 1. Here we mainly discuss the constrained $\ell_{1-2}$ minimization problem

\begin{equation}
\min_{x \in \mathbb{R}^n} \|x\|_1 - \|x\|_2 \text{ subject to } Ax = b
\end{equation}

and the unconstrained one

\begin{equation}
\min_{x \in \mathbb{R}^n} \frac{1}{2} \|Ax - b\|_2^2 + \lambda (\|x\|_1 - \|x\|_2),
\end{equation}

where $A \in \mathbb{R}^{m \times n}$ is an under-determined sensing matrix of full row rank and $b \in \mathbb{R}^m \setminus \{0\}$. We shall focus on the theoretical aspects such as sparsity of minimizers and convergence of minimization algorithms, and refer to the companion paper [40] for more extensive computational study with applications to imaging problems.

The rest of the paper is organized as follows. After presenting preliminaries, we prove an exact (stable) sparse recovery theorem via the constrained $\ell_{1-2}$ minimization (1.3) under a RIP condition of the sensing matrix $A$ in Section 2. We then prove the full rank property of the sensing matrix $A$ restricted to the support of a local minimizer for the $\ell_{1-2}$ minimization
problem. As a corollary, we infer that the number of local minimizers of either (1.3) or (1.4) is finite. In Section 3, we show an iterative computational method for (1.4) based on the difference of convex functions algorithm (DCA), and establish the convergence of the method to a stationary point where the first-order optimality condition holds. In Section 4, we further analyze a sparsity oriented simulated annealing algorithm for approaching a global minimizer almost surely. In Section 5, we compare our DCA-$\ell_{1-2}$ method with various CS solvers numerically. For ill-conditioned matrices $A$, such as an oversampled DCT matrix, DCA-$\ell_{1-2}$ is the best, as seen from the success rate vs. sparsity plot. In this regime of $A$, exact recovery is still possible provided that the peaks of solution are sufficiently separated. We also evaluate the three metrics on a two-dimensional example of reconstructing magnetic resonance imaging (MRI) from a limited number of projections. In this application, we minimize the metrics on the image gradient, where the image is a Shepp-Logan phantom of dimensions $256 \times 256$. Using $\ell_{1-2}$, we observed that 8 projections suffice for exact recovery, while IRLS for $\ell_{1/2}$ minimization takes 10. Still at 8 projections, the relative recovery error is a factor of $2 \times 10^6$ larger under the split Bregman for $\ell_1$. The concluding remarks are in Section 6.

Notations. Let us fix some notations. For any $x, y \in \mathbb{R}^n$, $\langle x, y \rangle = x^T y$ is their inner product. $\text{supp}(x) := \{1 \leq i \leq n : x_i \neq 0\}$ denotes the support of $x$, and $\|x\|_0 := |\text{supp}(x)|$ is cardinality of $\text{supp}(x)$. $B_r(x) = \{y \in \mathbb{R}^n : \|y - x\|_2 < r\}$ denotes the $n$-dimensional Euclidean ball centered at $x$ with radius $r > 0$. Let $T \subseteq \{1, \ldots, n\}$ be an index set, and let $|T|$ be the cardinality of $T$. Moreover, for $A \in \mathbb{R}^{m \times n}$, $A_T \in \mathbb{R}^{m \times |T|}$ is the submatrix of $A$ with column indices $T$. $I_m$ is the identity matrix of dimension $m$. The $\text{sgn}(x)$ is the signum function defined as

$$\text{sgn}(x) := \begin{cases} 
1 & \text{if } x > 0, \\
-1 & \text{if } x < 0, \\
0 & \text{if } x = 0.
\end{cases}$$

2. Theory of $\ell_{1-2}$ Minimization.

2.1. Preliminaries. Restricted isometry property (RIP) introduced by Candès et al. [8] is one of the most used frameworks for CS, which characterizes matrices that are nearly orthonormal.

**Definition 2.1.** For $T \subseteq \{1, \ldots, n\}$ and each number $s$, $s$-restricted isometry constant of $A$ is the smallest $\delta_s \in (0, 1)$ such that

$$(1 - \delta_s)\|x\|_2^2 \leq \|A_T x\|_2^2 \leq (1 + \delta_s)\|x\|_2^2$$

Fig. 1: Contours of three sparsity metrics. The level curves of $\ell_1 - \ell_2$ approach the x and y axes as the values get small, hence promoting sparsity.
for all subsets $T$ with $|T| \leq s$ and all $x \in \mathbb{R}^{|T|}$. The matrix $A$ is said to satisfy the $s$-RIP with $\delta_s$.

Sensing matrices with small $\delta_s$ are suitable for reconstruction of sparse signals [8, 9]. It has been shown that with overwhelming probability, random Gaussian, random Bernoulli, and random partial Fourier matrices satisfy the RIP (with small restricted isometry constants) [8, 18, 44]. Given a deterministic matrix $A$, it is generally NP-hard however, to verify whether $A$ is a RIP matrix or not [1]. Another commonly used CS concept is the so-called mutual coherence or coherence [21] for short.

**Definition 2.2.** The coherence of a matrix $A$ is the maximum absolute value of the cross-correlations between the columns of $A$, namely

$$
\mu(A) := \max_{i \neq j} \frac{|A_i^T A_j|}{\|A_i\|_2 \|A_j\|_2}.
$$

Coherence is closely related to the RIP yet easy to examine. Specifically, a matrix satisfying some RIP tends to have small coherence or to be incoherent. Conversely, a highly coherent matrix is unlikely to possess small restricted isometry constants.

### 2.2. Exact and stable recovery.

We have the following fundamental properties of the function $\|x\|_1 - \|x\|_2$, which will be frequently invoked later in the proofs.

**Lemma 2.1.** Suppose $x \in \mathbb{R}^n \setminus \{0\}$, $\Lambda = \text{supp}(x)$ and $\|x\|_0 = s$, then

(a) $(n - \sqrt{n}) \min_i |x_i| \leq \|x\|_1 - \|x\|_2 \leq (\sqrt{n} - 1)\|x\|_2$.

(b) $(s - \sqrt{s}) \min_{i \in \Lambda} |x_i| \leq \|x\|_1 - \|x\|_2 \leq (\sqrt{s} - 1)\|x\|_2$.

(c) $\|x\|_1 - \|x\|_2 = 0$ if and only if $s = 1$.

The proof is given in the appendix.

A RIP based sufficient condition was derived in [9] for exact recovery of BP (1.1), here we derive an analogous condition for that of $\ell_{1-2}$ minimization, demonstrating the capability of $\ell_{1-2}$ to promote sparsity.

**Theorem 2.1.** Let $\bar{x}$ be any vector with sparsity of $s$ satisfying

$$
a(s) = \left(\frac{\sqrt{3s} - 1}{\sqrt{s} + 1}\right)^2 > 1,
$$

and let $b = A\bar{x}$, suppose $A$ satisfies the following condition

$$
\delta_{3s} + a(s)\delta_{4s} < a(s) - 1,
$$

then $\bar{x}$ is the unique solution to (1.3).

**Proof.** The proof generally follows the lines of [9]. Let $x$ be any feasible solution satisfying the constraint $Ax = b$ yet with a smaller objective value, i.e.

$$
\|x\|_1 - \|x\|_2 \leq \|\bar{x}\|_1 - \|\bar{x}\|_2.
$$

We write $x = \bar{x} + v$ with $v \in \ker(A)$, and want to show $v = 0$.

Let $\Lambda = \text{supp}(\bar{x})$, we further decompose $v$ as $v = v_\Lambda + v_{\Lambda^c}$. Then (2.2) becomes

$$
\|\bar{x} + v_\Lambda + v_{\Lambda^c}\|_1 - \|\bar{x} + v_\Lambda + v_{\Lambda^c}\|_2 \leq \|\bar{x}\|_1 - \|\bar{x}\|_2.
$$

On the other hand,

$$
\|\bar{x} + v_\Lambda + v_{\Lambda^c}\|_1 - \|\bar{x} + v_\Lambda + v_{\Lambda^c}\|_2 = \|\bar{x} + v_\Lambda\|_1 + \|v_{\Lambda^c}\|_1 - \|\bar{x} + v_\Lambda + v_{\Lambda^c}\|_2 \\
\geq \|\bar{x}\|_1 - \|v_\Lambda\|_1 - \|v_{\Lambda^c}\|_1 - \|\bar{x}\|_2 - \|v_\Lambda\|_2 - \|v_{\Lambda^c}\|_2.
$$
So $v$ must obey the following inequality constraint

\begin{equation}
\|v_\Lambda\|_1 + \|v_\Lambda\|_2 \geq \|v_{\Lambda^c}\|_1 - \|v_{\Lambda^c}\|_2. \tag{2.3}
\end{equation}

Arrange the indices in $\Lambda^c$ in order of decreasing magnitude of $v_{\Lambda^c}$ and divide $\Lambda^c$ into subsets of size $3s$. Then $\Lambda^c = \Lambda_1 \cup \Lambda_2 \cup \cdots \cup \Lambda_t$, where each $\Lambda_i$ contains $m$ indices probably except $\Lambda_t$. Denoting $\Lambda_0 = \Lambda \cup \Lambda_1$ and using the RIP of $A$, we have

\begin{align*}
0 = \|Av\|_2 &= \|A_{\Lambda_0}v_{\Lambda_0} + \sum_{i=2}^t A_{\Lambda_i}v_{\Lambda_i}\|_2 \geq \|A_{\Lambda_0}v_{\Lambda_0}\|_2 - \|\sum_{i=2}^t A_{\Lambda_i}v_{\Lambda_i}\|_2 \\
&\geq \|A_{\Lambda_0}v_{\Lambda_0}\|_2 - \sum_{i=2}^t \|A_{\Lambda_i}v_{\Lambda_i}\|_2 \\
&\geq \sqrt{1 - \delta_{4s}}\|v_{\Lambda_0}\|_2 - \sqrt{1 + \delta_{3s}}\sum_{i=2}^t \|v_{\Lambda_i}\|_2 \tag{2.4}
\end{align*}

Now we set an upper bound on $\sum_{i=2}^t \|v_{\Lambda_i}\|_2$. For each $t \in \Lambda_i$, $i \geq 2$,

\[|v_t| \leq \min_{r \in \Lambda_{i-1}} |v_r| = \frac{\|v_{\Lambda_{i-1}}\|_1 - \|v_{\Lambda_{i-1}}\|_2}{3s - \sqrt{3}s},\]

where the second inequality follows from Lemma 2.1(a). Then it follows that

\[\|v_{\Lambda_i}\|_2 \leq \sqrt{3s} \frac{\|v_{\Lambda_{i-1}}\|_1 - \|v_{\Lambda_{i-1}}\|_2}{3s - \sqrt{3}s} = \frac{\|v_{\Lambda_{i-1}}\|_1 - \|v_{\Lambda_{i-1}}\|_2}{\sqrt{3s} - 1},\]

and

\begin{equation}
\sum_{i=2}^t \|v_{\Lambda_i}\|_2 \leq \frac{\sum_{i=1}^{t-1} \|v_{\Lambda_i}\|_1 - \|v_{\Lambda_i}\|_2}{\sqrt{3s} - 1} \leq \frac{\sum_{i=1}^t \|v_{\Lambda_i}\|_1 - \sum_{i=1}^t \|v_{\Lambda_i}\|_2}{\sqrt{3s} - 1}. \tag{2.5}
\end{equation}

Note that in (2.5)

\[\sum_{i=1}^t \|v_{\Lambda_i}\|_1 = \|v_{\Lambda^c}\|_1 \quad \text{and} \quad \sum_{i=1}^t \|v_{\Lambda_i}\|_2 \geq \sqrt{\sum_{i=1}^t \|v_{\Lambda_i}\|_2^2} = \|v_{\Lambda^c}\|_2.\]

Combining (2.5) and (2.3) gives

\[\sum_{i=2}^t \|v_{\Lambda_i}\|_2 \leq \frac{\|v_{\Lambda^c}\|_1 - \|v_{\Lambda^c}\|_2}{\sqrt{3s} - 1} \leq \frac{\|v_{\Lambda^c}\|_1 + \|v_{\Lambda^c}\|_2}{\sqrt{3s} - 1} \leq \frac{(\sqrt{s} + 1)\|v_{\Lambda}\|_2}{\sqrt{a(s)}} = \frac{\|v_{\Lambda}\|_2}{\sqrt{a(s)}}.\]

So it follows from (2.4) that

\[0 \geq \sqrt{1 - \delta_{4s}}\|v_{\Lambda_0}\|_2 - \frac{\sqrt{1 + \delta_{3s}}}{\sqrt{a(s)}}\|v_{\Lambda}\|_2 \geq \sqrt{1 - \delta_{4s}}\|v_{\Lambda_0}\|_2 - \frac{\sqrt{1 + \delta_{3s}}}{\sqrt{a(s)}}\|v_{\Lambda_0}\|_2.\]

Since (2.1) amounts to

\[\sqrt{1 - \delta_{4s}} - \frac{\sqrt{1 + \delta_{3s}}}{\sqrt{a(s)}} > 0,\]

we have $v_{\Lambda_0} = 0$. This implies $v = 0$, which completes the proof. \qed
Remark 2.1. (2.1) can be rewritten as
\[ \delta_{3s} < a(s)(1 - \delta_{4s}) - 1. \]

Note that the RIP condition for exact recovery of BP derived in [9] reads
\[ \delta_{3s} + 3\delta_{4s} < 2, \]
or equivalently
\[ \delta_{3s} < 3(1 - \delta_{4s}) - 1. \]

The condition (2.1) required for \( \ell_{1-2} \) exact recovery appears more stringent than (2.6) for \( \ell_1 \) recovery since \( a(s) < 3 \) (and thus also stronger than the RIP for \( \ell_p \) recovery with \( 0 < p < 1 \) [12]).

However, this does not mean \( \ell_1 \) norm is superior to \( \ell_{1-2} \) in terms of sparsity promoting. On the contrary, in Section 5 it will be shown numerically that \( \ell_{1-2} \) penalty consistently outperforms \( \ell_1 \). Besides possible technical issues lying in the proof (e.g. the estimate in (2.3) is in fact not sharp), another explanation can be that a RIP-based condition is just a sufficient condition to guarantee that a measurement matrix \( A \) fits for exact reconstruction. It happens that two matrices have exactly the same performance and yet one satisfies RIP whereas the other does not [60].

Remark 2.2. The assumptions of Theorem 2.1 require \( a(s) = \left( \frac{\sqrt{3s} - 1}{\sqrt{s} + 1} \right)^2 > 1 \), which implies \( s \geq 8 \). Then a natural question is whether the uniqueness of \( \hat{x} \) still holds for the case \( 1 \leq s \leq 7 \).

First of all, when \( s = 1 \), any minimizer of \( \|x\|_1 - \|x\|_2 \) other than \( x \) must be 1-sparse (and be a feasible solution of \( Ax = b \)). So the RIP condition to guarantee uniqueness is just \( \delta_2 < 1 \). When \( s \geq 2 \), we redefine \( a(s) \) as \( \left( \frac{\sqrt{6s} - 1}{\sqrt{s} + 1} \right)^2 \). It is easy to check that \( a(s) > 1 \) for \( s \geq 2 \). By similar argument, we can show that the following RIP condition suffices for the uniqueness of \( \hat{x} \):
\[ \delta_{6s} + a(s)\delta_{7s} < a(s) - 1. \]

Similar to [7], we also establish the following stable recovery of \( \ell_{1-2} \) when measurements are contaminated by noises.

Theorem 2.2. Under the assumptions of Theorem 2.1 except that \( b = Ax + e \), where \( e \in \mathbb{R}^m \) is any perturbation with \( \|e\|_2 \leq \tau \), we have that the solution \( x^{\text{opt}} \) to the following variant of problem (1.3)
\[ \min_{x \in \mathbb{R}^n} \|x\|_1 - \|x\|_2 \quad \text{subject to} \quad \|Ax - b\|_2 \leq \tau \]
obeys \( \|x^{\text{opt}} - \hat{x}\|_2 \leq C_s \tau \) for some constant \( C_s > 0 \) depending on \( \delta_{3s} \) and \( \delta_{4s} \).

Proof. Let \( \Lambda \) be the support of \( \hat{x} \) and \( x^{\text{opt}} = \hat{x} + v \). Then starting from \( \|x^{\text{opt}}\|_1 - \|x^{\text{opt}}\|_2 \leq \|\hat{x}\|_1 - \|\hat{x}\|_2 \) and repeating the arguments in the proof of Theorem 2.1, we obtain
\[ \sum_{i=2}^l \|v_{\Lambda_i}\|_2 \leq \frac{\|v_{\Lambda_0}\|_2}{\sqrt{a(s)}} \]
and
\[ \|Av\|_2 \geq \left( \sqrt{1 - \delta_{4s}} - \frac{\sqrt{1 + \delta_{4s}}}{\sqrt{a(s)}} \right) \|v_{\Lambda_0}\|_2. \]
From (2.7) it follows that
\[ \|v\|_2 = \sqrt{\|v_{A_0}\|_2^2 + \sum_{i=2}^{t} \|v_{A_i}\|_2^2} \leq \sqrt{\|v_{A_0}\|_2^2 + \frac{\|v_{A_2}\|_2^2}{a(s)}} \leq \sqrt{1 + \frac{1}{a(s)}} \|v_{A_0}\|_2, \]
so (2.8) becomes
\[ (2.9) \quad \|Av\|_2 \geq \frac{\sqrt{\alpha(s)(1 - \delta_{4s})} - \sqrt{1 + \delta_{3s}}}{\sqrt{1 + a(s)}} \|v\|_2. \]
On the other hand, since \( \|A\bar{x} - b\|_2 \leq \tau \) and \( \|Ax^{\text{opt}} - b\|_2 \leq \tau \), by the triangular inequality,
\[ (2.10) \quad \|Av\|_2 = \|(Ax^{\text{opt}} - b) - (A\bar{x} - b)\|_2 \leq \|A\bar{x} - b\|_2 + \|Ax^{\text{opt}} - b\|_2 \leq 2\tau. \]
Combining (2.9) and (2.10), we have \( \|v\|_2 \leq C_s \tau \), where \( C_s := \frac{2\sqrt{1 + a(s)}}{\sqrt{\alpha(s)(1 - \delta_{4s})} - \sqrt{1 + \delta_{3s}}} > 0. \)

**Remark 2.3.** The upper bound \( C_s \tau \) of the approximation error basically relies on how well the RIP condition (2.1) is satisfied. \( C_s \) is \( O(1) \) if \( \delta_{3s} \) and \( \delta_{4s} \) are small and \( a(s) \gg 1 \).

### 2.3. Sparsity of local minimizers

Next we shall prove that local minimizers of the problems (1.3) and (1.4) possess certain sparsity in the sense that they only extract linearly independent columns from the sensing matrix \( A \), whether \( A \) satisfies any RIP or not. In other words, minimizing \( \ell_{1-2} \) will rule out redundant columns of \( A \). It is worth noting that similar results was proved in [9] for the \( \ell_p \) unconstrained problem using the second-order optimality condition. This demonstrates an advantage of non-convex sparsity metrics over the convex \( \ell_1 \) norm.

**Theorem 2.3.** Let \( x^* \) be a local minimizer of the constrained problem (1.3) and \( \Lambda^* = \text{supp}(x^*) \), then \( A_{\Lambda^*} \) is of full column rank, i.e. the columns of \( A_{\Lambda^*} \) are linearly independent.

**Proof.** The proof simply uses the definition of local minimizer. Suppose the columns of \( A_{\Lambda^*} \) are linearly dependent, then there exists \( v \in \ker(A) \setminus \{0\} \) such that \( \text{supp}(v) \subseteq \Lambda^* \). For any fixed neighborhood \( B_r(x^*) \) of \( x^* \), we scale \( v \) so that
\[ \|v\|_2 < \min \left\{ \min_{\bar{x} \in \Lambda^*} \|\bar{x}^*\|, r \right\}. \]
Consider two feasible vectors in \( B_r(x^*) \), \( \hat{x} = x^* + v \) and \( \bar{x} = x^* - v \). Since \( \text{supp}(v) \subseteq \Lambda^* \), we have \( \text{supp}(\hat{x}) \subseteq \Lambda^* \) and \( \text{supp}(\bar{x}) \subseteq \Lambda^* \). Moreover,
\[ (x^* \pm v_i) = x^*_i \pm v_i = \text{sgn}(x^*_i)(|x^*_i| \pm \text{sgn}(x^*_i)v_i), \forall i \in \Lambda^*. \]
The above implies \( \text{sgn}(\hat{x}_i) = \text{sgn}(\bar{x}_i) = \text{sgn}(x^*_i) \) for all \( i \in \Lambda^* \), since
\[ |x^*_i| \pm \text{sgn}(x^*_i)v_i \geq |x^*_i| - |v_i| \geq \min_{i \in \Lambda^*} |x^*_i| - \|v\|_2 > 0, \forall i \in \Lambda^*. \]
In other words, \( x^* \), \( \hat{x} \) and \( \bar{x} \) are located in the same orthant. It follows that
\[ (2.11) \quad \|x^*\|_1 = \frac{1}{2}\|\hat{x} + \bar{x}\|_1 = \frac{1}{2}\|\hat{x}\|_1 + \frac{1}{2}\|\bar{x}\|_1, \]
and
\[ (2.12) \quad \|x^*\|_2 = \frac{1}{2}\|\hat{x} + \bar{x}\|_2 < \frac{1}{2}\|\hat{x}\|_2 + \frac{1}{2}\|\bar{x}\|_2. \]
Suppose (1.4) and both the numbers of local minimizers of (1.4) are finite. Let $x^*$ be a local minimizer of the unconstrained problem (1.4) then the columns of $A^*$ are linearly independent.

**Proof.** We claim that $x^*$ is also a local minimizer of the following constrained problem

$$
\min_{x \in \mathbb{R}^n} \|x\|_1 - \|x\|_2 \quad \text{subject to} \quad Ax = Ax^*.
$$

Suppose not, then \( \forall r > 0 \), there exists \( \bar{x} \in B_r(x^*) \) such that \( A\bar{x} = Ax^* \) and

$$
\|\bar{x}\|_1 - \|\bar{x}\|_2 < \|x^*\|_1 - \|x^*\|_2.
$$

This implies

$$
\frac{1}{2}\|A\bar{x} - b\|_2^2 + \lambda(\|\bar{x}\|_1 - \|\bar{x}\|_2) < \frac{1}{2}\|Ax^* - b\|_2^2 + \lambda(\|x^*\|_1 - \|x^*\|_2).
$$

Thus for any $r > 0$, we actually find a $\bar{x} \in B_r(x^*)$ yielding a smaller objective of (1.4) than $x^*$, which leads to a contradiction because $x^*$ is assumed to be a local minimizer. Thus the claim is validated.

Using the claim above and Theorem 2.3, we have that the columns of $A^*$ are linearly independent.

By Theorem 2.3 and 2.4, we readily conclude the following facts:

**Corollary 2.1.**

(a) Suppose $x^*$ is a local minimizer of (1.3) or (1.4), since \( \text{rank}(A) = m \), the sparsity of $x^*$ is at most $m$.

(b) If $x^*$ is a local minimizer of (1.3), then there is no such $x \in \mathbb{R}^n$ satisfying $Ax = b$ and \( \text{supp}(x) \subseteq A^* \), i.e. it is impossible to find a feasible solution whose support is contained in \( \text{supp}(x^*) \).

(c) Both the numbers of local minimizers of (1.3) and (1.4) are finite.

3. **Computational Approach.** In this section, we consider the minimization of the unconstrained $\ell_{1-2}$ problem (1.4).

3.1. **Difference of convex functions algorithm.** The DCA is a descent method without line search introduced by Tao and An [45, 46]. It copes with the minimization of an objective function $F(x) = G(x) - H(x)$ on the space $\mathbb{R}^n$, where $G(x)$ and $H(x)$ are lower semicontinuous proper convex functions. Then $G - H$ is called a DC decomposition of $F$, whereas $G$ and $H$ are DC components of $F$.

The DCA involves the construction of two sequences \( \{x^k\} \) and \( \{y^k\} \), the candidates for optimal solutions of primal and dual programs respectively. To implement the DCA, one iteratively computes

$$
\begin{align*}
    y^k & \in \partial H(x^k) \\
    x^{k+1} & = \arg\min_{x \in \mathbb{R}^n} G(x) - (H(x^k) + \langle y^k, x - x^k \rangle)
\end{align*}
$$
where \( y^k \in \partial H(x^k) \) means that \( y^k \) is a subgradient of \( H(x) \) at \( x^k \). By the definition of subgradient, we have
\[
H(x) \geq H(x^k) + \langle y^k, x - x^k \rangle, \quad \forall x \in \mathbb{R}^n.
\]

In particular, \( H(x^{k+1}) \geq H(x^k) + \langle y^k, x^{k+1} - x^k \rangle \), consequently
\[
F(x^k) = G(x^k) - H(x^k) \geq G(x^{k+1}) - (H(x^k) + \langle y^k, x^{k+1} - x^k \rangle) \geq G(x^{k+1}) - H(x^{k+1}) = F(x^{k+1}).
\]

The fact that \( x^{k+1} \) minimizes \( G(x) - (H(x^k) + \langle y^k, x - x^k \rangle) \) was used in the first inequality above. Therefore, the DCA iteration (3.1) yields a monotonically decreasing sequence \( \{F(x^k)\} \) of objective values, resulting in its convergence provided \( F(x) \) is bounded from below.

The objective in (1.4) naturally has the following DC decomposition
\[
(3.1) \quad F(x) = \left( \frac{1}{2} \|Ax - b\|_2^2 + \lambda \|x\|_1 \right) - \lambda \|x\|_2.
\]

Note that \( \|x\|_2 \) is differentiable with gradient \( \frac{x}{\|x\|_2} \) for all \( x \neq 0 \) and that \( \mathbf{0} \in \partial \|x\|_2 \) for \( x = \mathbf{0} \), thus the strategy to iterate is as follows:

\[
(3.2) \quad x^{k+1} = \begin{cases} \arg \min_{x \in \mathbb{R}^n} \frac{1}{2} \|Ax - b\|_2^2 + \lambda \|x\|_1 & \text{if } x^k = \mathbf{0}, \\ \arg \min_{x \in \mathbb{R}^n} \frac{1}{2} \|Ax - b\|_2^2 + \lambda \|x\|_1 - \langle x, \lambda \frac{x^k}{\|x^k\|_2} \rangle & \text{otherwise.} \end{cases}
\]

It will be shown in Proposition 3.1 that \( \|x^{k+1} - x^k\|_2 \to 0 \) as \( k \to \infty \), so a reasonable termination criterion for (3.2) can be
\[
(3.3) \quad \frac{\|x^{k+1} - x^k\|_2}{\max\{\|x^k\|_2, 1\}} < \epsilon
\]
for some given parameter \( \epsilon > 0 \).

The DCA in general does not guarantee a global minimum due to non-convex nature of the problem [45]. One could in principle prove convergence to the global minimum by the branch and bound procedure (as done in [39]), but the cost is often too high. A good initial guess is therefore crucial for the performance of algorithm. The experiments in Section 5 will show that the DCA often produces a solution that is close to global minimizer when starting with \( x^0 = \mathbf{0} \). The intuition behind our choice can be that the first step of (3.2) reduces to solving the unconstrained \( \ell_1 \) problem. So basically we are minimizing \( \ell_{1-2} \) on top of \( \ell_1 \), which possibly explains why we observed in the experiments that \( \ell_{1-2} \) regularization initialized by \( x^0 = \mathbf{0} \) always outperforms \( \ell_1 \) regularization. Hereby we summarize DCA-\( \ell_{1-2} \) in Algorithm 1 below.

### Algorithm 1 DCA-\( \ell_{1-2} \) for solving (1.4).

Define \( \epsilon > 0 \) and set \( x^0 = \mathbf{0} \), \( n = 0 \).

for \( k = 0, 1, 2, \ldots \), Maxoit do

if \( x^k = \mathbf{0} \) then

\( x^{k+1} = \arg \min_{x \in \mathbb{R}^n} \frac{1}{2} \|Ax - b\|_2^2 + \lambda \|x\|_1 \)

else

\( x^{k+1} = \arg \min_{x \in \mathbb{R}^n} \frac{1}{2} \|Ax - b\|_2^2 + \lambda \|x\|_1 - \langle x, \lambda \frac{x^k}{\|x^k\|_2} \rangle \)

end if

end for
3.2. Convergence analysis. Assuming each DCA iteration of (3.2) is solved accurately, we show that the sequence \( \{x^k\} \) is bounded and \( \|x^{k+1} - x^k\|_2 \to 0 \), and limit points of \( \{x^k\} \) are stationary points of (1.4) satisfying the first-order optimality condition. Note that \( \text{ker}(A^T A) \) is non-trivial, so both the DC components in (3.1) only have weak convexity. As a result, the convergence of (3.2) is not covered by the standard convergence analysis for the DCA (e.g. Theorem 3.7 of [46]), because strong convexity is otherwise needed.

**Lemma 3.1.** For all \( \lambda > 0 \), \( F(x) = \frac{1}{2}\|Ax - b\|_2^2 + \lambda(\|x\|_1 - \|x\|_2) \to \infty \) as \( \|x\|_2 \to \infty \), therefore \( F(x) \) is coercive in the sense that the level set \( \{x \in \mathbb{R}^n : F(x) \leq F(x^0)\} \) is bounded, \( \forall x^0 \in \mathbb{R}^n \).

**Proof.** It suffices to show that for any fixed \( x \in \mathbb{R}^n \setminus \{0\}, F(cx) \to \infty \) as \( c \to \infty \).

\[
F(cx) = \frac{1}{2}\|cAx - b\|_2^2 + c\lambda(\|x\|_1 - \|x\|_2) \\
\geq \frac{1}{2}(c\|Ax\|_2 - \|b\|_2)^2 + c\lambda(\|x\|_1 - \|x\|_2)
\]

If \( Ax = 0 \), i.e. \( x \in \text{ker}(A) \setminus \{0\} \), since \( \text{rank}(A) = m \), we have \( \|x\|_0 \geq m + 1 \geq 2 \). Lemma 2.1(c) implies \( \|x\|_1 - \|x\|_2 > 0 \), so \( F(cx) = \frac{1}{2}\|b\|_2^2 + c\lambda(\|x\|_1 - \|x\|_2) \to \infty \) as \( c \to \infty \).

If \( Ax \neq 0 \), the claim follows as we notice that \( c\|Ax\|_2 - \|b\|_2 \to \infty \) as \( c \to \infty \). \( \square \)

**Lemma 3.2.** Let \( \{x^k\} \) be the sequence generated by the DCA (3.2). For all \( k \in \mathbb{N} \), we have

\[
F(x^k) - F(x^{k+1}) \geq \frac{1}{2}\|A(x^k - x^{k+1})\|_2^2 + \lambda(\|x^{k+1}\|_2 - \|x^k\|_2 - \langle y^k, x^{k+1} - x^k \rangle) \geq 0,
\]

where \( y^k \in \partial\|x^k\|_2 \).

**Proof.** A simple calculation shows

\[
F(x^k) - F(x^{k+1}) = \frac{1}{2}\|A(x^{k+1} - x^k)\|_2^2 + \langle A(x^k - x^{k+1}), Ax^{k+1} - b \rangle \\
+ \lambda(\|x^{k+1}\|_1 - \|x^{k+1}\|_1 - \|x^k\|_2 + \|x^{k+1}\|_2)
\]

Recall that \( x^{k+1} \) is the solution to the problem

\[
\min_{x \in \mathbb{R}^n} \frac{1}{2}\|Ax - b\|_2^2 + \lambda(\|x\|_1 - \langle x, y^k \rangle)
\]

with \( y^k \in \partial\|x^k\|_2 \), then the first-order optimality condition holds at \( x^{k+1} \). More precisely, there exists \( w^{k+1} \in \partial\|x^{k+1}\|_1 \), such that

\[
A^T(Ax^{k+1} - b) + \lambda(w^{k+1} - y^k) = 0.
\]

Left multiplying (3.6) by \( (x^k - x^{k+1})^T \) gives

\[
\langle A(x^k - x^{k+1}), Ax^{k+1} - b \rangle + \lambda(\langle w^{k+1}, x^k \rangle - \|x^{k+1}\|_1) + \langle y^k, x^{k+1} - x^k \rangle = 0,
\]

where we used \( \langle w^{k+1}, x^{k+1} \rangle = \|x^{k+1}\|_1 \). Combining (3.5) and (3.7), we have

\[
F(x^k) - F(x^{k+1}) = \frac{1}{2}\|A(x^{k+1} - x^k)\|_2^2 + \lambda(\|x^{k+1}\|_1 - \langle w^{k+1}, x^k \rangle) \\
+ \lambda(\|x^{k+1}\|_2 - \|x^k\|_2 - \langle y^k, x^{k+1} - x^k \rangle) \\
\geq \frac{1}{2}\|A(x^{k+1} - x^k)\|_2^2 + \lambda(\|x^{k+1}\|_2 - \|x^k\|_2 - \langle y^k, x^{k+1} - x^k \rangle) \\
\geq 0.
\]
In the first inequality above, \(\|x^k\|_1 - \langle w^{k+1}, x^k \rangle \geq 0\) since \(|w_i^{k+1}| \leq 1\) for all \(1 \leq i \leq n\), while the second one holds because \(y^k \in \partial \|x^k\|_2\).

We now show the convergence results for Algorithm 1.

**Proposition 3.1.** Let \(\{x^k\}\) be the sequence of iterates generated by Algorithm 1, we have

(a) \(\{x^k\}\) is bounded.

(b) \(\|x^{k+1} - x^k\|_2 \to 0\) as \(k \to \infty\).

(c) Any nonzero limit point \(x^*\) of \(\{x^k\}\) satisfies the first-order optimality condition

\[
0 \in A^T(Ax^* - b) + \lambda(\partial \|x^*\|_1 - \frac{x^*}{\|x^*\|_2}),(3.8)
\]

which means \(x^*\) is a stationary point of (1.4).

**Proof.** (a) Using Lemma 3.1 and the fact that \(\{F(x^k)\}\) is monotonically decreasing, we have \(\{x^k\} \subseteq \{x \in \mathbb{R}^n : F(x) \leq F(x^0)\}\) is bounded.

(b) If \(x^1 = x^0 = 0\), we then stop the algorithm producing the solution \(x^* = 0\). Otherwise, it follows from (3.4) that

\[
F(0) - F(x^1) \geq \lambda \|x^1\|_2 > 0,
\]

so \(x^k \neq 0\) whenever \(k \geq 1\). In what follows, we assume \(x^k \neq 0\) for all \(k \geq 1\). Since \(\{F(x^k)\}\) is convergent, substituting \(y^k = \frac{x^k}{\|x^k\|_2}\) in (3.4), we must have

\[
\|A(x^k - x^{k+1})\|_2 \to 0
\]

(3.9)

\[
\|x^{k+1}\|_2 - \frac{\langle x^k, x^{k+1} \rangle}{\|x^k\|_2} \to 0
\]

(3.10)

We define \(c^k := \frac{\langle x^k, x^{k+1} \rangle}{\|x^k\|_2^2}\) and \(c^n := x^{k+1} - c^k x^k\), then it suffices to prove \(c^k \to 0\) and \(c^n \to 1\).

It is straightforward to check that

\[
\|c^n\|_2 = \|x^{k+1}\|_2^2 - \frac{\langle x^k, x^{k+1} \rangle^2}{\|x^k\|_2^2} \to 0,
\]

where we used (3.10). Then from (3.9) it follows that

\[
0 = \lim_{k \to \infty} \|A(x^k - x^{k+1})\|_2 = \lim_{k \to \infty} \|A((c^k - 1)x^k - e^k)\|_2 = \lim_{k \to \infty} |c^k - 1| \|Ax^k\|_2.
\]

If \(\lim_{k \to \infty} c^k - 1 \neq 0\), then there exists a subsequence \(\{x^{k_j}\}\) such that \(Ax^{k_j} \to 0\). So we have

\[
\lim_{k_j \to \infty} F(x^{k_j}) \geq \lim_{k_j \to \infty} \frac{1}{2} \|Ax^{k_j} - b\|_2^2 = \frac{1}{2} \|b\|_2^2 = F(x^0),
\]

which is contradictory to the fact that

\[
F(x^{k_j}) \leq F(x^1) < F(x^0), \quad \forall k_j \geq 1.
\]

Therefore \(c^k \to 1\), \(c^n \to 0\), and thus \(x^{k+1} - x^k \to 0\) as \(k \to \infty\).

(c) Let \(\{x^{k_j}\}\) be a subsequence of \(\{x^k\}\) converging to \(x^* \neq 0\), so the optimality condition at the \(k_j\)-th step of Algorithm 1 reads

\[
0 \in A^T(Ax^{k_j} - b) + \lambda(\partial \|x^{k_j}\|_1 - \frac{x^{k_j}}{\|x^{k_j}\|_2^2}).
\]
or
\[(3.11) \quad -(A^T(Ax_{k_j} - b) - \lambda \frac{x_{k_j} - 1}{\|x_{k_j} - 1\|_2}) \in \lambda \partial \|x^*\|_1.\]

Here \(\partial \|x\|_1 = \prod_{i=1}^n \text{SGN}(x_i) \subset \mathbb{R}^n\) with
\[
\text{SGN}(x_i) := \begin{cases} 
\{\text{sgn}(x_i)\} & \text{if } x_i \neq 0, \\
[-1, 1] & \text{otherwise.}
\end{cases}
\]

Since, by (b), \(x_{k_j} \rightarrow x^*\), we have, when \(k_j\) is sufficiently large, \(\text{supp}(x^*) \subseteq \text{supp}(x_{k_j})\) and \(\text{sgn}(x_{k_j}^i) = \text{sgn}(x^*_i)\) for all \(i \in \text{supp}(x^*)\), which implies
\[
\partial \|x_{k_j}\|_1 \subseteq \partial \|x^*\|_1.
\]

Then by (3.11), for large \(k_j\) we have
\[(3.12) \quad -(A^T(Ax_{k_j} - b) - \lambda \frac{x_{k_j} - 1}{\|x_{k_j} - 1\|_2}) \in \lambda \partial \|x^*\|_1.
\]

Moreover, since \(x^*\) is away from \(0\),
\[
\lim_{k_j \rightarrow \infty} A^T(Ax_{k_j} - b) - \lambda \frac{x_{k_j} - 1}{\|x_{k_j} - 1\|_2} = \lim_{k_j \rightarrow \infty} A^T(Ax_{k_j} - b) - \lambda \frac{x_{k_j} - 1}{\|x_{k_j} - 1\|_2} + \lambda \frac{x_{k_j} - 1}{\|x_{k_j} - 1\|_2} - \lambda \frac{x_{k_j} - 1}{\|x_{k_j} - 1\|_2}.
\]

Let \(k_j \rightarrow \infty\) in (3.12) and note that \(\partial \|x^*\|_1\) is a closed set, then (3.8) follows.

By choosing appropriate regularization parameters, we are able to control the sparsity of \(x^*\).

**Theorem 3.1.** \(\forall s \in \mathbb{N},\) there exists \(\lambda_s > 0\) such that for any parameter \(\lambda > \lambda_s\) in (1.4), we have \(\|x^*\|_0 \leq s\), where \(x^* \neq 0\) is a stationary point generated by Algorithm 1.

**Proof.** By the optimality condition (3.8), there exists \(w^* \in \partial \|x^*\|_1\) satisfying
\[(3.13) \quad w_{k_j}^* = \begin{cases} 
\text{sgn}(x_{k_j}^i) & \text{if } i \in \text{supp}(x^*), \\
[-1, 1] & \text{otherwise.}
\end{cases}
\]

such that
\[-A^T(Ax^* - b) = \lambda(w^* - \frac{x^*}{\|x^*\|_2}).\]

Since, by (3.13) \(\|w^*\|_2 \geq \sqrt{\|x^*\|_0}\), taking \(\ell_2\) norm of both sides gives
\[(3.14) \quad \|A^T(Ax^* - b)\|_2 = \lambda \|w^* - \frac{x^*}{\|x^*\|_2}\|_2 \geq \lambda(\|w^*\|_2 - \|\frac{x^*}{\|x^*\|_2}\|_2) \geq \lambda(\sqrt{\|x^*\|_0} - 1).\]

On the other hand,
\[(3.15) \quad \|A^T(Ax^* - b)\|_2 \leq \|A\|_2 \|Ax^* - b\|_2 = \|A\|_2 \|Ax^* - b\|_2 \leq \|A\|_2 \|b\|_2,\]
where we used
\[ \frac{1}{2} \|Ax^* - b\|_2^2 \leq \frac{1}{2} \|Ax^* - b\|_2^2 + \lambda (\|x^*\|_1 - \|x^*\|_2) = F(x^*) \leq F(x^0) = \frac{1}{2} \|b\|_2^2. \]
Combining (3.14) and (3.15), we obtain
\[ \sqrt{\|x^*\|_0} \leq \frac{\|A\|_2 \|b\|_2}{\lambda} + 1 \quad \text{or} \quad \|x^*\|_0 \leq \left( \frac{\|A\|_2 \|b\|_2}{\lambda} + 1 \right)^2. \]
Moreover,
\[ \left( \frac{\|A\|_2 \|b\|_2}{\lambda} + 1 \right)^2 < s + 1 \quad \iff \quad \lambda > \lambda_s := \frac{\|A\|_2 \|b\|_2}{\sqrt{s + 1} - 1} \]
In other words, if \( \lambda > \lambda_s \), then \( \|x^*\|_0 < s + 1 \). \( \|x^*\|_0 \) and \( s \) are integers, so \( \|x^*\|_0 \leq s \).

3.3. Solving the subproblem. In each DCA iteration, it requires solving a \( \ell_1 \)-regularized convex subproblem of the following form

\[ (3.16) \quad \min_{x \in \mathbb{R}^n} \frac{1}{2} \|Ax - b\|_2^2 + \langle x, v \rangle + \lambda \|x\|_1, \]

where \( v \in \mathbb{R}^n \) is a constant vector. This problem can be done by the alternating direction method of multipliers (ADMM), a versatile algorithm first introduced in [32, 29]. A recent result on the \( O(1/n) \) convergence rate of ADMM was established in [36]. Just like the split Bregman [33], the trick of ADMM form is to decouple the coupling between the quadratic term and \( \ell_1 \) penalty in (3.16). Specifically, (3.16) can be reformulated as

\[ \min_{x, z \in \mathbb{R}^n} \frac{1}{2} \|Ax - b\|_2^2 + \langle x, v \rangle + \lambda \|z\|_1 \quad \text{subject to} \quad x - z = 0, \]

We then form the augmented Lagrangian
\[ \mathcal{L}_\delta(x, z, y) = \frac{1}{2} \|Ax - b\|_2^2 + \langle x, v \rangle + \lambda \|z\|_1 + y^T(x - z) + \frac{\delta}{2} \|x - z\|_2^2, \]
where \( y \) is the Lagrange multiplier, \( \delta > 0 \) is the penalty parameter. ADMM consists of the iterations:

\[ \begin{align*}
x^{l+1} &= \arg \min_x \mathcal{L}_\delta(x, z^l, y^l) \\
z^{l+1} &= \arg \min_z \mathcal{L}_\delta(x^{l+1}, z, y^l) \\
y^{l+1} &= y^l + \delta (x^{l+1} - z^{l+1})
\end{align*} \]

The first two steps have closed-form solutions which are detailed in Algorithm 2. In the \( z \)-update step, \( \mathcal{S}(x, r) \) denotes the soft-thresholding operator given by

\[ (\mathcal{S}(x, r))_i = \text{sgn}(x_i) \max\{|x_i| - r, 0\}. \]

The computational complexity of Algorithm 2 mainly lies in the \( x \)-update step. Since \( A \in \mathbb{R}^{m \times n} \) and \( b \in \mathbb{R}^m \), the computational complexity of \( A^T A + \delta I \) is \( O(mn^2 + n) = O(mn^2) \) and that of \( A^T b - v + \delta z - y^k \) is \( O(mn + n) = O(mn) \). Moreover, the inversion of matrix \( A^T A + \delta I \) requires \( O(n^3) \). Therefore, the computational complexity of Algorithm 2 per iteration is \( O(n^3 + mn^2 + mn) = O(n^3 + mn^2) \).

According to [5], a stopping criterion of Algorithm 2 is given by:

\[ \|x^l\|_2 \leq \sqrt{m} \epsilon_{ab} + \epsilon_{rel} \max\{\|x^l\|_2, \|z^l\|_2\}, \quad \|z^l\|_2 \leq \sqrt{m} \epsilon_{ab} + \epsilon_{rel} \|y^l\|_2, \]

where \( x^l = x^l - z^l, s^l = \delta (z^l - z^{l-1}) \) are primal and dual residuals respectively at the \( l \)-th iteration. \( \epsilon_{ab} > 0 \) is an absolute tolerance and \( \epsilon_{rel} > 0 \) a relative tolerance.
Algorithm 2 ADMM for subproblem (3.16).

Define $x^0$, $z^0$ and $u^0$.
for $l = 0, 1, 2, \cdots$, MAXit do
\[ x^{l+1} = (A^T A + \delta I)^{-1} (A^T b - v + \delta z^l - y^l) \]
\[ z^{l+1} = S(x^{l+1} + y^l / \delta, \lambda / \delta) \]
\[ y^{l+1} = y^l + \delta (x^{l+1} - z^{l+1}) \]
end for

4. Hybrid Simulated Annealing. In this section, we employ a technique called simulated annealing (SA) to traverse local minima to reach a global solution. Combining the DCA with SA, we propose a hybrid simulated annealing DCA. There are many generic SA algorithms, see Kirkpatrick [37], Geman and Geman [30], Hajek [35], Gidas [31], and the references therein. In addition, this technique has many applications to image processing, such as Carnevali et al. [11].

The term "annealing" is analogous to the cooling of a liquid or solid in a physical system. Consider the problem of minimizing the cost function $F(x)$. Simulated annealing algorithm begins with an initial solution and iteratively generates new ones, each of which is randomly selected among the "neighborhood" of the previous state. If the new solution is better than the previous one, it is accepted; otherwise, it is accepted with certain probability. The probability of accepting a new state is given by
\[ \exp \left( - \frac{F(x_{new}) - F(x_{curr})}{T} \right) > \alpha, \]
where $\alpha$ is a random number between 0 and 1, and $T$ is a temperature parameter. The algorithm usually starts with a high temperature, and then gradually goes down to 0. The cooling must be slow enough so that the system does not get stuck into local minima of $F(x)$. The hybrid simulated annealing algorithm can be summarized as follows:

1. Choose an initial temperature $T$, an initial state $x_{curr}$, and evaluate $F(x_{curr})$.
2. Randomly determine a new state $x_{new}$, and run the DCA to get the near optimal solution $DCA(x_{new})$.
3. Evaluate $F(DCA(x_{new}))$. If $F(DCA(x_{new})) < F(x_{curr})$, accept $DCA(x_{new})$, i.e., $x_{curr} = DCA(x_{new})$; otherwise, accept $DCA(x_{new})$ if $\exp \left( - \frac{F(DCA(x_{new})) - F(x_{curr})}{T} \right) > \alpha$, where $\alpha$ is a random number between 0 and 1.
4. Repeat Steps 2 and 3 for some iterations with temperature $T$.
5. Lower $T$ according to the annealing schedule, and return to step 2. Continue this process until some criteria of convergence is satisfied.

There are two important aspects in implementing simulated annealing. One is how to lower the temperature $T$. Kirkpatrick [37] suggests that $T$ decays geometrically in the number of cooling phases. Hajek [35] proves that if $T$ decreases at the rate of $\frac{d}{\log(k)}$, where $k$ is the number of iterations and $d$ is some certain constant, then the probability distribution for the algorithm converges to the set of global minimum points with probability one. In our algorithm, we follow Hajek’s [35] method by decreasing $T$ at the rate of $\frac{d}{\log(k)}$, with some constant $d$.

Another aspect is how to advance to a new state based on the current one in Step 2. One of the most common methods is to add random Gaussian noise, such as the method in [51]. We generate the Gaussian perturbation by the following probability density function
\[ p(x, T_k) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi(T_k + \beta)}} \exp \left( -x_i^2 / 2(T_k + \beta) \right), \]

where the temperature $T_k = \frac{T_0}{\log(k)}$, $\beta$ is a constant. We assume that there is only one iteration in each cooling scheme, i.e., the temperature $T_k$ decreases after each iteration. Then we have
the following theorem.

**Theorem 4.1.** If we choose the \( k \)-th new state by the probability density function given in (4.1), then the hybrid simulated annealing algorithm converges to the global minimum \( F^* \) in probability.

**Proof.** The proof is similar to Corollary 1 in [51]. We assume that there exists a bounded set \( E \subset \mathbb{R}^n \) containing all the global minima. Let \( r_i = \max_{x,y \in E} |y_i - x_i|, 1 \leq i \leq n \), it suffices to show that there exist constants \( C > 0 \) and \( k_0 > 0 \), such that \( \min_{x,y \in E} p(y - x, T_k) \geq \frac{C}{r} \), \( \forall k > k_0 \). Take \( k_0 = 1 \) and \( C = \frac{(2\pi)^{n/2}(T_0 + \beta)^{n/2} \exp(||r||^2/2\beta)}{1} \), since \( 0 \leq T_k \leq T_0 \), we have the following inequalities for all \( k \geq 1 \):

\[
\min_{x,y \in E} p(y - x, T_k) = \min \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi(T_k + \beta)}} \exp\left(-\frac{(y_i - x_i)^2}{2(T_k + \beta)}\right) \\
\geq \frac{1}{(2\pi)^{n/2}(T_k + \beta)^{n/2} \exp(||r||^2/2\beta)} \\
\geq \frac{1}{(2\pi)^{n/2}(T_0 + \beta)^{n/2} \exp(||r||^2/2\beta)} \\
= C \geq C/k.
\]

\( \square \)

However, due to the presence of a large number of local minima, this Gaussian perturbation method would converge slowly. To overcome this difficulty, we propose a novel perturbation method. First, let us define the space \( V = \{ x \in \mathbb{R}^n : x_i = 0,1 \ or \ -1 \} \) and \( V_s = \{ x \in V : ||x||_0 \leq s \} \). Note that given any \( x \in \mathbb{R}^n \), \( sgn(x) \in V_s \), where the sign function applies element wise. Define the mapping \( W \) from \( \mathbb{R}^n \) to \( V_s \) by \( W(x) = \text{sgn}(S(x, \nu)) \) for some small number \( \nu > 0 \), where \( S \) is the soft-thresholding operator. Denote the DCA function by \( DCA(x) \) being output of DCA initialized by \( x \). We randomly choose the new state \( x_{\text{new}} \in V_s \) for some \( 0 < s < n \), such that \( ||x_{\text{new}} - W(DCA(x_{\text{curr}}))||_2 \leq \eta \).

The idea of this perturbation method is to keep the sparse properties of the current state and perturb inside the space \( V_s \). Hence, we call this perturbation method the sparse perturbation method.

To prove the convergence to global minima of this hybrid simulated annealing with sparse perturbation, we shall work with the following assumption.

**Assumption 4.1.** If \( x^* \) is a global minimizer of the cost function \( F \) over \( \mathbb{R}^n \), then there is a global minimizer of the cost function \( J(x) := F(DCA(x)) \) denoted by \( x^*_J \in \arg \min_{x \in V_s} J(x) \), such that \( x^* = DCA(x^*_J) \).

The above assumption says that a global minimizer of \( F \) can be reached by a local DCA descent from a global minimizer of \( J \) defined over a smaller set \( V_s \) whose elements are vectors with components \( 0, \pm 1 \). This assumption is akin to an interesting property of the Bregman iteration of \( \ell_1 \) minimization [56] where if the \( n \)-th step iteration gets the signs and support of an \( \ell_1 \) minimizer, the minimizer is reached at the \( (n+1) \)-th step. Though one could minimize \( F \) directly as stated in Theorem 4.1, the passage to a global minimum of \( F \) from that of \( J \) via DCA is observed to be a short cut in our numerical experiments, largely because the global minima of \( F \) in our problem are sparse. Under this assumption, we aim to show that the sequence \( W(x_{\text{curr}}) \) converges to a global minimizer of \( J \) over space \( V_s \). By our algorithm, for each state \( x \in V_s \), we have a neighborhood of \( x \), \( U(x) \subset V_s \), where we generate the next state. We also assume that there is a transition probability matrix \( Q \) such that \( Q(x,y) > 0 \) if and only if \( y \in U(x) \).

We only need one iteration in each cooling scheme, because the temperature \( T_k \) decreases after each iteration, and \( \lim_{k \to \infty} T_k = 0 \). Denote the sequence of states by \( x_{\text{curr}}^1, x_{\text{curr}}^2, \cdots \) and the initial state by \( x_0 \). Define \( y_k = W(x_{\text{curr}}^k) \). Given \( y^k = i \), a new potential next state \( x_{\text{new}}^k \)
Under the above assumptions, the state sequence \( \{y^k\} \) generated by SA algorithm satisfies

\[
\lim_{k \to \infty} \lim_{T_k \to T} P(y^k \in \mathbb{V}^*) = 1.
\]

**Proof.** Since the temperature \( T_k = T \) for all \( k \), the sequence \( y^k \) is a homogeneous Markov chain. Assume that its associated transition matrix is \( P_T \). Define a probability distribution by

\[
\pi_T(i) = \frac{a(i)}{Z_T} \exp(-\frac{J(i)}{T}),
\]

where \( Z_T = \sum_i a(i) \exp(-\frac{J(i)}{T}) \). A simple computation shows that \( \pi_T = \pi_T P_T \). So \( \pi_T \) is the invariant distribution for the Markov chain \( y^k \). By the reversibility of \( Q \) and the irreducibility of \( y^k \), the Markov ergodic convergence theorem implies that

\[
P(y^k \in \mathbb{V}^*) = \sum_{i \in \mathbb{V}^*} \pi_T(i).
\]

Since \( J(i) > J(j) \) for all \( i \in \mathbb{V} \setminus \mathbb{V}^* \), \( j \in \mathbb{V}^* \), and \( \lim_{T \to 0} \pi_T(i) = 0 \) for all \( i \in \mathbb{V} \setminus \mathbb{V}^* \), we have

\[
\lim_{T \to 0} \lim_{T_k \to T} P(y^k \in \mathbb{V}^*) = 1.
\]

\qed
To extend the convergence result to the case where \( \lim_{k \to \infty} T_k = 0 \), we introduce the following concept.

**Definition 4.1.** We say that the state \( i \) communicates with \( V^* \) at height \( h \) if there exists a path in \( V \) such that the largest value of \( J \) along the path is \( J(i) + h \).

The main theorem is given as follows.

**Theorem 4.2.** Let \( d^* \) be the smallest number such that every \( i \in V \) communicates with \( V^* \) at height \( d^* \). Then the SA algorithm sequence \( y_k \) converges to the global minima set \( V^* \) with probability one, if and only if

\[
\sum_{k=1}^{\infty} \exp(-d^*/T_k) = \infty.
\]

The detailed proof is given in [35]. By the theorem, we just need to choose \( T_k = d/\log(k) \), where \( d \geq d^* \). To estimate \( d^* \), we can simply set \( d^* = |V_s| \), since \( V_s \) is a finite space.

5. **Numerical Results.** In this section, we present numerical experiments to demonstrate the efficiency of DCA-\( \ell_{1-2} \) method. We will compare it with the following state-of-the-art CS solvers:

- **ADMM-lasso** [5] that solves the lasso problem (1.2) by ADMM.
- A greedy method termed CoSaMP [43] which involves a sequence of support detections and least squares.
- The accelerated version of IHT (AIHT) [4] that solves

\[
\min_{x \in \mathbb{R}^n} \|Ax - b\|_2^2 \quad \text{subject to} \quad \|x\|_0 \leq s.
\]

by hard thresholding iterations.

- An improved IRLS-\( \ell_p \) algorithm [38] that solves the unconstrained \( \ell_p \) problem with \( 0 < p < 1 \):

\[
(5.1) \quad \min_{x \in \mathbb{R}^n} \frac{1}{2}\|Ax - b\|_2^2 + \lambda\|x\|_p^p.
\]

- Reweighted \( \ell_1 \) [10] which is at heart a non-convex CS solver based on IRL1 algorithm attempting to solve

\[
\min_{x \in \mathbb{R}^n} \sum_{i=1}^{n} \log(|x_i| + \varepsilon) \quad \text{subject to} \quad Ax = b.
\]

- **Half thresholding** [52] (Scheme 2) for \( \ell_{1/2} \) regularization, i.e., (5.1) with \( p = 0.5 \).

Note that all the proposed methods except ADMM-lasso are non-convex in nature.

**Sensing matrix for tests.** We will test the commonly used random Gaussian matrix, which is defined as

\[
A_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I_m/m), \quad i = 1, \cdots, n
\]

and random partial discrete cosine transform (DCT) matrix

\[
A_i = \frac{1}{\sqrt{m}} \cos(2i\pi\xi), \quad i = 1, \cdots, n
\]

where \( \xi \in \mathbb{R}^m \sim \mathcal{U}([0, 1]^m) \) whose components are uniformly and independently sampled from \([0, 1]\). These sensing matrices fit for CS, being incoherent and having small RIP constants with high probability.
We also test more ill-conditioned sensing matrix of significantly higher coherence. Specifically, a randomly oversampled partial DCT matrix $A$ is defined as

$$A_i = \frac{1}{\sqrt{m}} \cos\left(\frac{2\pi i \xi}{F}\right), \quad i = 1, \cdots, n$$

where $\xi \in \mathbb{R}^m \sim U([0,1]^m)$ and $F \in \mathbb{N}$ is the refinement factor. Actually it is the real part of the random partial Fourier matrix analyzed in [25]. The number $F$ is closely related to the conditioning of $A$ in the sense that $\mu(A)$ tends to get larger as $F$ increases. For $A \in \mathbb{R}^{100 \times 2000}$, $\mu(A)$ easily exceeds 0.99 when $F = 10$. This quantity is above 0.9999 when $F$ increases to 20. Although $A$ sampled in this way does not have good RIP by any means, it is still possible to recover the sparse vector $\bar{x}$ provided its spikes are sufficiently separated. Specifically, we randomly select the elements of $\text{supp}(\bar{x})$ so as to satisfy the following condition:

$$\min_{j,k \in \text{supp}(\bar{x})} |j - k| \geq L.$$ 

Here $L$ is called the minimum separation. For traditional inversion methods to work, it is necessary for $L$ to be at least 1 Rayleigh length (RL) which is unity in the frequency domain [25, 20]. In our case, the value of 1 RL is nothing but $F$.

5.1. Selection of parameters. The regularization parameter $\lambda$ controls data fitting and sparsity of the solution. For noiseless case, a tiny value should be chosen. When measurements are noisy, a reasonable $\lambda$ should depend on the noise level. In this case, $\lambda$ needs to be tuned empirically (typically by cross-validation technique). Although our convergence result is established on the assumption that the sequence of subproblems is solved exactly by Algorithm 2, it suffices for practical use that the relative tolerance $\epsilon^{\text{rel}}$ and absolute tolerance $\epsilon^{\text{abs}}$ are adequately small. Fig. 2 shows that in the noiseless case, the relative error $\frac{\|x^* - \bar{x}\|_2}{\|\bar{x}\|_2}$ is linear in the tolerance at moderate sparsity level when $\epsilon^{\text{rel}} \leq 10^{-3}$ and $\epsilon^{\text{abs}} = 10^{-2} \epsilon^{\text{rel}}$. Here $\bar{x}$ is the test signal and $x^*$ is the recovered one by DCA-$\ell_1 - 2$ from the measurements $b = A\bar{x}$. $\delta$ in Algorithm 2 should be well chosen, since sometimes the convergence can be sensitive to its value. [5] suggests to vary $\delta$ by iteration, aiming to stabilize the ratio between primal and dual residuals as they both go to zero. We adopt this strategy when having noise in measurements. More precisely,

$$\delta^{l+1} = \begin{cases} 
2\delta^l & \text{if } \|r^l\|_2 > 10\|s^l\|_2, \\
\delta^l/2 & \text{if } 10\|r^l\|_2 < \|s^l\|_2, \\
\delta^l & \text{otherwise.}
\end{cases}$$

Recall that $r^l$ and $s^l$ are the primal and dual residuals respectively. In noiseless case where $\lambda$ is always set to a small number, it turns out that just taking $\delta = 10\lambda$ works well enough.

5.2. Exact recovery of sparse vectors. In the noiseless case, we compare the proposed methods in terms of success percentage and computational cost.

Test on RIP matrix. We carry out the experiments as follows. After sampling a sensing matrix $A \in \mathbb{R}^{m \times n}$, we generate a test signal $\bar{x} \in \mathbb{R}^n$ of sparsity $s$ supported on a random index set with i.i.d. Gaussian entries. We then compute the measurement $b = A\bar{x}$ and apply each solver to produce a reconstruction $x^*$ of $\bar{x}$. The reconstruction is considered as a success if the relative error

$$\frac{\|x^* - \bar{x}\|_2}{\|\bar{x}\|_2} < 10^{-3}.$$ 

We run 100 independent realizations and record the corresponding success rates at various sparsity levels.
Minimization of $\ell_{1-2}$ for Compressed Sensing

Fig. 2: Relative error v.s. relative tolerance using random Gaussian matrix (blue) and oversampled DCT matrix (red) with $F = 10, L = 2F$. Relative tolerance $\epsilon^{rel} = 10^{-1}, 10^{-2}, \cdots, 10^{-6}$ and absolute tolerance $\epsilon^{abs} = 10^{-2}\epsilon^{rel}$. $A \in \mathbb{R}^{128 \times 1024}, \|\bar{x}\|_0 = 24, \lambda = 10^{-8}$ and $\delta = 10^{-7}$. The relative error is averaged over 10 independent trials.

We chose $\epsilon^{abs} = 10^{-7}$ and $\epsilon^{rel} = 10^{-5}$ for DCA-$\ell_{1-2}$ in Algorithm 2. In the outer stopping criterion (3.3) in Algorithm 1, we set $\epsilon = 10^{-2}$. MAXoit and MAXit are 10 and 5000 respectively. For ADMM-lasso, we let $\lambda = 10^{-6}$, $\beta = 1$, $\rho = 10^{-5}$, $\epsilon^{abs} = 10^{-7}$, $\epsilon^{rel} = 10^{-5}$, and the maximum number of iterations maxiter = 5000. For CoSaMP, maxiter = 50, the tolerance tol = $10^{-8}$. The tol for AIHT was $10^{-12}$. For IRLS-$\ell_p$, $p = 0.5$, maxiter = 1000, tol = $10^{-8}$. For reweighted $\ell_1$, the smoothing parameter $\varepsilon$ was adaptively updated as introduced in [10], and the outer stopping criterion adopted was the same as that of the DCA-$\ell_{1-2}$. We solved its weighted $\ell_1$ minimization subproblems using the more efficient YALL1 solver (available at http://yall1.blogs.rice.edu/) instead of the default $\ell_1$-MAGIC. The tolerance for YALL1 was set to $10^{-6}$. For half thresholding, we let maxiter = 5000. In addition, CoSaMP, AIHT and half thresholding require an estimate on the sparsity of $\bar{x}$, which we set to the ground truth. All other settings of the algorithms were set to default ones.

Fig. 3 depicts the success rates of the proposed methods with $m = 64$ and $n = 256$. For both Gaussian matrix and partial DCT matrix, IRLS-$\ell_p$ with $p = 0.5$ has the best performance, followed by reweighted $\ell_1$. DCA-$\ell_{1-2}$ is comparable to half thresholding and CoSaMP, which outperform both ADMM-lasso and AIHT.

Test on highly coherent matrix. Fixing the size of $A$ at 100-by-2000 and $L = 2F$, we repeat the experiment and present the results in Fig. 4 for $F = 10$ (left) and 20 (right). Note that in this example, the task of non-convex CS has become more challenging since ill-conditioning of the sensing matrix $A$ makes it much easier for the solvers to stall at spurious local minima. Here we do not take CoSaMP and AIHT into consideration in the comparison, because preliminary results show that even with $\bar{x}$ at low sparsity level, they do not work for matrix of large coherence at all (in terms of exact reconstruction). In this example, the DCA-$\ell_{1-2}$ is the best and provides robust performance regardless of large coherence of $A$. In contrast, the other non-convex solvers clearly encounter the trapping of local minima and perform worse than the convex ADMM-lasso. Moreover, by comparing the two plots in Fig. 4, one can tell that their
Fig. 3: Success rates using incoherent sensing matrix. $m = 64, n = 256, s = 10, 12, \ldots, 36$. Left: random Gaussian matrix. Right: random partial DCT matrix.

Fig. 4: Success rates using randomly oversampled partial DCT matrices with $m = 100, n = 2000, s = 5, 7, 9, \ldots, 35$, minimum separation $L = 2F$.

reconstruction qualities suffer a decline as $A$ becomes more and more coherent.

The left plot of Fig. 5 shows the success rates for DCA-$\ell_{1-2}$ with/without aid of hybrid simulated annealing (HSA) methods. For each HSA method, we apply at most 100 iterations. The matrix size is $100 \times 1500$, $F = 20$, and the minimum separation $L = 2F$. We also compare the two different perturbation methods, referred to as HSA with Gaussian perturbations and HSA with sparse perturbations. Both of these HSA methods can improve the reconstruction capability of the plain DCA-$\ell_{1-2}$. However, HSA with sparse perturbations has the best performance. On the other hand, though the limit point of DCA-$\ell_{1-2}$ is not known theoretically to be a global minimum, in practice it is quite close. This can be seen from Fig. 5 where the additional improvement from the HSA is at most about 15% in the intermediate sparsity regime.
Minimization of $\ell_{1-2}$ for Compressed Sensing

Fig. 5: Left: Comparison of success rates of HSA algorithms using randomly oversampled partial DCT matrices with $m = 100$, $n = 1500$, $s = 15, 17, \ldots, 35$, minimum separation $L = 2F$. Right: Comparison of CPU time using random Gaussian matrix with $n = 2^8, 2^9, \ldots, 2^{13}$, $m = n/4$, $s = m/8$. All the resulting relative errors were roughly $10^{-5}$ except those of CoSaMP (about $10^{-15}$).

Comparison of time efficiency under Gaussian measurements. The comparison of CPU time using random Gaussian sensing matrix and non-convex CS solvers is presented in the right plot of Fig. 5. For each $n$, we fix $m = n/4$ and $s = m/8$ and run 10 independent realizations. Parameters (mainly the tolerances) for the algorithms were tuned such that all resulting relative errors were roughly $10^{-5}$, for which except CoSaMP. CoSaMP stands out as its success relies on correct identification of the support of $\bar{x}$. Once the support is correctly identified, followed by least squares minimization, it naturally produces a solution of perfect accuracy (tiny relative error) which is close to the machine precision. It turns out that AIHT enjoy the best overall performances in terms of time consumption, being slightly faster than CoSaMP. But CoSaMP did provide substantially higher quality solutions in the absence of noise. When $n > 1000$, DCA-$\ell_{1-2}$ is faster than the other regularization methods like IRLS-$\ell_p$ and half thresholding. This experiment was carried out on a laptop with 16GB RAM memory and 2.40 GHz Intel Core i7 CPU.

5.3. Robust recovery in presence of noise. In this example, we show robustness of DCA-$\ell_{1-2}$ in the noisy case. White Gaussian noise is added to the clean data $A\bar{x}$ to get contaminated measurements $b$, by calling $b = \text{awgn}(A\bar{x}, \text{snr})$ in MATLAB, where snr corresponds to the value of signal-to-noise ratio (SNR) measured in dB. We then obtain the reconstruction $x^*$ using DCA-$\ell_{1-2}$, and compute the SNR of reconstruction given by

$$10 \log_{10} \frac{\|x^* - \bar{x}\|_2^2}{\|\bar{x}\|_2^2},$$

with $\frac{\|x^* - \bar{x}\|_2^2}{\|\bar{x}\|_2^2}$ being the relative mean squared error. Varying amount of the noise, we test DCA-$\ell_{1-2}$, ADMM-lasso, half thresholding, CoSaMP, and AIHT on both Gaussian matrix and the ill-conditioned oversampled DCT. For Gaussian measurements, we chose $n = 1024$, $m = 256$, and $s = 48$. For oversampled DCT, $n = 2000$, $m = 100$, $s = 15$, $F = 20$, and the minimum separation $L = 2F$. At each noise level, we run 50 times and record the average SNR of reconstruction (in dB).
Penghang Yin, Yifei Lou, Qi He and Jack Xin

<table>
<thead>
<tr>
<th>snr (dB)</th>
<th>DCA-ℓ₁-²</th>
<th>ADMM-lasso</th>
<th>half thresholding</th>
<th>CoSaMP</th>
<th>AIHT</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>-29.2398</td>
<td>-27.6103</td>
<td>-37.3863</td>
<td>-32.8737</td>
<td>-36.9471</td>
</tr>
<tr>
<td>10</td>
<td>-3.6700</td>
<td>-3.1970</td>
<td>-1.4126</td>
<td>1.1522</td>
<td>-1.4048</td>
</tr>
</tbody>
</table>

Table 1: SNR of reconstruction (dB) under Gaussian measurements. $n = 1024, m = 256, s = 48$. Each recorded value is the mean of 50 random realizations.

<table>
<thead>
<tr>
<th>snr (dB)</th>
<th>DCA-ℓ₁-²</th>
<th>ADMM-lasso</th>
<th>half thresholding</th>
<th>AIHT</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td><strong>-35.2119</strong></td>
<td>-25.9895</td>
<td>-3.8896</td>
<td>-3.8393</td>
</tr>
<tr>
<td>35</td>
<td><strong>-17.0934</strong></td>
<td>-12.2916</td>
<td>-4.2793</td>
<td>-3.7375</td>
</tr>
<tr>
<td>20</td>
<td><strong>-3.1806</strong></td>
<td>-3.0157</td>
<td>-2.6428</td>
<td>-0.8141</td>
</tr>
</tbody>
</table>

Table 2: SNR of reconstruction (dB) using overampled DCT matrix. $n = 2000, m = 100, s = 15, F = 10, L = 2F$. Each recorded value is the mean of 50 random realizations.

Table 1 shows the results under Gaussian measurements. Choosing an appropriate value of $\lambda$ is necessary for both DCA-ℓ₁-² and ADMM-lasso being well functional, for which we employ a "trial and error" strategy. CoSaMP and AIHT do not need such a parameter, whereas half thresholding embraces a self-adjusting $\lambda$ during iterations. As a trade-off, however, they all require an estimate on the sparsity of $\bar{x}$, for which we used the true value in the experiment. With this piece of crucial information, it then appears reasonable that they perform better than DCA-ℓ₁-² and ADMM-lasso when there is not much noise, producing relatively smaller SNR of reconstruction. Table 2 shows the results for oversampled DCT with $F = 10$. In this case, we do not display the result for CoSaMP since it yields huge errors. Half thresholding and AIHT are not robust as suggested by Table 2. In contrast, DCA-ℓ₁-² and ADMM-lasso perform much better, still doing the job under moderate amount of noise. Nevertheless, due to the large coherence of $A$, their performance have sunk compared to that in the Gaussian case. In either case, DCA-ℓ₁-² consistently beats ADMM-lasso.

5.4. MRI reconstruction. We present a two-dimensional example of reconstructing Magnetic Resonance Imaging (MRI) from a limited number of projections. It is first introduced in [6] to demonstrate the success of compressed sensing. The signal/image is a Shepp-Logan phantom of size $256 \times 256$, as shown in Fig. 6. In this case, it is the gradient of the signal that is sparse, and therefore the work [6] is to minimize ℓ₁ norm of the gradient, or the so-called total variation (TV),

\[
\min \| \nabla u \|_1 \quad \text{subject to} \quad RFu = f
\]

where $F$ denotes the Fourier transform, $R$ is the sampling mask in the frequency space and $f$ is the data. It is claimed in [6] that 22 projections are necessary to have exact recovery, while we find 10 projections suffice by using the split Bregman method.

Our proposed ℓ₁-² on the gradient is expressed as

\[
\min |\partial_x u| + |\partial_y u| - \sqrt{|\partial_x u|^2 + |\partial_y u|^2} \quad \text{subject to} \quad RFu = f
\]
It is anisotropic TV. We apply the technique of DCA by linearizing the $\ell_2$ norm of the gradient, 

$$u^{k+1} = \arg \min_{u_d, u_y, d_x, d_y} |d_x| + |d_y| - \frac{(d_x, d_y)^T (\partial_x u^k, \partial_y u^k)}{\sqrt{\|\partial_x u^k\|^2 + \|\partial_y u^k\|^2}}$$  

(5.4) 

$$+ \frac{\mu}{2} \|RFu - f\|_2^2 + \frac{\lambda}{2} \|d_x - \partial_x u\|_2^2 + \frac{\lambda}{2} \|d_y - \partial_y u\|_2^2$$

Let $(t_x, t_y) = (\partial_x u^k, \partial_y u^k)/\sqrt{\|\partial_x u^k\|^2 + \|\partial_y u^k\|^2}$ at the current step $u^k$. The subproblem to obtain a new solution $u^{k+1}$ can be solved by the split Bregman method, as detailed in Algorithm 3. Note that the matrix to be inverted in the algorithm is diagonal.

**Algorithm 3** The split Bregman method to solve (5.4)

Define $u = d_x = d_y = b_x = b_y = 0, z = f$ and MAXinner, MAXouter. Let $D$ and $D^T$ be forward and backward difference operators respectively.

for 1 to MAXinner do
  for 1 to MAXouter do
    $u = (\mu R^T R - \lambda \Delta R^T \Delta)^{-1} \mu R^T Rz + \lambda D_x^T (d_x - b_x) + \lambda D_y^T (d_y - b_y)$
    $d_x = S(D_x u + b_x + t_x/\lambda, 1/\lambda)$
    $d_y = S(D_y u + b_y + t_y/\lambda, 1/\lambda)$
    $b_x = b_x + D_x u - d_x$
    $b_y = b_y + D_y u - d_y$
  end for
  $z = z + f - RFu$
end for

Fig. 6 shows the exact recovery of 8 projections using the proposed method. We also compare with the classical filtered-back projection (FBP) and $\ell_1$ on the gradient or TV minimization, whose relative errors are 0.99 and 0.1 respectively. A similar work is reported in [12], where 10 projections are required for $\ell_p$ ($p=0.5$) on the gradient.

6. Concluding Remarks. We have studied compressed sensing problems under a non-convex Lipschitz continuous metric $\ell_1$ in terms of exact and stable sparse signal recovery under RIP condition for the constrained problem and full rank property of the restricted sensing matrix for the unconstrained problem. We also presented iterative minimization method based on DCA, its convergence to a stationary point, and its almost sure convergence to a global minimum with the help of a simulated annealing procedure. If the sensing matrix is well-conditioned, computational examples suggest that IRLS-$\ell_p$ ($p = 1/2$) is the best in terms of the success rates of sparse signal recovery. For highly coherent matrix, DCA-$\ell_1$ becomes the best. In either regime, DCA-$\ell_1$ is always better than ADMM-lasso. The MRI phantom image recovery test also indicates that $\ell_{1/2}$ outperforms $\ell_{1/2}$ and $\ell_1$.

In future work, we plan to investigate further why $\ell_{1/2}$ improves on $\ell_1$ in a robust manner.

**Appendix: Proof of Lemma 2.1.**

**Lemma 2.1.** (a) The upper bound is immediate from Cauchy-Schwarz inequality. To show the lower bound, without loss of generality, let us assume 

$$|x_1| \geq |x_2| \geq \cdots \geq |x_n|.$$ 

Let $t = \lfloor \sqrt{n} \rfloor$, then we have 

$$||x|| = \sum_{i=1}^{t} |x_i| + (\sqrt{n} - t)|x_{t+1}|.$$ 

(6.1)
Shepp-Logan phantom sampled projections

FBP, ER=0.99

$L_1$, $ER = 0.1$

$L_1 - L_2$, $ER = 5 \times 10^{-8}$

Fig. 6: MRI reconstruction results. It is demonstrated that 8 projections are enough to have exact recovery using $\ell_1 - \ell_2$. The relative errors are provided for each method.

To see this, we square both sides

$$\sum_{i=1}^{n} |x_i|^2 \leq \sum_{i=1}^{t} |x_i|^2 + \sum_{i=1}^{t} \sum_{j=1, j \neq i}^{t} |x_i||x_j| + 2(\sqrt{n} - t)|x_{t+1}| \sum_{i=1}^{t} |x_i| + (\sqrt{n} - t)^2|x_{t+1}|^2$$

or equivalently,

$$\sum_{i=t+1}^{n} |x_i|^2 \leq \sum_{i=1}^{t} \sum_{j=1, j \neq i}^{t} |x_i||x_j| + 2(\sqrt{n} - t)|x_{t+1}| \sum_{i=1}^{t} |x_i| + (\sqrt{n} - t)^2|x_{t+1}|^2$$

Then (6.1) holds because

$$\sum_{i=t+1}^{n} |x_i|^2 \leq \sum_{i=t+1}^{n} |x_{i+1}|^2 = (n - t)|x_{t+1}|^2$$
Minimization of $\ell_{1-2}$ for Compressed Sensing

and

$$
\sum_{i=1}^{t} \sum_{j=1, j \neq i}^{t} |x_i| |x_j| + 2(\sqrt{n} - t)|x_{t+1}| \sum_{i=1}^{t} |x_i| + (\sqrt{n} - t)^2 |x_{t+1}|^2
$$

$$
\geq \sum_{i=1}^{t} \sum_{j=1, j \neq i}^{t} |x_{t+1}|^2 + 2(\sqrt{n} - t)|x_{t+1}| \sum_{i=1}^{t} |x_{t+1}| + (\sqrt{n} - t)^2 |x_{t+1}|^2
$$

$$
= (t^2 - t) + 2(\sqrt{n} - t^2) + (\sqrt{n} - t)^2 |x_{t+1}|^2
$$

$$
= (n - t)|x_{t+1}|^2.
$$

It follows from (6.1) that

$$
\|x\|_1 - \|x\|_2 \geq \|x\|_1 - \left( \sum_{i=1}^{t} |x_i| + (\sqrt{n} - t)|x_{t+1}| \right) = (t + 1 - \sqrt{n})|x_{t+1}| + \sum_{i=t+2}^{n} |x_i|
$$

$$
\geq (t + 1 - \sqrt{n})|x_n| + \sum_{i=t+2}^{N} |x_n| = (n - \sqrt{n})|x_n|
$$

$$
= (n - \sqrt{n}) \min_i |x_i|.
$$

(b) Note that $\|x\|_1 - \|x\|_2 = \|x_{\Lambda}\|_1 - \|x_{\Lambda}\|_2$, (b) follows as we apply (a) to $x_{\Lambda}$.

(c) If $\|x\|_1 - \|x\|_2 = 0$, then by (b)

$$
0 = \|x\|_1 - \|x\|_2 \geq (s - \sqrt{s}) \min_{i \in \Lambda} |x_i|
$$

So $s - \sqrt{s} \leq 0$ and thus $s = 1$.

The other direction is trivial.

Acknowledgments. The authors would like to thank Professor Wotao Yin of the Department of Mathematics, UCLA, for providing us with Matlab code of IRLS-$\ell_p$ algorithm published in [38]. We thank Dr. Ernie Esser for helpful conversations. We also thank anonymous referees for their constructive comments.

REFERENCES


Minimization of $\ell_{1-2}$ for Compressed Sensing


