SIAM J. SCI. COMPUT.
Vol. 37, No. 1, pp. A536–A563

MINIMIZATION OF ℓ_{1-2} FOR COMPRESSED SENSING*

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

Abstract. We study minimization of the difference of ℓ_1 and ℓ_2 norms as a nonconvex 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 ℓ_{1-2} minimization based on the difference of convex functions algorithm and prove that it converges to a stationary point satisfying the first-order optimality condition. We propose a sparsity oriented simulated annealing 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 nonconvex compressed sensing solvers in the literature. Likewise in the magnetic resonance imaging (MRI) phantom image recovery problem, ℓ_{1-2} succeeds with eight projections. Irrespective of the conditioning of the sensing matrix, ℓ_{1-2} is better than ℓ_1 in both the sparse signal and the MRI phantom image recovery problems.

Key words. ℓ_{1-2} minimization, compressed sensing, nonconvex, difference of convex functions algorithm, simulate annealing

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

DOI. 10.1137/140952363

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 ℓ_1 minimization or the basis pursuit (BP) problem [6, 15]:

$$\min_x \|x\|_1 \quad \text{subject to} \quad Ax = b. \tag{1.1}$$

In the above formulation, the ℓ_1 norm works as the convex relaxation of ℓ_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

*Submitted to the journal's Methods and Algorithms for Scientific Computing section January 13, 2014; accepted for publication (in revised form) December 1, 2014; published electronically February 24, 2015. This work was partially supported by NSF grants DMS-0928427 and DMS-1222507.
http://www.siam.org/journals/sisc/37-1/95236.html

†Department of Mathematics, University of California Irvine, Irvine, CA 92697 (penghany@uci.edu, ylou1@math.uci.edu, qhe2@uci.edu, jxin@math.uci.edu).

imaging problems (radar and medical imaging) when the grid spacing is below the Rayleigh threshold [25], ℓ_1 minimization may not give the sparsest solution [25, 55].

In the past decade, great efforts have been devoted to exploring efficient and stable algorithms for solving the BP problem and its associated ℓ_1-regularized problem (also called lasso [47]):

$$\min_x \ \frac{1}{2}\|Ax-b\|_2^2+\lambda\|x\|_1, \tag{1.2}$$

where $\lambda > 0$ is a free parameter. The 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 the lasso problem (1.2), such as the split Bregman [33], which is equivalent to ADMM [5, 27], FPC [34], FISTA [2], and others [49, 54, 58, 50].

Nonconvex (concave) functions, such as the ℓ_p (quasi-)norm ($p < 1$) [12, 13] and the log-det functional [10], have also been proposed as alternatives to ℓ_0. Such non-Lipschitz continuous metrics usually require additional smoothing in minimization to avoid division by zero and to enhance sparsity. A general class of penalty functions satisfying unbiasedness, sparsity, and continuity can be found in [24, 41]. While nonconvex metrics are generally more challenging to minimize, they have advantages over the convex ℓ_1 norm. Simply put, nonconvex 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 nonconvex 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 ℓ_0 minimization include compressive sampling matching pursuit (CoSaMP) [43], which is a greedy method among variants of orthogonal matching pursuit [48], the iterative hard thresholding (IHT) algorithm [4, 3], and the penalty decomposition method [42], whereas iteratively reweighted least squares (IRLS) [19, 14, 38] and iteratively reweighted ℓ_1 (IRL1) [17, 61, 10, 26] can be applied to minimize nonconvex proxies of ℓ_0. Particularly for minimization of the ℓ_p norm, empirical studies [14, 53] show that whenever $p \in [1/2, 1)$, the smaller the p, the sparser the solutions by minimizing the ℓ_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 nonconvex yet Lipschitz continuous metric ℓ_{1-2} for sparse signal recovery and compare it with various CS solvers. ℓ_{1-2} was first addressed in [28] by Esser, Lou, and Xin in the context of nonnegative least squares problems and group sparsity with applications to spectroscopic imaging. A contour plot of the ℓ_{1-2} metric can be seen in Figure 1. Here we mainly discuss the constrained ℓ_{1-2} minimization problem,

$$\min_{x\in\mathbb{R}^n} \|x\|_1-\|x\|_2 \quad \text{subject to} \quad Ax=b, \tag{1.3}$$

and the unconstrained one,

$$\min_{x\in\mathbb{R}^n} \frac{1}{2}\|Ax-b\|_2^2+\lambda(\|x\|_1-\|x\|_2), \tag{1.4}$$

where $A \in \mathbb{R}^{m\times n}$ is an underdetermined sensing matrix of full row rank and $b \in \mathbb{R}^m \setminus \{\mathbf{0}\}$. We shall focus on the theoretical aspects such as sparsity of minimizers

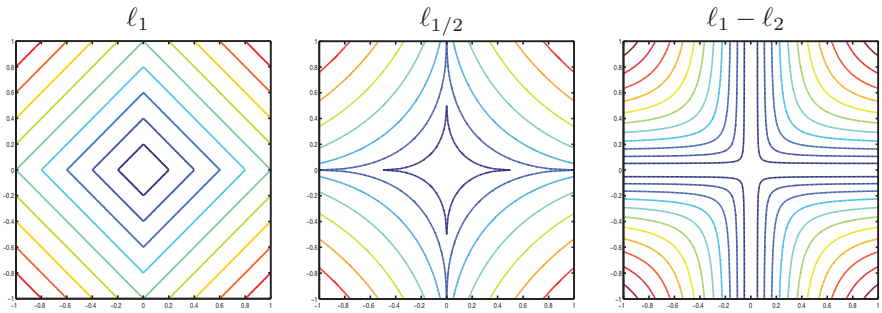

FIG. 1. *Contours of three sparsity metrics. The level curves of $\ell_1 - \ell_2$ approach the x and y axes as the values get small, hence promoting sparsity.*

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 ℓ_{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 ℓ_{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-ℓ_{1-2} method with various CS solvers numerically. For ill-conditioned matrices A, such as an oversampled discrete cosign transform (DCT) matrix, DCA-ℓ_{1-2} is the best, as seen from the success rate versus sparsity plot. In this regime of A, exact recovery is still possible provided that the peaks of the 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 256×256. Using ℓ_{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 2×10^6 larger under the split Bregman for ℓ_1. The concluding remarks are in section 6.

Notation. Let us fix some notation. For any $x, y \in \mathbb{R}^n$, $\langle x, y\rangle = x^{\mathrm{T}} y$ is their inner product. $\mathrm{supp}(x) := \{1 \le i \le n : x_i \neq 0\}$ denotes the support of x, and $\|x\|_0 := |\,\mathrm{supp}(x)|$ is cardinality of $\mathrm{supp}(x)$. $\mathbf{B}_r(x) = \{y \in \mathbb{R}^n : \|y - x\|_2 < r\}$ 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 $\mathrm{sgn}(x)$ is the signum function defined as

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

2. Theory of ℓ_{1-2} minimization.

2.1. Preliminaries. RIP, introduced by Candès and Tao [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*

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

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 δ_s.

Sensing matrices with small δ_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 [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,*

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

Coherence is closely related to the RIP yet is 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 $x \in \mathbb{R}^n \setminus \{\mathbf{0}\}$, $\Lambda = \mathrm{supp}(x)$ 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 ℓ_{1-2} minimization, demonstrating the capability of ℓ_{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 condition

$$\delta_{3s} + a(s)\delta_{4s} < a(s) - 1; \tag{2.1}$$

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.,

$$\|x\|_1 - \|x\|_2 \leq \|\bar{x}\|_1 - \|\bar{x}\|_2. \tag{2.2}$$

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

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

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

On the other hand,

$$\begin{aligned}
&\|\bar{x} + v_\Lambda + v_{\Lambda^c}\|_1 - \|\bar{x} + v_\Lambda + v_{\Lambda^c}\|_2 \\
&\quad = \|\bar{x} + v_\Lambda\|_1 + \|v_{\Lambda^c}\|_1 - \|\bar{x} + v_\Lambda + v_{\Lambda^c}\|_2 \\
&\quad \ge \|\bar{x}\|_1 - \|v_\Lambda\|_1 + \|v_{\Lambda^c}\|_1 - \|\bar{x}\|_2 - \|v_\Lambda\|_2 - \|v_{\Lambda^c}\|_2.
\end{aligned}$$

So v must obey the following inequality constraint:

$$\|v_\Lambda\|_1 + \|v_\Lambda\|_2 \ge \|v_{\Lambda^c}\|_1 - \|v_{\Lambda^c}\|_2. \tag{2.3}$$

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

$$\begin{aligned}
0 = \|Av\|_2 &= \left\| A_{\Lambda_0} v_{\Lambda_0} + \sum_{i=2}^{l} A_{\Lambda_i} v_{\Lambda_i} \right\|_2 \ge \|A_{\Lambda_0} v_{\Lambda_0}\|_2 - \left\| \sum_{i=2}^{l} A_{\Lambda_i} v_{\Lambda_i} \right\|_2 \\
&\ge \|A_{\Lambda_0} v_{\Lambda_0}\|_2 - \sum_{i=2}^{l} \|A_{\Lambda_i} v_{\Lambda_i}\|_2 \\
&\ge \sqrt{1-\delta_{4s}} \|v_{\Lambda_0}\|_2 - \sqrt{1+\delta_{3s}} \sum_{i=2}^{l} \|v_{\Lambda_i}\|_2 \qquad (2.4)
\end{aligned}$$

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

$$|v_t| \le \min_{r \in \Lambda_{i-1}} |v_r| \le \frac{\|v_{\Lambda_{i-1}}\|_1 - \|v_{\Lambda_{i-1}}\|_2}{3s - \sqrt{3s}},$$

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

$$\|v_{\Lambda_i}\|_2 \le \sqrt{3s} \frac{\|v_{\Lambda_{i-1}}\|_1 - \|v_{\Lambda_{i-1}}\|_2}{3s - \sqrt{3s}} = \frac{\|v_{\Lambda_{i-1}}\|_1 - \|v_{\Lambda_{i-1}}\|_2}{\sqrt{3s} - 1}$$

and

$$\sum_{i=2}^{l} \|v_{\Lambda_i}\|_2 \le \sum_{i=1}^{l-1} \frac{\|v_{\Lambda_i}\|_1 - \|v_{\Lambda_i}\|_2}{\sqrt{3s} - 1} \le \frac{\sum_{i=1}^{l} \|v_{\Lambda_i}\|_1 - \sum_{i=1}^{l} \|v_{\Lambda_i}\|_2}{\sqrt{3s} - 1}. \tag{2.5}$$

Note that in (2.5)

$$\sum_{i=1}^{l} \|v_{\Lambda_i}\|_1 = \|v_{\Lambda^c}\|_1 \quad \text{and} \quad \sum_{i=1}^{l} \|v_{\Lambda_i}\|_2 \ge \sqrt{\sum_{i=1}^{l} \|v_{\Lambda_i}\|_2^2} = \|v_{\Lambda^c}\|_2.$$

Combining (2.5) and (2.3) gives

$$\sum_{i=2}^{l} \|v_{\Lambda_i}\|_2 \le \frac{\|v_{\Lambda^c}\|_1 - \|v_{\Lambda^c}\|_2}{\sqrt{3s} - 1} \le \frac{\|v_\Lambda\|_1 + \|v_\Lambda\|_2}{\sqrt{3s} - 1} \le \frac{(\sqrt{s} + 1)\|v_\Lambda\|_2}{\sqrt{3s} - 1} = \frac{\|v_\Lambda\|_2}{\sqrt{a(s)}}.$$

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 $v_{\Lambda_0} = \mathbf{0}$. This implies $v = \mathbf{0}$, which completes the proof. ∎

Remark 2.1. Equation (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

$$\delta_{3s} + 3\delta_{4s} < 2, \tag{2.6}$$

or equivalently

$$\delta_{3s} < 3(1-\delta_{4s}) - 1.$$

The condition (2.1) required for ℓ_{1-2} exact recovery appears more stringent than (2.6) for ℓ_1 recovery since $a(s) < 3$ (and thus also stronger than the RIP for ℓ_p recovery with $0 < p < 1$ [12]). However, this does not mean the ℓ_1 norm is superior to ℓ_{1-2} in terms of sparsity promoting. On the contrary, in section 5 it will be shown numerically that the ℓ_{1-2} penalty consistently outperforms ℓ_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 $a(s) = (\frac{\sqrt{3s}-1}{\sqrt{s}+1})^2 > 1$, which implies $s \geq 8$. Then a natural question is whether the uniqueness of $\bar{x}$ 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 $\bar{x}$ 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 $(\frac{\sqrt{6s}-1}{\sqrt{s}+1})^2$. It is easy to check that $a(s) > 1$ for $s \geq 2$. By a similar argument, we can show that the following RIP condition suffices for the uniqueness of $\bar{x}$:

$$\delta_{6s} + a(s)\delta_{7s} < a(s) - 1.$$

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

THEOREM 2.2. *Under the assumptions of Theorem* 2.1 *except that* $b = A\bar{x} + e$, *where* $e \in \mathbb{R}^m$ *is any perturbation with* $\|e\|_2 \leq \tau$, *we have that the solution* x^{opt} *to the 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 $\|x^{\text{opt}} - \bar{x}\|_2 \leq C_s\tau$ *for some constant* $C_s > 0$ *depending on* δ_{3s} *and* δ_{4s}.

Proof. Let Λ be the support of $\bar{x}$ and $x^{\text{opt}} = \bar{x} + v$. Then starting from $\|x^{\text{opt}}\|_1 - \|x^{\text{opt}}\|_2 \le \|\bar{x}\|_1 - \|\bar{x}\|_2$ and repeating the arguments in the proof of Theorem 2.1, we obtain

$$\sum_{i=2}^{l} \|v_{\Lambda_i}\|_2 \le \frac{\|v_\Lambda\|_2}{\sqrt{a(s)}} \tag{2.7}$$

and

$$\|Av\|_2 \ge \left(\sqrt{1-\delta_{4s}} - \frac{\sqrt{1+\delta_{3s}}}{\sqrt{a(s)}} \right) \|v_{\Lambda_0}\|_2. \tag{2.8}$$

From (2.7) it follows that

$$\|v\|_2 = \sqrt{\|v_{\Lambda_0}\|_2^2 + \sum_{i=2}^{l} \|v_{\Lambda_i}\|_2^2} \le \sqrt{\|v_{\Lambda_0}\|_2^2 + \frac{\|v_\Lambda\|_2^2}{a(s)}} \le \sqrt{1 + \frac{1}{a(s)}} \|v_{\Lambda_0}\|_2,$$

so (2.8) becomes

$$\|Av\|_2 \ge \frac{\sqrt{a(s)(1-\delta_{4s})} - \sqrt{1+\delta_{3s}}}{\sqrt{1+a(s)}} \|v\|_2. \tag{2.9}$$

On the other hand, since $\|A\bar{x} - b\|_2 \le \tau$ and $\|Ax^{\text{opt}} - b\|_2 \le \tau$, by the triangular inequality,

$$\|Av\|_2 = \|(Ax^{\text{opt}} - b) - (A\bar{x} - b)\|_2 \le \|A\bar{x} - b\|_2 + \|Ax^{\text{opt}} - b\|_2 \le 2\tau. \tag{2.10}$$

Combining (2.9) and (2.10), we have $\|v\|_2 \le C_s \tau$, where

$$C_s := \frac{2\sqrt{1+a(s)}}{\sqrt{a(s)(1-\delta_{4s})} - \sqrt{1+\delta_{3s}}} > 0. \qquad \square$$

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 δ_{3s} and δ_{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 ℓ_{1-2} will rule out redundant columns of A. It is worth noting that similar results were proved in [9] for the ℓ_p unconstrained problem using the second-order optimality condition. This demonstrates an advantage of nonconvex sparsity metrics over the convex ℓ_1 norm.

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

Proof. The proof simply uses the definition of local minimizer. Suppose the columns of A_{Λ^*} are linearly dependent; then there exists $v \in \ker(A) \setminus \{0\}$ such that $\text{supp}(v) \subseteq \Lambda^*$. For any fixed neighborhood $\mathbf{B}_r(x^*)$ of x^*, we scale v so that

$$\|v\|_2 < \min\left\{ \min_{i \in \Lambda^*} |x_i^*|, r \right\}.$$

Consider two feasible vectors in $\mathbf{B}_r(x^*)$, $\hat{x} = x^* + v$ and $\check{x} = x^* - v$. Since $\text{supp}(v) \subseteq \Lambda^*$, we have $\text{supp}(\hat{x}) \subseteq \Lambda^*$ and $\text{supp}(\check{x}) \subseteq \Lambda^*$. 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 $\text{sgn}(\hat{x}_i) = \text{sgn}(\check{x}_i) = \text{sgn}(x_i^*)\ \forall i \in \Lambda^*$ 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 $\check{x}$ are located in the same orthant. It follows that

$$\|x^*\|_1 = \frac{1}{2}\|\hat{x} + \check{x}\|_1 = \frac{1}{2}\|\hat{x}\|_1 + \frac{1}{2}\|\check{x}\|_1 \tag{2.11}$$

and

$$\|x^*\|_2 = \frac{1}{2}\|\hat{x} + \check{x}\|_2 < \frac{1}{2}\|\hat{x}\|_2 + \frac{1}{2}\|\check{x}\|_2. \tag{2.12}$$

Equation (2.11) holds since $\hat{x}$ and $\check{x}$ are in the same orthant, and (2.12) holds because of the fact that $\hat{x}$ and $\check{x}$ are not collinear since they both satisfy the linear constraint $A\,x = b$. So

$$\begin{aligned}\|x^*\|_1 - \|x^*\|_2 &> \frac{1}{2}(\|\hat{x}\|_1 - \|\hat{x}\|_2 + \|\check{x}\|_1 - \|\check{x}\|_2) \\ &\geq \min\{\|\hat{x}\|_1 - \|\hat{x}\|_2, \|\check{x}\|_1 - \|\check{x}\|_2\},\end{aligned}$$

which contradicts the assumption that x^* is a minimizer in $\mathbf{B}_r(x^*)$. ∎

Local minimizers of the unconstrained problem share the same property.

THEOREM 2.4. *Let x^* be a local minimizer of the unconstrained problem* (1.4). *Then the columns of A_{Λ^*} 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 $\check{x} \in \mathbf{B}_r(x^*)$ such that $A\check{x} = A\,x^*$ and

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

This implies

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

Thus for any $r > 0$, we actually find a $\check{x} \in \mathbf{B}_r(x^*)$ 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_{Λ^*} are linearly independent. ∎

By Theorems 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* $\text{supp}(x) \subseteq \Lambda^*$, *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 ℓ_{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

$$\begin{cases} y^k \in \partial H(x^k), \\ x^{k+1} = \arg\min_{x\in\mathbb{R}^n} G(x) - (H(x^k) + \langle y^k, x - x^k\rangle), \end{cases}$$

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

$$\begin{aligned} F(x^k) = G(x^k) - H(x^k) &\geq G(x^{k+1}) - (H(x^k) + \langle y^k, x^{k+1} - x^k\rangle) \\ &\geq G(x^{k+1}) - H(x^{k+1}) = F(x^{k+1}). \end{aligned}$$

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:

$$F(x) = \left(\frac{1}{2}\|Ax - b\|_2^2 + \lambda\|x\|_1\right) - \lambda\|x\|_2. \tag{3.1}$$

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

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

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

$$\frac{\|x^{k+1} - x^k\|_2}{\max\{\|x^k\|_2, 1\}} < \epsilon \tag{3.3}$$

for some given parameter $\epsilon > 0$.

The DCA in general does not guarantee a global minimum due to the nonconvex 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 the 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 ℓ_1 problem. So basically we are minimizing ℓ_{1-2} on top of ℓ_1, which possibly explains why we observed in the experiments that ℓ_{1-2} regularization initialized by $x^0 = \mathbf{0}$ always outperforms ℓ_1 regularization. Hereby we summarize DCA-ℓ_{1-2} in Algorithm 1 below.

ALGORITHM 1. DCA-ℓ_{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
    $x^{k+1} = \arg\min_{x\in\mathbb{R}^n} \ \frac{1}{2}\|Ax - b\|_2^2 + \lambda\|x\|_1 - \left\langle x, \lambda\frac{x^k}{\|x^k\|_2}\right\rangle$
  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 $\ker(A^{\mathrm{T}}A)$ is nontrivial, 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 $\|x\|_2 \to \infty$, and therefore $F(x)$ is coercive in the sense that the level set $\{x \in \mathbb{R}^n : F(x) \le F(x^0)\}$ is bounded $\forall x^0 \in \mathbb{R}^n$.*

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

$$\begin{aligned} F(cx) &= \frac{1}{2}\|cAx - b\|_2^2 + c\lambda(\|x\|_1 - \|x\|_2) \\ &\ge \frac{1}{2}(c\|Ax\|_2 - \|b\|_2)^2 + c\lambda(\|x\|_1 - \|x\|_2). \end{aligned}$$

If $Ax = \mathbf{0}$, i.e., $x \in \ker(A) \setminus \{\mathbf{0}\}$, since $\operatorname{rank}(A) = m$, we have $\|x\|_0 \ge m + 1 \ge 2$. Lemma 2.1(c) implies $\|x\|_1 - \|x\|_2 > 0$, so

$$F(cx) = \frac{1}{2}\|b\|_2^2 + c\lambda(\|x\|_1 - \|x\|_2) \to \infty \quad \text{as} \quad c \to \infty.$$

If $Ax \neq \mathbf{0}$, the claim follows as we notice that $c\|Ax\|_2 - \|b\|_2 \to \infty$ as $c \to \infty$. ∎

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

(3.4)
$$F(x^k) - F(x^{k+1}) \ge \frac{1}{2}\|A(x^k - x^{k+1})\|_2^2 + \lambda(\|x^{k+1}\|_2 - \|x^k\|_2 - \langle y^k, x^{k+1} - x^k\rangle) \ge 0,$$

where $y^k \in \partial\|x^k\|_2$.

Proof. A simple calculation shows

$$\begin{aligned} F(x^k) - F(x^{k+1}) = \frac{1}{2}\|A(x^{k+1} - x^k)\|_2^2 + \langle A(x^k - x^{k+1}), Ax^{k+1} - b\rangle \\ + \lambda(\|x^k\|_1 - \|x^{k+1}\|_1 - \|x^k\|_2 + \|x^{k+1}\|_2). \end{aligned} \tag{3.5}$$

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

$$A^{\mathrm{T}}(Ax^{k+1}-b)+\lambda(w^{k+1}-y^k)=0. \tag{3.6}$$

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

$$\langle A(x^k-x^{k+1}),Ax^{k+1}-b\rangle+\lambda(\langle w^{k+1},x^k\rangle-\|x^{k+1}\|_1)+\langle y^k,x^{k+1}-x^k\rangle=0, \tag{3.7}$$

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

$$\begin{aligned}F(x^k)-F(x^{k+1})&=\frac{1}{2}\|A(x^{k+1}-x^k)\|_2^2+\lambda(\|x^k\|_1-\langle w^{k+1},x^k\rangle)\\&\quad+\lambda(\|x^{k+1}\|_2-\|x^k\|_2-\langle y^k,x^{k+1}-x^k\rangle)\\&\geq\frac{1}{2}\|A(x^{k+1}-x^k)\|_2^2+\lambda(\|x^{k+1}\|_2-\|x^k\|_2-\langle y^k,x^{k+1}-x^k\rangle)\\&\geq 0.\end{aligned}$$

In the first inequality above, $\|x^k\|_1-\langle w^{k+1},x^k\rangle\geq 0$ since $|w_i^{k+1}|\leq 1\ \forall 1\leq i\leq n$, 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. *Letting $\{x^k\}$ 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*

$$\mathbf{0}\in A^{\mathrm{T}}(A\,x^*-b)+\lambda\left(\partial\|x^*\|_1-\frac{x^*}{\|x^*\|_2}\right), \tag{3.8}$$

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

$$F(\mathbf{0})-F(x^1)\geq\lambda\|x^1\|_2>0,$$

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

$$\|A(x^k-x^{k+1})\|_2\to 0, \tag{3.9}$$

$$\|x^{k+1}\|_2-\frac{\langle x^k,x^{k+1}\rangle}{\|x^k\|_2}\to 0. \tag{3.10}$$

We define $c^k := \frac{\langle x^k, x^{k+1}\rangle}{\|x^k\|_2^2}$ and $e^n := x^{k+1} - c^k x^k$. Then it suffices to prove $e^k \to \mathbf{0}$ and $c^k \to 1$. It is straightforward to check that

$$\|e^k\|_2^2 = \|x^{k+1}\|_2^2 - \frac{\langle x^k, x^{k+1}\rangle^2}{\|x^k\|_2^2} \to 0,$$

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 $Ax^{k_j} \to \mathbf{0}$. 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 $c^k \to 1$, $e^k \to \mathbf{0}$, and thus $x^{k+1} - x^k \to \mathbf{0}$ as $k \to \infty$.

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

$$\mathbf{0} \in A^{\mathrm{T}}(Ax^{k_j} - b) + \lambda\partial\|x^{k_j}\|_1 - \lambda\frac{x^{k_j-1}}{\|x^{k_j-1}\|_2}$$

or

$$-\left(A^{\mathrm{T}}(Ax^{k_j} - b) - \lambda\frac{x^{k_j-1}}{\|x^{k_j-1}\|_2}\right) \in \lambda\partial\|x^{k_j}\|_1. \tag{3.11}$$

Here $\partial\|x\|_1 = \prod_{i=1}^n \mathrm{SGN}(x_i) \subset \mathbb{R}^n$ with

$$\mathrm{SGN}(x_i) := \begin{cases} \{\mathrm{sgn}(x_i)\} & \text{if } x_i \neq 0, \\ [-1,1] & \text{otherwise.} \end{cases}$$

Since, by (b), $x^{k_j} \to x^*$, we have, when k_j is sufficiently large, $\mathrm{supp}(x^*) \subseteq \mathrm{supp}(x^{k_j})$ and $\mathrm{sgn}(x_i^{k_j}) = \mathrm{sgn}(x_i^*)\ \forall i \in \mathrm{supp}(x^*)$, which implies

$$\partial\|x^{k_j}\|_1 \subseteq \partial\|x^*\|_1.$$

Then by (3.11), for large k_j we have

$$-\left(A^{\mathrm{T}}(Ax^{k_j} - b) - \lambda\frac{x^{k_j-1}}{\|x^{k_j-1}\|_2}\right) \in \lambda\partial\|x^*\|_1. \tag{3.12}$$

Moreover, since x^* is away from $\mathbf{0}$,

$$\begin{aligned}
&\lim_{k_j\to\infty} A^{\mathrm{T}}(Ax^{k_j} - b) - \lambda\frac{x^{k_j-1}}{\|x^{k_j-1}\|_2} \\
&\quad= \lim_{k_j\to\infty} A^{\mathrm{T}}(Ax^{k_j} - b) - \lambda\frac{x^{k_j}}{\|x^{k_j}\|_2} + \lambda\left(\frac{x^{k_j}}{\|x^{k_j}\|_2} - \frac{x^{k_j-1}}{\|x^{k_j-1}\|_2}\right) \\
&\quad= A^{\mathrm{T}}(Ax^* - b) - \lambda\frac{x^*}{\|x^*\|_2}.
\end{aligned}$$

Let $k_j \to \infty$ 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. *For all $s \in \mathbb{N}$, there exists $\lambda_s > 0$ such that for any parameter $\lambda > \lambda_s$ in* (1.4)*, we have $\|x^*\|_0 \le s$, where $x^* \neq \mathbf{0}$ is a stationary point generated by Algorithm* 1.

Proof. By the optimality condition (3.8), there exists $w^* \in \partial\|x^*\|_1$ satisfying

$$w_i^* = \begin{cases} \operatorname{sgn}(x_i^*) & \text{if } i \in \operatorname{supp}(x^*), \\ \in [-1,1] & \text{otherwise} \end{cases} \tag{3.13}$$

such that

$$-A^{\mathrm{T}}(Ax^* - b) = \lambda\left(w^* - \frac{x^*}{\|x^*\|_2}\right).$$

Since by (3.13) $\|w^*\|_2 \ge \sqrt{\|x^*\|_0}$, taking the ℓ_2 norm of both sides gives

$$\|A^{\mathrm{T}}(Ax^* - b)\|_2 = \lambda\left\|w^* - \frac{x^*}{\|x^*\|_2}\right\|_2 \ge \lambda\left(\|w^*\|_2 - \|\frac{x^*}{\|x^*\|_2}\|_2\right) \ge \lambda\left(\sqrt{\|x^*\|_0} - 1\right). \tag{3.14}$$

On the other hand,

$$\|A^{\mathrm{T}}(Ax^* - b)\|_2 \le \|A^{\mathrm{T}}\|_2\|Ax^* - b\|_2 = \|A\|_2\|Ax^* - b\|_2 \le \|A\|_2\|b\|_2, \tag{3.15}$$

where we used

$$\frac{1}{2}\|Ax^* - b\|_2 \le \frac{1}{2}\|A\,x^* - b\|_2 + \lambda(\|x^*\|_1 - \|x^*\|_2) = F(x^*) \le F(x^0) = \frac{1}{2}\|b\|_2.$$

Combining (3.14) and (3.15), we obtain

$$\sqrt{\|x^*\|_0} \le \frac{\|A\|_2\|b\|_2}{\lambda} + 1 \quad \text{or} \quad \|x^*\|_0 \le \left(\frac{\|A\|_2\|b\|_2}{\lambda} + 1\right)^2.$$

Moreover,

$$\left(\frac{\|A\|_2\|b\|_2}{\lambda} + 1\right)^2 < s + 1 \quad \Longleftrightarrow \quad \lambda > \lambda_s := \frac{\|A\|_2\|b\|_2}{\sqrt{s+1} - 1}.$$

In other words, if $\lambda > \lambda_s$, then $\|x^*\|_0 < s + 1$. $\|x^*\|_0$ and s are integers, so $\|x^*\|_0 \le s$. ∎

3.3. Solving the subproblem. Each DCA iteration requires solving a ℓ_1-regularized convex subproblem of the following form:

$$\min_{x\in\mathbb{R}^n} \frac{1}{2}\|Ax - b\|_2^2 + \langle x, v\rangle + \lambda\|x\|_1, \tag{3.16}$$

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 the ADMM form is to decouple the coupling between the quadratic term and the ℓ_1 penalty in (3.16). Specifically,

(3.16) can be reformulated as

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

We then form the augmented Lagrangian

$$\mathcal{L}_\delta(x,z,y) = \frac{1}{2}\|Ax-b\|_2^2 + \langle x, v\rangle + \lambda\|z\|_1 + y^{\mathrm{T}}(x-z) + \frac{\delta}{2}\|x-z\|_2^2,$$

where y is the Lagrange multiplier, and $\delta > 0$ is the penalty parameter. ADMM consists of the iterations

$$\begin{cases} x^{l+1} = \arg\min_x \ \mathcal{L}_\delta(x, z^l, y^l), \\ z^{l+1} = \arg\min_z \ \mathcal{L}_\delta(x^{l+1}, z, y^l), \\ y^{l+1} = y^l + \delta(x^{l+1} - z^{l+1}). \end{cases}$$

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

$$(\mathcal{S}(x,r))_i = \operatorname{sgn}(x_i)\max\{|x_i| - r, 0\}.$$

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 $A^{\mathrm{T}}A + \delta I$ is $O(mn^2 + n) = O(mn^2)$ and that of $A^{\mathrm{T}}b - v + \delta z^k - y^k$ is $O(mn + n) = O(mn)$. Moreover, the inversion of matrix $A^{\mathrm{T}}A + \delta I$ requires $O(n^3)$. Therefore, the computational complexity of Algorithm 2 per iteration is $O(n^3 + mn^2 + mn) = O(n^3 + mn^2)$.

ALGORITHM 2. ADMM FOR SUBPROBLEM (3.16).

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

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

$$\|r^l\|_2 \le \sqrt{n}\epsilon^{\mathrm{abs}} + \epsilon^{\mathrm{rel}}\max\{\|x^l\|_2, \|z^l\|_2\}, \quad \|s^l\|_2 \le \sqrt{n}\epsilon^{\mathrm{abs}} + \epsilon^{\mathrm{rel}}\|y^l\|_2,$$

where $r^l = x^l - z^l$, $s^l = \delta(z^l - z^{l-1})$ are primal and dual residuals, respectively, at the lth iteration. $\epsilon^{\mathrm{abs}} > 0$ is an absolute tolerance and $\epsilon^{\mathrm{rel}} > 0$ a relative tolerance.

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 SA (HSA) DCA. There are many generic SA algorithms; see Kirkpatrick, Gelatt, and Vecchi [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, Coletti, and Patarnello [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 $F(x)$. The SA 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

$\exp(-\frac{F_{\text{new}}-F_{\text{curr}}}{T}) > \alpha$, where α 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 HSA algorithm can be summarized as follows:

1. Choose an initial temperature T and an initial state x_{curr}, and evaluate $F(x_{\text{curr}})$.
2. Randomly determine a new state x_{new}, and run the DCA to get the near optimal solution $DCA(x_{\text{new}})$.
3. Evaluate $F(DCA(x_{\text{new}}))$. If $F(DCA(x_{\text{new}})) < F(x_{\text{curr}})$, accept $DCA(x_{\text{new}})$, i.e., $x_{\text{curr}} = DCA(x_{\text{new}})$; otherwise, accept $DCA(x_{\text{new}})$ if $\exp(-\frac{F(DCA(x_{\text{new}}))-F(x_{\text{curr}})}{T}) > \alpha$, where α 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 SA. One is how to lower the temperature T. Kirkpatrick, Gelatt, and Vecchi [37] suggest 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:

$$p(x, T_k) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi(T_k+\beta)}} \exp(-x_i^2/2(T_k+\beta)), \tag{4.1}$$

where the temperature $T_k = \frac{T_0}{\log(k)}$, and β 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 kth new state by the probability density function given in* (4.1)*, then the HSA 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. Letting $r_i = \max_{x,y\in E} |y_i - x_i|, 1 \le i \le n$, it suffices to show that there exist constants $C > 0$ and $k_0 > 0$ such that $\min_{x,y\in E} p(y-x, T_k) \ge \frac{C}{k}\ \forall k > k_0$. Taking $k_0 = 1$ and $C = \frac{1}{(2\pi)^{N/2}(T_0+\beta)^{N/2}\exp(\|r\|_2^2/2\beta)}$, since $0 \le T_k \le T_0$, we have the following inequalities for all $k \ge 1$:

$$\begin{aligned}
\min_{x,y\in E} p(y-x, T_k) &= \min \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi(T_k+\beta)}} \exp(-(y_i-x_i)^2/2(T_k+\beta)) \\
&\ge \frac{1}{(2\pi)^{n/2}(T_k+\beta)^{n/2}\exp(\|r\|_2^2/2(T_k+\beta))} \\
&\ge \frac{1}{(2\pi)^{n/2}(T_0+\beta)^{n/2}\exp(\|r\|_2^2/2\beta)} \\
&= C \ge C/k. \qquad \square
\end{aligned}$$

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 $\mathbb{V} = \{x \in \mathbb{R}^n : x_i = 0, 1 \text{ or } -1\}$ and $\mathbb{V}_s = \{x \in \mathbb{V} : \|x\|_0 \le s\}$. Note that given any $x \in \mathbb{R}^n$, $\operatorname{sgn}(x) \in \mathbb{V}$, where the signum function applies elementwise. Define the mapping W from $\mathbb{R}^n$ to $\mathbb{V}$ by $W(x) = \operatorname{sgn}(\mathcal{S}(x, \nu))$ for some small number $\nu > 0$, where $\mathcal{S}$ is the soft-thresholding operator. Denote the DCA function by $DCA(x)$ being an output of DCA initialized by x. We randomly choose the new state $x_{\text{new}} \in \mathbb{V}_s$ for some $0 < s < n$ such that $\|x_{\text{new}} - W(DCA(x_{\text{curr}}))\|_2 \le \eta$.

The idea of this perturbation method is to keep the sparse properties of the current state and perturb inside the space $\mathbb{V}_s$. Hence, we call this perturbation method the sparse perturbation method.

To prove the convergence to global minima of this hybrid SA 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 $x_J^* \in \arg\min_{x \in \mathbb{V}} J(x)$ 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 $\mathbb{V}$ whose elements are vectors with components $0, \pm 1$. This assumption is akin to an interesting property of the Bregman iteration of ℓ_1 minimization [56], where if the nth step iteration gets the signs and support of an ℓ_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 shortcut 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 $\mathbb{V}_s$. By our algorithm, for each state $x \in \mathbb{V}$, we have a neighborhood of x, $\mathrm{U}(x) \subset \mathbb{V}$, 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 $y \in \mathrm{U}(x)$.

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 $x_{\text{curr}}^1, x_{\text{curr}}^2, \ldots$ 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_{new}^k is chosen from the neighborhood set $\mathrm{U}(i)$ with the conditional probability $\mathbf{P}(x_{\text{new}}^k = j | y^k = i) = Q(i, j)$. Then we update the algorithm as follows. If $J(x_{\text{new}}^{k+1}) \le J(y^k)$, $y^{k+1} = x_{\text{new}}^{k+1}$. If $J(x_{\text{new}}^{k+1}) > J(y^k)$,

$$y^{k+1} = \begin{cases} x_{\text{new}}^{k+1} & \text{with probability } \exp(-(J(x_{\text{new}}^{k+1}) - J(y^k))/T_k), \\ x_{\text{new}}^k & \text{otherwise.} \end{cases} \tag{4.2}$$

In summary,

$$\mathbf{P}(y^{k+1} = j | y^k = i) = Q(i, j) \exp\left(-\frac{\max(J(j) - J(i)), 0)}{T_k}\right)$$

for $j \in \mathrm{U}(i) \subset \mathbb{V}$.

By the above updating method, the SA algorithm is best understood as a nonhomogeneous Markov chain y^k in which the transition matrix is dependent on the temperature T_k. Denote the set of the global minimizers of J on $\mathbb{V}$ by $\mathbb{V}^*$. We aim to prove that

$$\lim_{k\to\infty} \mathbf{P}(y^k \in \mathbb{V}^*) = 1.$$

To motivate the rationale behind the SA algorithm, we assume that the temperature T_k is kept at constant value T. In addition, we assume that y_k is irreducible, which means that for any two states $i, j \in \mathbb{V}$, we can choose a sequence of states $y^0 = i, y^1, \ldots, y^l = j$ for some $l \geq 1$ such that $y^{k+1} \in \mathrm{U}(y^k), 1 \leq k \leq l-1$. We also assume that Q is reversible, i.e., there is a distribution $a(i)$ on $\mathbb{V}$ such that $a(i)Q(i,j) = a(j)Q(j,i)\ \forall i, j \in \mathbb{V}$. One simple choice for Q is

$$Q(i,j) = \begin{cases} \frac{1}{|\mathrm{U}(i)|} & \text{if } j \in \mathrm{U}(i), \\ 0 & \text{otherwise.} \end{cases} \tag{4.3}$$

We then introduce the following lemma.

LEMMA 4.1. *Under the above assumptions, the state sequence $\{y^k\}$ generated by the SA algorithm satisfies*

$$\lim_{T \to 0} \lim_{\substack{k \to \infty \\ T_k = T}} \mathbf{P}(y^k \in \mathbb{V}^*) = 1.$$

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

$$\pi_T(i) = \frac{a(i)}{Z_T} \exp\left(-\frac{J(i)}{T}\right),$$

where $Z_T = \sum_i a(i) \exp(-\frac{J(i)}{T})$. A simple computation shows that $\pi_T = \pi_T P_T$. So π_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

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

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

$$\lim_{T \to 0} \lim_{\substack{k \to \infty \\ T_k = T}} \mathbf{P}(y^k \in \mathbb{V}^*) = 1. \qquad \square$$

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 $\mathbb{V}^*$ at height h if there exists a path in $\mathbb{V}$ such that the largest value of J alone 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 $i \in \mathbb{V}$ communicates with $\mathbb{V}$ at height d^*. Then the SA algorithm sequence y^k converges to the global minima set $\mathbb{V}^*$ 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 $d^* = |\mathbb{V}_s|$, since $\mathbb{V}_s$ is a finite space.

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

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

$$\min_{x\in\mathbb{R}^n} \|Ax-b\|_2^2 \quad \text{subject to} \quad \|x\|_0 \le s$$

  by hard thresholding iterations;
- an improved IRLS-ℓ_p algorithm [38] that solves the unconstrained ℓ_p problem with $0<p<1$,

$$\min_{x\in\mathbb{R}^n} \frac{1}{2}\|Ax-b\|_2^2 + \lambda\|x\|_p^p; \tag{5.1}$$

- reweighted ℓ_1 [10] which is at heart a nonconvex CS solver based on the 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 nonconvex in nature.

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

$$A_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\mathbf{0}, I_m/m), \quad i=1,\dots,n,$$

and the random partial DCT matrix

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

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 a more ill-conditioned sensing matrix of significantly higher coherence. Specifically, a randomly oversampled partial DCT matrix A is defined as

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

where $\xi \in \mathbb{R}^m \sim \mathcal{U}([0,1]^m)$ 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 $\mathrm{supp}(\bar{x})$ so as to satisfy the following condition:

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

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 λ controls data fitting and sparsity of the solution. For the noiseless case, a tiny value should be chosen. When measurements are noisy, a reasonable λ should depend on the noise level. In this case, λ needs to be tuned empirically (typically by a 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 ϵ^{rel} and absolute tolerance ϵ^{abs} are adequately small. Figure 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 $\epsilon^{\mathrm{rel}} \leq 10^{-3}$ and $\epsilon^{\mathrm{abs}} = 10^{-2}\epsilon^{\mathrm{rel}}$. Here $\bar{x}$ is the test signal and x^* is the recovered one by DCA-ℓ_{1-2} from the measurements $b = A\bar{x}$. δ in Algorithm 2 should be well chosen, since sometimes the convergence can be sensitive to its value. Boyd et al. [5] suggest varying δ 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,

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

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

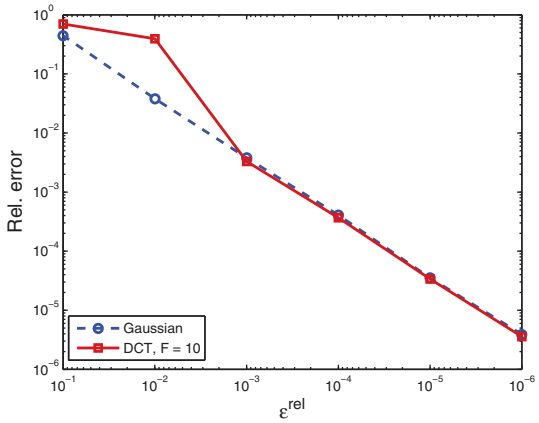

FIG. 2. *Relative error versus relative tolerance using random Gaussian matrix (blue/dash line circle) and oversampled DCT matrix (red/solid line square) with* $F = 10, L = 2F$. *Relative tolerance* $\epsilon^{\mathrm{rel}} = 10^{-1}, 10^{-2}, \ldots, 10^{-6}$ *and absolute tolerance* $\epsilon^{\mathrm{abs}} = 10^{-2}\epsilon^{\mathrm{rel}}$. $A \in \mathbb{R}^{128\times 1024}$, $\|\bar{x}\|_0 = 24$, $\lambda = 10^{-8}$, *and* $\delta = 10^{-7}$. *The relative error is averaged over* 10 *independent trials.*

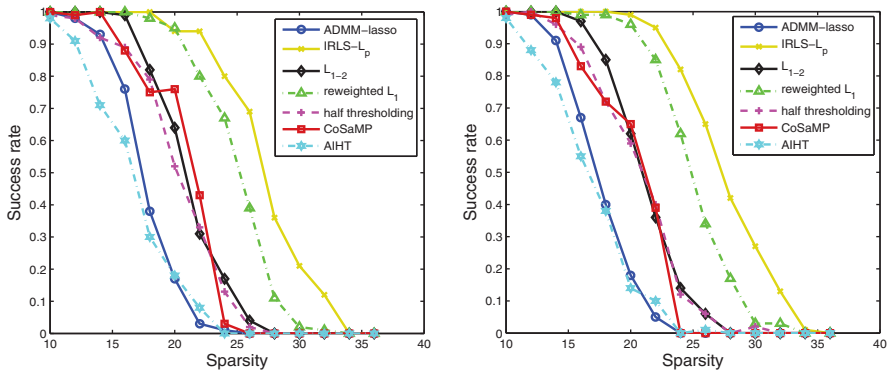

FIG. 3. *Success rates using incoherent sensing matrix.* $m = 64, n = 256, s = 10, 12, \ldots, 36$. *Left: random Gaussian matrix. Right: random partial DCT matrix.*

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 independent and identically distributed. 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 a success if the relative error is

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

We run 100 independent realizations and record the corresponding success rates at various sparsity levels.

We chose $\epsilon^{\text{abs}} = 10^{-7}$ and $\epsilon^{\text{rel}} = 10^{-5}$ for DCA-ℓ_{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 $\lambda = 10^{-6}$, $\beta = 1$, $\rho = 10^{-5}$, $\epsilon^{\text{abs}} = 10^{-7}$, $\epsilon^{\text{rel}} = 10^{-5}$, 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-ℓ_p, $p = 0.5$, `maxiter` $= 1000$, `tol`$= 10^{-8}$. For reweighted ℓ_1, the smoothing parameter ε was adaptively updated as introduced in [10], and the outer stopping criterion adopted was the same as that of the DCA-ℓ_{1-2}. We solved its weighted ℓ_1 minimization subproblems using the more efficient YALL1 solver (available at http://yall1.blogs.rice.edu/) instead of the default ℓ_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.

Figure 3 depicts the success rates of the proposed methods with $m = 64$ and $n = 256$. For both the Gaussian matrix and the partial DCT matrix, IRLS-ℓ_p with $p = 0.5$ has the best performance, followed by reweighted ℓ_1. DCA-ℓ_{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 Figure 4 for $F = 10$ (left) and 20 (right). Note that in this example, the task of nonconvex CS has become

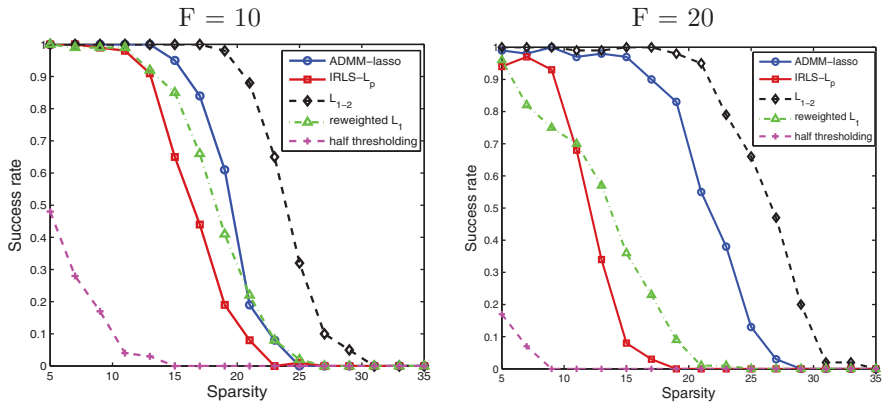

FIG. 4. *Success rates using randomly oversampled partial DCT matrices with* $m = 100, n = 2000, s = 5, 7, 9, \ldots, 35$*, minimum separation* $L = 2F$*.*

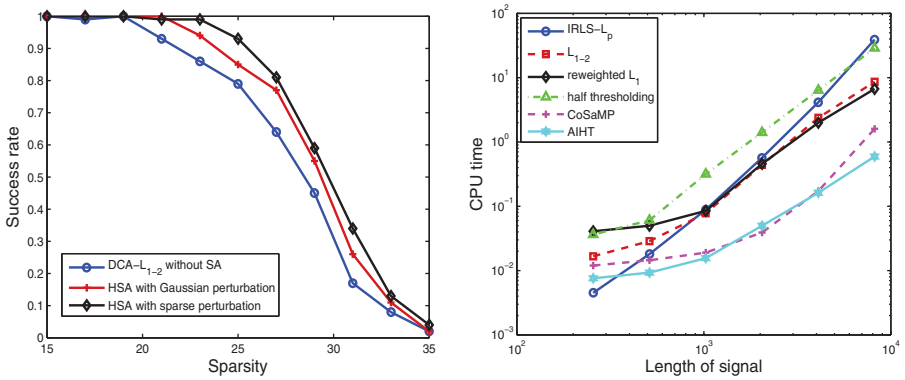

FIG. 5. *Left: Comparison of success rates of HSA algorithms using randomly oversampled partial DCT matrices with* $m = 100, n = 1500, s = 15, 17, \ldots, 35$*, minimum separation* $L = 2F$*. Right: Comparison of CPU time using random Gaussian matrix with* $n = 2^8, 2^9, \ldots, 2^{13}, m = n/4, s =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

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 a low sparsity level, they do not work for a matrix of large coherence at all (in terms of *exact reconstruction*). In this example, the DCA-ℓ_{1-2} is the best and provides robust performance regardless of large coherence of A. In contrast, the other nonconvex solvers clearly encounter the trapping of local minima and perform worse than the convex ADMM-lasso. Moreover, by comparing the two plots in Figure 4, one can tell that their reconstruction qualities suffer a decline as A becomes more and more coherent.

The left plot of Figure 5 shows the success rates for DCA-ℓ_{1-2} with and without aid of HSA methods. For each HSA method, we apply at most 100 iterations. The matrix size is 100×1500, $F = 20$, 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-ℓ_{1-2}. However, HSA with sparse perturbations has the best performance. On the other hand, though the limit point of DCA-ℓ_{1-2} is not known theoretically to be a global minimum, in practice it is quite close. This can be seen from Figure 5, where the additional improvement from the HSA is at most about 15% in the intermediate sparsity regime.

Comparison of time efficiency under Gaussian measurements. The comparison of CPU time using random Gaussian sensing matrix and nonconvex CS solvers is presented in the right plot of Figure 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}, except for 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 performance 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-ℓ_{1-2} is faster than the other regularization methods like IRLS-ℓ_p and half thresholding. This experiment was carried out on a laptop with 16 GB RAM and a 2.40-GHz Intel Core i7 CPU.

5.3. Robust recovery in presence of noise. In this example, we show robustness of DCA-ℓ_{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 = awgn(Ax,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-ℓ_{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 the amount of noise, we test DCA-ℓ_{1-2}, ADMM-lasso, half thresholding, CoSaMP, and AIHT on both the Gaussian matrix and the ill-conditioned oversampled DCT. For Gaussian measurements, we chose $n = 1024$, $m = 256$, and $s = 48$. For oversampled DCT, $n = 2000$, $m = 100$, $s = 15$, $F = 10$, and the minimum separation $L = 2F$. At each noise level, we run 50 times and record the average SNR of reconstruction (in dB).

Table 1 shows the results under Gaussian measurements. Choosing an appropriate value of λ is necessary for both DCA-ℓ_{1-2} and ADMM-lasso to function well, and for this we employ a "trial and error" strategy. CoSaMP and AIHT do not need such a parameter, whereas half thresholding embraces a self-adjusting λ 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-ℓ_{1-2} 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-ℓ_{1-2} and ADMM-lasso perform much better, still doing the job under a moderate amount of noise. Nevertheless, due to the large coherence of A, their performance dropped compared to that in the Gaussian case. In either case, DCA-ℓ_{1-2} consistently beats ADMM-lasso.

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.

`snr` (dB)	DCA-ℓ_{1-2}	ADMM-lasso	Half thresholding	CoSaMP	AIHT
50	−38.8116	−37.1611	−48.2594	−43.4206	**−48.3623**
40	−29.2398	−27.6103	**−37.3863**	−32.8737	−36.9471
30	−19.5802	−18.3454	**−26.0555**	−21.2330	−25.1658
20	−11.0752	−9.8646	**−11.5311**	−8.1797	−10.6132
10	**−3.6700**	−3.1970	−1.4126	1.1522	−1.4048

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.

`snr` (dB)	DCA-ℓ_{1-2}	ADMM-lasso	Half thresholding	AIHT
50	**−35.2119**	−25.9895	−3.8896	−3.8393
35	**−17.0934**	−12.2916	−4.2793	−3.7375
20	**−3.1806**	−3.0157	−2.6428	−0.8141

5.4. MRI reconstruction. We present a two-dimensional example of reconstructing MRI from a limited number of projections. It was first introduced in [6] to demonstrate the success of CS. The signal/image is a Shepp–Logan phantom of size 256×256, as shown in Figure 6. In this case, it is the gradient of the signal that is sparse, and therefore the work [6] is to minimize the ℓ_1 norm of the gradient, or the so-called total variation (TV),

$$\min \|\nabla u\|_1 \quad \text{subject to} \quad R\mathcal{F}u = f, \tag{5.2}$$

where $\mathcal{F}$ 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 ℓ_{1-2} on the gradient is expressed as

$$\min |\partial_x u| + |\partial_y u| - \sqrt{|\partial_x u|^2 + |\partial_y u|^2} \quad \text{subject to} \quad R\mathcal{F}u = f. \tag{5.3}$$

It is anisotropic TV. We apply the technique of DCA by linearizing the ℓ_2 norm of the gradient,

$$\begin{aligned} u^{k+1} = \arg\min_{u,d_x,d_y} \; & |d_x| + |d_y| - \frac{(d_x, d_y)^{\mathrm{T}}(\partial_x u^k, \partial_y u^k)}{\sqrt{|\partial_x u^k|^2 + |\partial_y u^k|^2}} \\ & + \frac{\mu}{2}\|R\mathcal{F}u - f\|_2^2 + \frac{\lambda}{2}\|d_x - \partial_x u\|_2^2 + \frac{\lambda}{2}\|d_y - \partial_y u\|_2^2. \end{aligned} \tag{5.4}$$

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.

Figure 6 shows the exact recovery of 8 projections using the proposed method. We also compare with the classical filtered back projection (FBP) and ℓ_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.

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

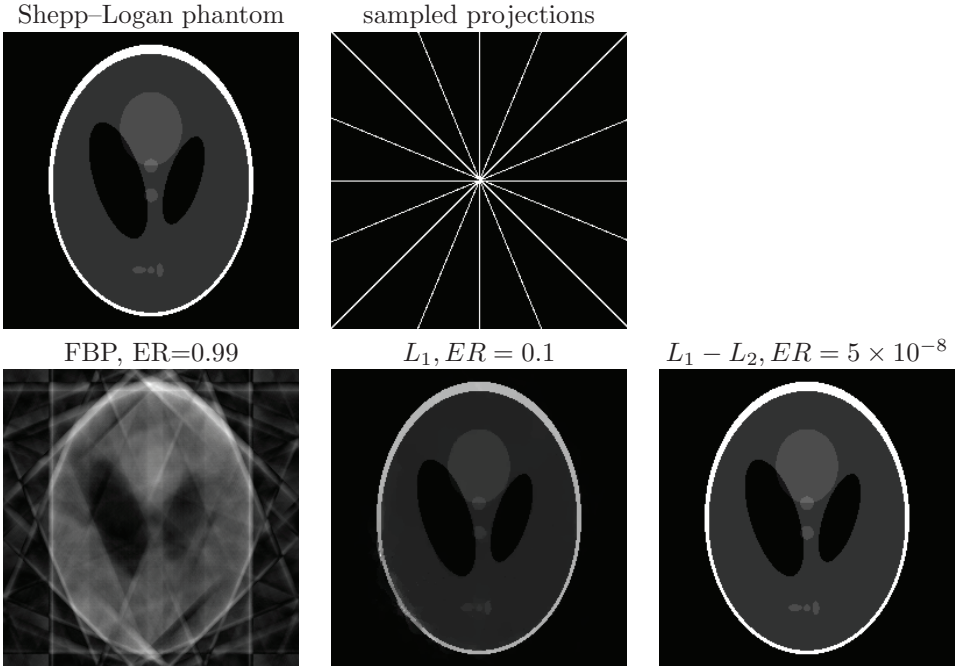

FIG. 6. *MRI reconstruction results. It is demonstrated that* 8 *projections are enough to have exact recovery using* $\ell_1 - \ell_2$. *The relative errors are provided for each method.*

6. Concluding remarks. We have studied CS problems under a nonconvex Lipschitz continuous metric ℓ_{1-2} 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 an 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 an SA procedure. If the sensing matrix is well-conditioned, computational examples suggest that IRLS-ℓ_p ($p = 1/2$) is the best in terms of the success rates of sparse signal recovery. For a highly coherent matrix, DCA-ℓ_{1-2} becomes the best. In either regime, DCA-ℓ_{1-2} is always better than ADMM-lasso. The MRI phantom image recovery test also indicates that ℓ_{1-2} outperforms $\ell_{1/2}$ and ℓ_1.

In future work, we plan to investigate further why ℓ_{1-2} improves on ℓ_1 in a robust manner.

Appendix. Proof of Lemma 2.1.

Proof of Lemma 2.1.

(a) The upper bound is immediate from the 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

$$\|x\|_2 \leq \sum_{i=1}^{t} |x_i| + (\sqrt{n} - t)|x_{t+1}|. \tag{6.1}$$

To see this, we square both sides,

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

or equivalently,

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

Then (6.1) holds because

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

and

$$\begin{aligned}
&\sum_{i=1}^{t} \sum_{\substack{j=1 \\ j \neq i}}^{t} |x_i||x_j| + 2(\sqrt{n}-t)|x_{t+1}| \sum_{i=1}^{t} |x_i| + (\sqrt{n}-t)^2 |x_{t+1}|^2 \\
&\quad \geq \sum_{i=1}^{t} \sum_{\substack{j=1 \\ j \neq i}}^{t} |x_{t+1}|^2 + 2(\sqrt{n}-t)|x_{t+1}| \sum_{i=1}^{t} |x_{t+1}| + (\sqrt{n}-t)^2 |x_{t+1}|^2 \\
&\quad = ((t^2 - t) + 2(\sqrt{n}t - t^2) + (\sqrt{n}-t)^2)|x_{t+1}|^2 \\
&\quad = (n-t)|x_{t+1}|^2.
\end{aligned}$$

It follows from (6.1) that

$$\begin{aligned}\|x\|_1 - \|x\|_2 &\geq \|x\|_1 - \left(\sum_{i=1}^{t} |x_i| + (\sqrt{n}-t)|x_{t+1}|\right) \\ &= (t+1-\sqrt{n})|x_{t+1}| + \sum_{i=t+2}^{n} |x_i| \\ &\geq (t+1-\sqrt{n})|x_n| + \sum_{i=t+2}^{N} |x_n| = (n-\sqrt{n})|x_n| \\ &= (n-\sqrt{n}) \min_i |x_i|.\end{aligned}$$

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

(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 the IRLS-ℓ_p algorithm published in [38]. We thank Dr. Ernie Esser for helpful conversations. We also thank anonymous referees for their constructive comments.

REFERENCES

[1] A. Bandeira, E. Dobriban, D. Mixon, and W. Sawin, *Certifying the restricted isometry property is hard*, IEEE Trans. Inform. Theory, 59 (2013), pp. 3448–3450.

[2] A. Beck and M. Teboulle, *A fast iterative shrinkage-thresholding algorithm for linear inverse problems*, SIAM J. Imaging Sci., 2 (2009), pp. 183–202.

[3] T. Blumensath, *Accelerated Iterative hard thresholding*, Signal Process., 92 (2012), pp. 752–756

[4] T. Blumensath and M. Davies, *Iterative hard thresholding for compressed sensing*, Appl. Comput. Harmon. Anal., 27 (2009), pp. 265–274.

[5] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, *Distributed optimization and statistical learning via the alternating direction method of multipliers*, Found. Trends Mach. Learning, 3 (2011), pp. 1–122.

[6] E. Candès, J. Romberg, and T. Tao, *Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information*, IEEE Trans. Inform. Theory, 52 (2006), pp. 489–509.

[7] E. Candès, J. Romberg, and T. Tao, *Stable signal recovery from incomplete and inaccurate measurements*, Comm. Pure Appl. Math., 59 (2006), pp. 1207–1223.

[8] E. Candès and T. Tao, *Decoding by linear programming*, IEEE Trans. Inform. Theory, 51 (2005).

[9] E. Candès, M. Rudelson, T. Tao, and R. Vershynin, *Error correction via linear programming*, in Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science, 2005.

[10] E. Candès, M. Wakin, and S. Boyd, *Enhancing sparsity by reweighted ℓ_1 minimization*, J. Fourier Anal. Appl., 14 (2008), pp. 877–905.

[11] P. Carnevali, L. Coletti, and S. Patarnello, *Image processing by simulated annealing*, IBM J. Res. Development, 29 (1985), pp. 569–579.

[12] R. Chartrand, *Exact reconstruction of sparse signals via nonconvex minimization*, IEEE Signal Process. Lett., 14 (2007), pp. 707–710.

[13] R. Chartrand and V. Staneva *Restricted isometry properties and nonconvex compressive sensing*, Inverse Problems, 24 (2008), pp. 1–14.
[14] R. Chartrand and W. Yin, *Iteratively reweighted algorithms for compressive sensing*, in Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, 2008, pp. 3869–3872.
[15] S. Chen, D. Donoho, and M. A. Saunders, *Atomic decomposition by basis pursuit*, SIAM J. Sci. Comput., 20 (1998), pp. 33–61.
[16] X. Chen, F. Xu, and Y. Ye, *Lower bound theory of nonzero entries in solutions of ℓ_2-ℓ_p minimization*, SIAM J. Sci. Comput., 32 (2010), pp. 2832–2852.
[17] X. Chen and W. Zhou, *Convergence of the reweighted ℓ_1 minimization algorithm for ℓ_2-ℓ_p minimization*, Comput. Optim. Appl., 59 (2014), pp. 47–61.
[18] A. Cohen, W. Dahmen, and R. Devore, *Compressed sensing and best k-term approximation*, J. Amer. Math. Soc., 22 (2009), pp. 221–231.
[19] I. Daubechies, R. DeVore, M. Fornasier, and C. Güntük, *Iteratively reweighted least squares minimization for sparse recovery*, Commun. Pure Appl. Math., 63 (2010), pp. 1–38.
[20] D. Donoho, *Superresolution via sparsity constraints*, SIAM J. Math. Anal., 23 (1992), pp. 1309–1331.
[21] D. Donoho and X. Huo, *Uncertainty principles and ideal atomic decomposition*, IEEE Trans. Inform. Theory, 47 (2001), pp. 2845–2862.
[22] D. Donoho and M. Elad, *Optimally sparse representation in general (nonorthogonal) dictionaries via l^1 minimization*, Proc. Natl. Acad. Sci. USA, 100 (2003), pp. 2197–2202.
[23] D. Donoho, *Compressed sensing*, IEEE Trans. Inform. Theory, 52 (2006), pp. 1289–1306.
[24] J. Fan and R. Li, *Variable selection via nonconcave penalized likelihood and its oracle properties*, J. Amer. Statist. Assoc., 96 (2001), pp. 1348–1360.
[25] A. Fannjiang and W. Liao, *Coherence pattern-guided compressive sensing with unresolved grids*, SIAM J. Imaging Sci., 5 (2012), pp. 179–202.
[26] S. Foucart and M. Lai, *Sparsest solutions of underdetermined linear systems via ℓ_q-minimization for $0 < q \leq 1$*, Appl. Comput. Harmon. Anal., 26 (2009), pp. 395–407.
[27] E. Esser, *Applications of Lagrangian-Based Alternating Direction Methods and Connections to Split Bregman*, CAM-report 09-31, UCLA, Los Angeles, CA, 2009.
[28] E. Esser, Y. Lou, and J. Xin, *A method for finding structured sparse solutions to non-negative least squares problems with applications*, SIAM J. Imaging Sci., 6 (2013), pp. 2010–2046.
[29] D. Gabay and B. Mercier, *A dual algorithm for the solution of nonlinear variational problems via finite element approximations*, Comput. Math. Appl., 2 (1976), pp. 17–40.
[30] S. Geman and D. Geman, *Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images*, IEEE Trans. Pattern Anal. Mach. Intelligence, 6 (1984), pp. 721–741.
[31] B. Gidas, *Nonstationary Markov chains and convergence of the annealing algorithm*, J. Statist. Phys., 39 (1985), pp. 73–131.
[32] R. Glowinski and A. Marrocco, *Sur l'approximation, par éléents finis d'ordre un, et la résolution, par péalisation-dualité, d'une classe de problèmes de dirichelet non linéires*, Rev. Française Automat. Inform. Recherche Opér., (1975), pp. 41–76.
[33] T. Goldstein and S. Osher, *The split Bregman method for ℓ_1-regularized problems*, SIAM J. Imaging Sci., 2 (2009), pp. 323–343.
[34] E. Hale, W. Yin, and Y. Zhang, *Fixed-point continuation for ℓ_1-minimization: Methodology and convergence*, SIAM J. Optim., 19 (2008), pp. 1107–1130.
[35] B. Hajek, *Cooling schedules for optimal annealing*, Math. Oper. Res., 13 (1988), pp. 311–329.
[36] B. He and X. Yuan, *On the $O(1/n)$ convergence rate of the Douglas-Rachford alternating direction method*, SIAM J. Numer. Anal., 50 (2012), pp. 700–709.
[37] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, *Optimization by simulated annealing*, Science, 220 (1983), pp. 671–680.
[38] M.-J. Lai, Y. Xu, and W. Yin, *Improved Iteratively reweighted least squares for unconstrained smoothed ℓ_q minimization*, SIAM J. Numer. Anal., 51 (2013), pp. 927–957.
[39] S. Liu, X. Shen, and W. Wong, *Computational developments of ψ-learning*, in Proceedings of the SIAM Data Mining Conference, 2005, pp. 1–12.
[40] Y. Lou, P. Yin, Q. He, and J. Xin, *Computing sparse representation on a highly coherent dictionary based on difference of L_1 and L_2*, J. Sci. Comput., 2014, DOI:10.1007/s10915-014-9930-1.
[41] J. Lv and Y. Fan, *Unified approach to model selection and sparse recovery using regularized least squares*, Ann. Statist., 27 (2009), pp. 3498–3528.
[42] Z. Lu and Y. Zhang, *Sparse approximation via penalty decomposition methods*, SIAM J. Optim., 23 (2013), pp. 2448–2478.

[43] D. Needell and J. Tropp *CoSaMP: Iterative signal recovery from incomplete and inaccurate samples*, Appl. Comput. Harmon. Anal., 26 (2009), pp. 301–221.
[44] H. Rauhut, *Compresssive sensing and structred random matrices*, Radon Ser. Comput. Appl. Math., 9 (2010), pp. 1–92.
[45] P. D. Tao and L. T. H. An, *Convex analysis approach to dc programming: Theory, algorithms and applications*, Acta Math. Vietnam., 22 (1997), pp. 289–355.
[46] P. D. Tao and L. T. H. An, *A D.C. optimization algorithm for solving the trust-region subproblem*, SIAM J. Optim., 8 (1988), pp. 476–505.
[47] R. Tibshirani, *Regression shrinkage and selection via the Lasso*, J. Roy. Statist. Soc. Ser. B., 58 (1996), pp. 267–288.
[48] J. Tropp and A. Gilbert, *Signal recovery from random measurements via orthogonal matching pursuit*, IEEE Trans. Inform. Theory, 53 (2007), pp. 4655–4666.
[49] Z. Wen, W. Yin, D. Goldfarb, and Y. Zhang, *A fast algorithm for sparse reconstruction based on shrinkage, subspace optimization, and continuation*, SIAM J. Sci. Comput., 32 (2010), pp. 1832–1857.
[50] S. Wright, R. Nowak, and M. Figueiredo, *Sparse reconstruction by separable approximation*, IEEE Trans. Signal Process., 57 (2009), pp. 2479–2493.
[51] F. Xu and S. Wang, *A hybrid simulated annealing thresholding algorithm for compressed sensing*, Signal Process., 93 (2013), pp. 1577–1585.
[52] Z. Xu, X. Chang, F. Xu, and H. Zhang, *$L_{1/2}$ regularization: A thresholding representation theory and a fast solver*, IEEE Trans. Neural Networks Learning Systems, 23 (2012), pp. 1013–1027.
[53] Z. Xu, H. Guo, Y. Wang, and H. Zhang, *The representative of $L_{1/2}$ regularization among L_q $(0 < q \leq 1)$ regularizations: An experimental study based on phase diagram*, Acta Automat. Sinica, 38 (2012), pp. 1225–1228.
[54] J. Yang and Y. Zhang, *Alternating direction algorithms for L_1-problems in compressive sensing*, SIAM J. Sci. Comput., 33 (2011), pp. 250–278.
[55] P. Yin, E. Esser, and J. Xin, *Ratio and difference of L_1 and L_2 norms and sparse representation with coherent dictionaries*, Commun. Inform. Systems, 14 (2014), pp. 87–109.
[56] W. Yin and S. Osher, *Error forgetting of Bregman iteration*, J. Sci. Comput., 54 (2013), pp. 684–695.
[57] W. Yin, S. Osher, D. Goldfarb, and J. Darbon, *Bregman iterative algorithms for l_1 minimization with applications to compressed sensing*, SIAM J. Imaging Sci., 1 (2008), pp. 143–168.
[58] S. Yun and K.-C. Toh, *A coordinate gradient descent method for ℓ_1-regularized convex minimization*, Comput. Optim. Appl., 48 (2011), pp. 273–307.
[59] J. Zeng, S. Lin, Y. Wang, and Z. Xu, *$L_{1/2}$ regularization: Convergence of iterative half thresholding algorithm*, IEEE Trans. Image Process., 62 (2014), pp. 2317–2329.
[60] Y. Zhang, *Theory of compressive sensing via ℓ_1-minimization: A non-RIP analysis and extensions*, J. Oper. Res. Soc. China, 1 (2013), pp. 79–105.
[61] Y. Zhao and D. Li, *Reweighted ℓ_1-minimization for sparse solutions to underdetermined linear systems*, SIAM J. Optim., 22 (2012), pp. 1065–1088.