

GREEDY MINIMIZATION OF L_1 -NORM WITH HIGH EMPIRICAL SUCCESS

Martin Sundin, Saikat Chatterjee and Magnus Jansson

ACCESS Linnaeus Center, School of Electrical Engineering
KTH Royal Institute of Technology, Stockholm, Sweden

masundi@kth.se, sach@kth.se, magnus.jansson@ee.kth.se

ABSTRACT

We develop a greedy algorithm for the basis-pursuit problem. The algorithm is empirically found to provide the same solution as convex optimization based solvers. The method uses only a subset of the optimization variables in each iteration and iterates until an optimality condition is satisfied. In simulations, the algorithm converges faster than standard methods when the number of measurements is small and the number of variables large.

Index Terms— Convex optimization, basis-pursuit, greedy algorithms.

1. INTRODUCTION

Convex optimization based methods are efficient for finding sparse solutions to underdetermined linear systems of equations. This has received much attention [1–4] and has several applications in signal processing [5–9]. One convex method for finding sparse representations is basis-pursuit [10] (or l_1 -minimization) which finds a sparse solution as

$$\hat{\mathbf{x}}_{BP} = \min \|\mathbf{x}\|_1, \quad (1)$$
$$\text{s.t. } \mathbf{y} = \mathbf{A}\mathbf{x}$$

where $\mathbf{A} \in \mathbb{R}^{m \times n}$ is the dictionary, $\mathbf{y} \in \mathbb{R}^m$ is the data/observations and $\mathbf{x} \in \mathbb{R}^n$ is the optimization variable. Typically $m \ll n$.

1.1. Prior work

Many efficient methods have been constructed to solve (1) and related noisy problems [11–16]. Methods have also been developed for large scale problems [17–20] using specialized and distributed optimization methods. Another approach to increasing the speed is to start with the full set of variables and then gradually eliminate variables based on *screening principles* [21, 22].

While solvers for (1) are efficient, they are often slower than greedy search methods [23–26] which in a greedy manner solve the problem

$$\min \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2, \quad \text{s.t. } \|\mathbf{x}\|_0 \leq K,$$

where the sparsity $\|\mathbf{x}\|_0$ is assumed to be known.

To develop fast methods for efficiently finding sparse approximations, it is desirable to combine the effectiveness of convex optimization with the speed of greedy algorithms.

This work was partially supported by the Swedish Research Council under contract 621-2011-5847.

1.2. Our contribution

By using that the basis-pursuit solution is at most m -sparse [4] we construct an iterative algorithm, *GLI*, which performs basis-pursuit (1) in a greedy fashion using only m variables in each iteration. The algorithm replaces vectors in an active set for vectors which lower the l_1 -norm of the solution. This is done until no replacement lowers the l_1 -norm or an optimality condition is met. If the optimality condition is satisfied, then the algorithm recovers the basis-pursuit solution. Even though we were not able to prove convergence, the algorithm has been found to empirically converge to the basis-pursuit solution in all problem realizations.

1.3. Notation

We denote the active set by $I = \{i_1, i_2, \dots, i_m\} \subset [n] = \{1, 2, \dots, n\}$ and use $|J|$ to denote the number of elements in a set J . We set $\mathbf{A}_I = [\mathbf{a}_{i_1}, \mathbf{a}_{i_2}, \dots, \mathbf{a}_{i_m}]$, $\mathbf{A} = \mathbf{A}_{[n]}$ and use \mathbf{x}_I to denote the vector consisting of the elements of \mathbf{x} with indices in I . The support set of a vector \mathbf{x} is denoted by $\text{supp}(\mathbf{x}) = \{i | x_i \neq 0\}$ and the l_0 -norm is the size of the support set, i.e. $\|\mathbf{x}\|_0 = |\text{supp}(\mathbf{x})|$. We use \circ to denote the Hadamard (elementwise) product of two vectors and assume that \mathbf{A} has full rank and column vectors of unit l_2 -norm.

2. THE GEOMETRY OF BASIS PURSUIT

Given an at most m -sparse vector \mathbf{x} such that $\text{supp}(\mathbf{x}) = J$, $|J| \leq m$ and $\mathbf{A}_J \mathbf{x}_J = \mathbf{y}$, we can always enlarge the set J to a set I such that $J \subset I$, $|I| = m$ and \mathbf{A}_I is full rank (since \mathbf{A} is full rank). Thus $\mathbf{x}_I = \mathbf{A}_I^{-1} \mathbf{y}$. Since the basis-pursuit solution has at most m non-zero components, this means that (1) is equivalent to

$$I = \arg \min_{|I|=m} \|\mathbf{A}_I^{-1} \mathbf{y}\|_1, \text{ s.t. } \mathbf{A}_I \text{ is invertible} \quad (2)$$
$$(\hat{\mathbf{x}}_{BP})_I = \mathbf{A}_I^{-1} \mathbf{y}, \quad (\hat{\mathbf{x}}_{BP})_{I^c} = 0.$$

We find that (2) is an exhaustive search over all subsets $I \subset [n]$ of size $|I| = m$. However, due to the geometry of the basis-pursuit problem, many subsets can be eliminated from the search.

Let $\mathcal{C}(I, \mathbf{s})$ denote the *convex cone*

$$\mathcal{C}(I, \mathbf{s}) = \{\mathbf{A}_I \mathbf{r}, \mathbf{r} \in \mathbb{R}^{|I|}, r_i s_i \geq 0 \forall i \in I\}$$

of \mathbf{A} , where $|I| \leq m$ and $\mathbf{s} = (s_1, s_2, \dots, s_{|I|})$ with $s_i = \pm 1$ for $1 \leq i \leq m$. We say that a cone $\mathcal{C}(I, \mathbf{s})$ is *minimal* (in \mathbf{A}) if there is no $j \notin I$ such that $\mathbf{a}_j = \mathbf{A}_I \mathbf{x}'_I$ and $\text{sign}(\mathbf{x}'_I) = \pm \mathbf{s}$. An important property of basis-pursuit is that the solution is always contained in a minimal cone. We formulate this as a proposition.

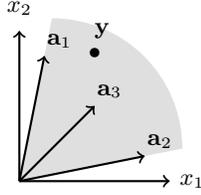


Fig. 1. The cone $\mathcal{C}(\{1, 2\}, (1, 1)^\top)$ is not minimal since it contains \mathbf{a}_3 . Basis-pursuit gives a solution with support set $\{1, 3\}$ since it is the only minimal cone containing \mathbf{y} .

Proposition 1. The support set of $\hat{\mathbf{x}}_{BP}$ is contained in a minimal cone $\mathcal{C}(I, \mathbf{s})$, where $\text{supp}(\hat{\mathbf{x}}_{BP}) = J \subset I$ and $\text{sign}(\hat{x}_{j_k}) = s_k$ for $k = 1, 2, \dots, |J|$.

The proof is given in the Appendix. An illustration in two dimensions is given in Figure 1. In two dimensions the basis-pursuit solution is given by the unique minimal cone. Note that the minimal cone containing \mathbf{y} need not be unique in general. When \mathbf{y} has a sparse representation, then \mathbf{y} lies on the boundary of several minimal cones. By proposition 1 it is sufficient to search over all minimal cones in (2).

If basis-pursuit recovers an m -sparse vector \mathbf{x} from measurements $\mathbf{A}\mathbf{x}$, then basis-pursuit recovers any other vector \mathbf{x}' with $\text{supp}(\mathbf{x}') = \text{supp}(\mathbf{x})$ and $\text{sign}(\mathbf{x}'_j) = \text{sign}(\mathbf{x}_j)$ [27], i.e. (1) and (2) gives a solution with $\text{supp}(\hat{\mathbf{x}}_{BP}) = I$ and $\text{sign}((\hat{\mathbf{x}}_{BP})_I) = \mathbf{s}$ for all \mathbf{y} in the interior of $\mathcal{C}(I, \mathbf{s})$.

3. GREEDY L_1 -MINIMIZATION

Since basis-pursuit gives a solution with the same support set and sign-pattern for all measurements inside the cone of the basis-pursuit solution, we cannot interchange a vector in the support set for a vector in the complement to lower the l_1 -norm. This can be verified without explicitly computing the new solutions, as the following theorem explains.

Theorem 1. Let \mathbf{x} be an m -sparse solution to $\mathbf{y} = \mathbf{A}\mathbf{x}$, $I = \text{supp}(\mathbf{x})$ and $\mathbf{s} = \text{sign}(\mathbf{x}_I)$. Then the l_1 -norm of the solution cannot be lowered by replacing the k 'th column vector of \mathbf{A}_I for a vector \mathbf{a}_j ($j \notin I$) if

$$1 \geq \text{sign}(s_k z_k) (\mathbf{s}^\top \mathbf{z}), \quad (3)$$

where $\mathbf{z} = \mathbf{A}_I^{-1} \mathbf{a}_j$. Furthermore, if (3) is satisfied with strict inequality for all k and \mathbf{a}_j ($j \in I^c$), then $|\mathbf{s}^\top \mathbf{z}| < 1$ for all \mathbf{a}_j ($j \in I^c$) and $\mathbf{x} = \hat{\mathbf{x}}_{BP}$.

Noting that we can write

$$\mathbf{s}^\top \mathbf{z} = \mathbf{s}^\top \mathbf{A}_I^{-1} \mathbf{a}_j = \mathbf{h}^\top \mathbf{a}_j$$

where $\mathbf{h} = (\mathbf{A}_I^\top)^{-1} \mathbf{s}$, we find that if $|\mathbf{h}^\top \mathbf{a}_j| \leq 1$, then no column vector in \mathbf{A}_I can be replaced by \mathbf{a}_j to lower the l_1 -norm. By the theorem, if $|\mathbf{h}^\top \mathbf{a}_j| < 1$ for all $j \in I^c$, then $\mathbf{x} = \hat{\mathbf{x}}_{BP}$.

3.1. The GL1 algorithm

Using (3) we construct the *greedy algorithm for l_1 -minimization*, GL1. The algorithm starts with an initial *active set* I and use I to

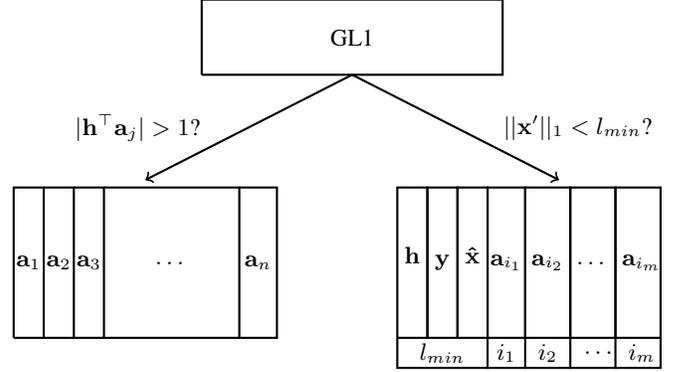


Fig. 2. Illustration of the algorithm. The algorithm reads from the complement set and computes using the active set.

construct a *certificate* \mathbf{h} . The algorithm searches for candidate vectors in I^c that satisfy $|\mathbf{h}^\top \mathbf{a}_j| > 1$. The candidate vectors are tested (for all k such that (3) is violated) if replacing \mathbf{a}_{i_k} by the candidate vector lowers the l_1 -norm, i.e. if

$$\|\mathbf{x}'\|_1 = \|\mathbf{A}_{I'}^{-1} \mathbf{A}_I \hat{\mathbf{x}}\|_1 = \left\| \hat{\mathbf{x}} + (\mathbf{e}_k - \mathbf{z}) \frac{\hat{x}_k}{z_k} \right\|_1 < \|\hat{\mathbf{x}}\|_1, \quad (4)$$

where \mathbf{e}_k denotes the k 'th basis vector in the coordinate basis. This is done until (4) is fulfilled.

The algorithm can get stuck in a local optima (of the algorithm) if it encounters a sparse solution. If this happens, we slightly perturb \mathbf{y} to ensure that it lies in the interior of a cone. When the algorithm terminates, \mathbf{y} is restored and the estimate is computed. The GL1 algorithm can be summarized as follows.

1. Input: $\mathbf{y}, \mathbf{A}, I$.
2. Initialization: $\mathbf{y}' = \mathbf{y}$
3. Repeat:
 4. $\hat{\mathbf{x}} = \mathbf{A}_I^{-1} \mathbf{y}'$, $\mathbf{s} = \text{sign}(\hat{\mathbf{x}})$, $l_{min} = \|\hat{\mathbf{x}}\|_1$, $\mathbf{h} = (\mathbf{A}_I^\top)^{-1} \mathbf{s}$.
 5. If $\|\hat{\mathbf{x}}\|_0 < m$: perturb $\mathbf{y}' = \mathbf{y} + \Delta \cdot \mathbf{A}_I \mathbf{1}$, go to 4.
 6. For all $j \in I^c$ such that $|\mathbf{h}^\top \mathbf{a}_j| > 1$:
 - (a) Compute $\mathbf{z} = \mathbf{A}_I^{-1} \mathbf{a}_j$ and $\mathbf{t} = \text{sign}(\mathbf{s} \circ \mathbf{z}) (\mathbf{s}^\top \mathbf{z})$.
 - (b) For all k such that $t_k > 1$ and $|z_k| > \epsilon$:
 - i. If $\|\hat{\mathbf{x}} + (\mathbf{e}_i - \mathbf{z}) \frac{\hat{x}_i}{z_i}\|_1 < l_{min}$:
 - A. $I \rightarrow (I \cup \{j\}) \setminus \{i_k\}$,
 - B. $I^c \rightarrow (I^c \cup \{i_k\}) \setminus \{j\}$, go to 4.
 7. If $\|\hat{\mathbf{x}}\|_1 = l_{min}$: break.
8. Output: $I, \hat{\mathbf{x}}_I = \mathbf{A}_I^{-1} \mathbf{y}$.

Notes on the algorithm:

- $\Delta, \epsilon > 0$ are small constants (e.g. 10^{-5}).
- Usually, $\hat{\mathbf{x}}$ is not exactly sparse due to numerical errors. Then it is better to perturb as

$$\mathbf{y}' = \mathbf{y} + \Delta \cdot \mathbf{A} \mathbf{s},$$

where $\mathbf{s} = \text{sign}(\hat{\mathbf{x}})$.

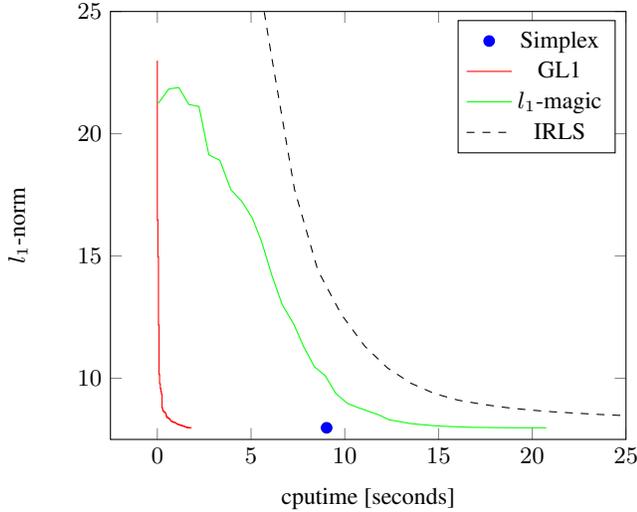


Fig. 3. l_1 -norm vs. cputime for $n = 8000$ and $m = 50$.

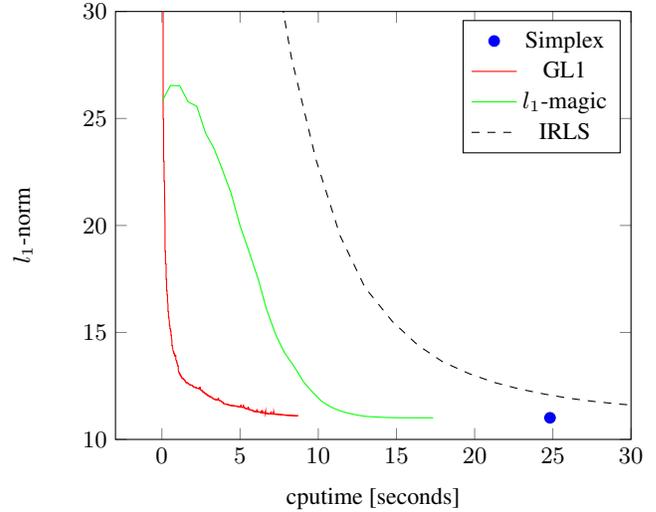


Fig. 4. l_1 -norm vs. cputime for $n = 8000$ and $m = 100$.

- In implementation, matrix inverses are replaced by solving the set of linear equations, to increase numerical accuracy and speed.
- The runtime can be decreased by selecting the initial set I to be the m vectors which have largest inner products $|\mathbf{a}_i^\top \mathbf{y}|$.
- When determining if the candidate vectors can lower the l_1 -norm, it is beneficial to start with the vector with largest inner product $|\mathbf{h}^\top \mathbf{a}_j|$ and then proceed to the one with next largest inner product.

We note that the algorithm consists of two parts, determining if $|\mathbf{h}^\top \mathbf{a}_j| > 1$ and if replacing a column vector with \mathbf{a}_j lowers the l_1 -norm, see Figure 2. Both of these operations can be parallelized, making the algorithm suitable for large scale problems.

4. NUMERICAL COMPARISON

To evaluate the performance of the algorithm, we compared it with three other solvers for (1). We used the simplex method [28], l_1 -magic [12] and the Iterative Reweighted Least Squares (IRLS) [29] for l_1 -minimization.

For the Simplex method we reformulate (1) as

$$\begin{aligned} \min \quad & \mathbf{1}^\top (\mathbf{x}_+ + \mathbf{x}_-), \\ \text{s.t.} \quad & \mathbf{y} = \mathbf{A}(\mathbf{x}_+ - \mathbf{x}_-) \\ & \mathbf{x}_+, \mathbf{x}_- \geq \mathbf{0} \end{aligned}$$

where $\mathbf{1} \in \mathbb{R}^n$ is a vector of ones and we used MATLAB's `linprog` to run the simplex algorithm. For l_1 -magic, (1) is reformulated as

$$\begin{aligned} \min \quad & \mathbf{1}^\top \mathbf{t} \\ \text{s.t.} \quad & \mathbf{y} = \mathbf{A}\mathbf{x} \\ & -t_i \leq x_i \leq t_i, \text{ for } i = 1, 2, \dots, n \end{aligned}$$

and solved using a primal-dual interior point method [12]. The IRLS algorithm approximates the l_1 -norm by a weighted l_2 -norm which is updated iteratively [29].

In the simulation we used $n = 8000$ and varied m . The measurements were generated by drawing the elements of \mathbf{A} from

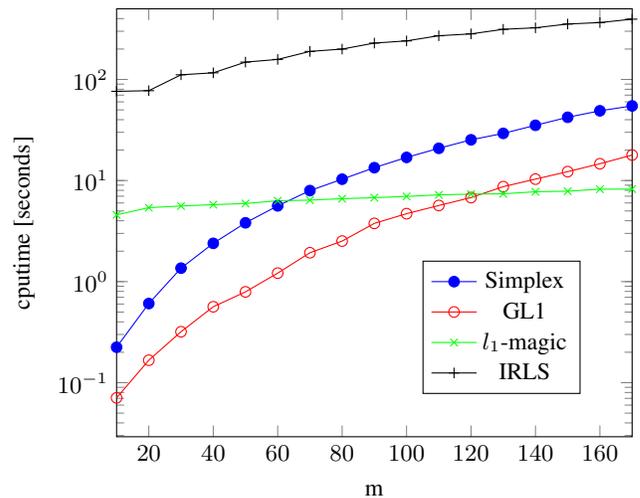


Fig. 5. Mean cputime vs. m for $n = 8000$.

$\mathcal{N}(0, 1)$ and normalizing the column vectors, the vector \mathbf{x} was generated as a $\lceil 0.25m \rceil$ -sparse vector with its non-zero elements drawn from $\mathcal{N}(0, 1)$. Finally we calculated $\mathbf{y} = \mathbf{A}\mathbf{x}$. Because \mathbf{A} is Gaussian, $\hat{\mathbf{x}}_{BP}$ in (1) is unique (with probability 1).

Because the algorithms have different complexity per iteration, we measure the total cputime and the l_1 -norm rather than the number of iterations. We were not able to access the intermediate times and l_1 -norms of `linprog` and therefore only display the final norm and cputime. We show the convergence in individual realizations for $m = 50$ and $m = 100$ in Figure 3 and 4. We find that the simplex method converged faster than l_1 -magic for $m = 50$, but slower for $m = 100$. In both realizations, GL1 was the fastest algorithm. In Figure 5 we show the average cputime for different values of m averaged over 10 realizations of \mathbf{A} and \mathbf{x} . We see that l_1 -magic is slower than the simplex method for $m \leq 80$ and slower than GL1 for $m \leq 140$. The simplex method was about 3 times slower than GL1 for all values of m .

5. CONCLUSION

In this paper we constructed a greedy algorithm for l_1 -minimization which empirically converges to the basis-pursuit solution. The algorithm interchanges column vectors in an active set until an optimality condition is reached or the l_1 -norm no longer decreases. In simulations, the algorithm converged faster than the simplex method, l_1 -magic and IRLS when the number of measurements is small. Questions remain whether the algorithm always converges to the basis-pursuit solution and if it can be extended to other versions of basis-pursuit.

6. APPENDIX

Proof of proposition 1. We need to show that if the cone $\mathcal{C}(I, \mathbf{s})$ contains a vector \mathbf{a}_j , then we can replace a column vector in \mathbf{A}_I for \mathbf{a}_j to lower the l_1 -norm of the solution.

Let $I' = (I \cup \{j\}) \setminus \{k\}$ and set

$$\mathbf{y} = \sum_{i \in I} x_i \mathbf{a}_i, \quad (5)$$

$$\mathbf{a}_j = \sum_{i \in I} z_i \mathbf{a}_i, \quad (6)$$

$$\mathbf{y} = x'_j \mathbf{a}_j + \sum_{i \in I, i \neq k} x'_i \mathbf{a}_i. \quad (7)$$

Without loss of generality we can assume that $x_i, z_i, x'_i \geq 0$ for all $i \in I \cup \{j\}$. By inserting (6) into (7) we find that

$$x'_j \mathbf{a}_j + \sum_{i \in I, i \neq k} x'_i \mathbf{a}_i = x'_j z_k \mathbf{a}_k + \sum_{i \in I, i \neq k} (x'_i + x'_j z_i) \mathbf{a}_i$$

By comparing to (5) we get that when \mathbf{A}_I and $\mathbf{A}_{I'}$ are full rank, then $x_k = x'_j z_k$ and $x_i = x'_i + x'_j z_i$ for $i \neq k$. Using that

$$1 = \|\mathbf{a}_j\|_2 < \sum_{i \in I} |z_i| \cdot \|\mathbf{a}_i\|_2 = \|\mathbf{z}\|_1,$$

where we have strong inequality because the column vectors in \mathbf{A}_I are not parallell, we find that

$$\begin{aligned} \|\mathbf{x}\|_1 &= \sum_{i \in I} x_i = x'_j z_k + \sum_{i \in I, i \neq k} (x'_i + x'_j z_i) \\ &= x'_j \sum_{i \in I} z_i + \sum_{i \in I, i \neq k} x'_i \\ &= x'_j (\|\mathbf{z}\|_1 - 1) + \|\mathbf{x}'\|_1 > \|\mathbf{x}'\|_1, \end{aligned}$$

provided that $x'_j > 0$. \square

Proof of Theorem 1. Assume that we cannot interchange the k 'th column vector of \mathbf{A}_I for a vector \mathbf{a}_j to get a solution with lower l_1 -norm. Then all measurements in the cone $\mathcal{C}(I, \mathbf{s})$ give basis-pursuit solutions with the same support set and sign-pattern, i.e. for all $\mathbf{w} \in \mathbb{R}^m$, $\mathbf{w} \geq \mathbf{0}$

$$\|\mathbf{w}\|_1 \leq \|\mathbf{A}_{I'}^{-1} \mathbf{A}_I \mathbf{S} \mathbf{w}\|_1, \quad (8)$$

where $\mathbf{S} = \text{diag}(\mathbf{s})$, $I' = (I \cup \{j\}) \setminus \{k\}$ and we assumed that $\mathbf{A}_{I'}$ is invertible.

Replacing \mathbf{a}_{i_k} for \mathbf{a}_j corresponds to making a rank-1 update of \mathbf{A}_I , i.e.

$$\mathbf{A}_{I'} = \mathbf{A}_I + (\mathbf{a}_j - \mathbf{a}_{i_k}) \mathbf{e}_k^\top.$$

Setting $\mathbf{z} = \mathbf{A}_I^{-1} \mathbf{a}_j$, we get that

$$\begin{aligned} \mathbf{A}_{I'}^{-1} \mathbf{A}_I \mathbf{S} \mathbf{w} &= \left(\mathbf{A}_I^{-1} - \frac{\mathbf{A}_I^{-1} (\mathbf{a}_j - \mathbf{a}_{i_k}) \mathbf{e}_k^\top \mathbf{A}_I^{-1}}{1 + \mathbf{e}_k^\top \mathbf{A}_I^{-1} (\mathbf{a}_j - \mathbf{a}_{i_k})} \right) \mathbf{A}_I \mathbf{S} \mathbf{w} \\ &= \mathbf{S} \mathbf{w} - \frac{(\mathbf{A}_I^{-1} \mathbf{a}_j - \mathbf{e}_k) \mathbf{e}_k^\top \mathbf{S} \mathbf{w}}{\mathbf{e}_k^\top \mathbf{A}_I^{-1} \mathbf{a}_j} = \mathbf{S} \mathbf{w} + (\mathbf{e}_k - \mathbf{z}) \frac{s_k w_k}{z_k}. \end{aligned}$$

So if (8) holds, then

$$\begin{aligned} 1 &\leq \min_{\substack{\mathbf{w} \geq \mathbf{0} \\ \mathbf{1}^\top \mathbf{w} = 1}} \left\| \mathbf{S} \mathbf{w} + (\mathbf{e}_k - \mathbf{z}) \frac{s_k w_k}{z_k} \right\|_1 \\ &= \min_{\substack{\mathbf{w} \geq \mathbf{0} \\ \mathbf{1}^\top \mathbf{w} = 1}} \frac{w_k}{|z_k|} + \sum_{l \in I \setminus \{i_k\}} \left| s_l w_l - \frac{z_l}{z_k} s_k w_k \right| \end{aligned}$$

Using that

$$\left| s_l w_l - \frac{z_l}{z_k} s_k w_k \right| \geq \begin{cases} 0 & , \text{ if } z_l s_l z_k s_k > 0 \\ \frac{|z_l|}{|z_k|} w_k & , \text{ else} \end{cases},$$

we find that (8) holds if

$$\begin{aligned} \min_{\substack{\mathbf{w} \geq \mathbf{0} \\ \mathbf{1}^\top \mathbf{w} = 1}} \frac{w_k}{|z_k|} + \sum_{l \in I \setminus \{i_k\}} \left| s_l w_l - \frac{z_l}{z_k} s_k w_k \right| &= \\ \min_{w_i \geq 0} \frac{w_i}{|z_i|} \left(1 + \sum_{l \in J_-} |z_l| \right) & \\ w_k / |z_k| (|z_k| + \sum_{l \in J_+} |z_l|) = 1 & \\ = \frac{1 + \sum_{l \in J_-} |z_l|}{|z_k| + \sum_{l \in J_+} |z_l|} \geq 1, & \quad (9) \end{aligned}$$

where

$$\begin{aligned} J_+ &= \{l | l \in I \setminus \{i_k\}, z_l s_l z_k s_k > 0\}, \\ J_- &= \{l | l \in I \setminus \{i_k\}, z_l s_l z_k s_k \leq 0\}. \end{aligned}$$

Rewriting (9) as

$$\begin{aligned} \sum_{l: z_l s_l z_k s_k > 0} |z_l| \leq 1 + \sum_{l: z_l s_l z_k s_k \leq 0} |z_l| \\ \Leftrightarrow 1 \geq \sum_l \text{sign}(z_l s_l z_k s_k) |z_l| \\ = \text{sign}(z_k s_k) \sum_l s_l z_l = \text{sign}(z_k s_k) (\mathbf{s}^\top \mathbf{z}), \end{aligned}$$

we recover the optimality condition in (3).

To show that strict inequality in (3) implies that $|\mathbf{s}^\top \mathbf{z}| < 1$, assume that $|\mathbf{s}^\top \mathbf{z}| \geq 1$, but $\text{sign}(s_k z_k) (\mathbf{s}^\top \mathbf{z}) < 1$ for all k . This implies that $\text{sign}(s_k z_k) = -\text{sign}(\mathbf{s}^\top \mathbf{z})$ for all k . However, if all terms $s_k z_k$ have the same sign, then $\text{sign}(s_k z_k) = \text{sign}(\mathbf{s}^\top \mathbf{z})$, giving a contradiction. Thus, strict inequality in (3) implies that $|\mathbf{s}^\top \mathbf{z}| < 1$.

We find that

$$\mathbf{s}^\top \mathbf{z} = \mathbf{s}^\top \mathbf{A}_I^{-1} \mathbf{a}_j = \mathbf{h}^\top \mathbf{a}_j$$

where $\mathbf{h} = (\mathbf{A}_I^\top)^{-1} \mathbf{s}$. This gives that if $|\mathbf{h}^\top \mathbf{a}_j| < 1$ for all $j \in I^c$, then $\mathbf{A}_I^\top \mathbf{h} = \mathbf{s}$ and $\|\mathbf{A}_I^\top \mathbf{h}\|_\infty < 1$. The vector \mathbf{h} is thus the *dual certificate* of the basis-pursuit solution [27], giving us that $\mathbf{x} = \hat{\mathbf{x}}_{BP}$. \square

7. REFERENCES

- [1] David L. Donoho, “Compressed sensing,” *Information Theory, IEEE Transactions on*, vol. 52, no. 4, pp. 1289–1306, 2006.
- [2] Emmanuel J. Candès, Justin Romberg, and Terence Tao, “Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information,” *Information Theory, IEEE Transactions on*, vol. 52, no. 2, pp. 489–509, 2006.
- [3] Emmanuel J. Candès and Michael B. Wakin, “An introduction to compressive sampling,” *Signal Processing Magazine, IEEE*, vol. 25, no. 2, pp. 21–30, 2008.
- [4] Michael Elad, *Sparse and redundant representations: from theory to applications in signal and image processing*, Springer, 2010.
- [5] Michael Lustig, David Donoho, and John M. Pauly, “Sparse mri: The application of compressed sensing for rapid mr imaging,” *Magnetic resonance in medicine*, vol. 58, no. 6, pp. 1182–1195, 2007.
- [6] Michael Elad, Mario A.T. Figueiredo, and Yi Ma, “On the role of sparse and redundant representations in image processing,” *Proceedings of the IEEE*, vol. 98, no. 6, pp. 972–982, 2010.
- [7] Moshe Mishali and Yonina C. Eldar, “From theory to practice: Sub-nyquist sampling of sparse wideband analog signals,” *Selected Topics in Signal Processing, IEEE Journal of*, vol. 4, no. 2, pp. 375–391, 2010.
- [8] Mark D. Plumbley, Thomas Blumensath, Laurent Daudet, Rémi Gribonval, and Mike E. Davies, “Sparse representations in audio and music: from coding to source separation,” *Proceedings of the IEEE*, vol. 98, no. 6, pp. 995–1005, 2010.
- [9] Noam Wagner, Yonina C. Eldar, and Zvi Friedman, “Compressed beamforming in ultrasound imaging,” *Signal Processing, IEEE Transactions on*, vol. 60, no. 9, pp. 4643–4657, 2012.
- [10] Scott Shaobing Chen, David L. Donoho, and Michael A. Saunders, “Atomic decomposition by basis pursuit,” *SIAM Rev.*, vol. 43, no. 1, pp. 129–159, Jan. 2001.
- [11] Ingrid Daubechies, Michel Defrise, and Christine De Mol, “An iterative thresholding algorithm for linear inverse problems with a sparsity constraint,” *Communications on pure and applied mathematics*, vol. 57, no. 11, pp. 1413–1457, 2004.
- [12] Emmanuel Candès and Justin Romberg, “l1-magic: Recovery of sparse signals via convex programming,” *URL: www.acm.caltech.edu/l1magic/downloads/l1magic.pdf*, 2005.
- [13] Patrick L. Combettes and Valérie R. Wajs, “Signal recovery by proximal forward-backward splitting,” *Multiscale Modeling & Simulation*, vol. 4, no. 4, pp. 1168–1200, 2005.
- [14] Mário A.T. Figueiredo, Robert D. Nowak, and Stephen J. Wright, “Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems,” *Selected Topics in Signal Processing, IEEE Journal of*, vol. 1, no. 4, pp. 586–597, 2007.
- [15] Elaine T. Hale, Wotao Yin, and Yin Zhang, “A fixed-point continuation method for l1-regularized minimization with applications to compressed sensing,” *CAAM TR07-07, Rice University*, 2007.
- [16] Stephen Boyd and Lieven Vandenberghe, *Convex optimization*, Cambridge university press, 2009.
- [17] Seung-Jean Kim, Kwangmoo Koh, Michael Lustig, Stephen Boyd, and Dimitry Gorinevsky, “An interior-point method for large-scale l1-regularized least squares,” *Selected Topics in Signal Processing, IEEE Journal of*, vol. 1, no. 4, pp. 606–617, 2007.
- [18] Amir Beck and Marc Teboulle, “A fast iterative shrinkage-thresholding algorithm for linear inverse problems,” *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 183–202, 2009.
- [19] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” *Foundations and Trends® in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.
- [20] João F.C. Mota, Joao Xavier, Pedro M.Q. Aguiar, and Markus Puschel, “Distributed basis pursuit,” *Signal Processing, IEEE Transactions on*, vol. 60, no. 4, pp. 1942–1956, 2012.
- [21] Liang Dai and Kristiaan Pelckmans, “An ellipsoid based, two-stage screening test for bpdn,” in *Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European*. IEEE, 2012, pp. 654–658.
- [22] Antoine Bonnefoy, Valentin Emiya, Liva Ralaviola, and Rémi Gribonval, “A dynamic screening principle for the lasso,” in *Signal Processing Conference (EUSIPCO), 2014 Proceedings of the 22th European*. IEEE, 2014.
- [23] Stéphane G. Mallat and Zhifeng Zhang, “Matching pursuits with time-frequency dictionaries,” *Signal Processing, IEEE Transactions on*, vol. 41, no. 12, pp. 3397–3415, 1993.
- [24] Joel A. Tropp and Anna C. Gilbert, “Signal recovery from random measurements via orthogonal matching pursuit,” *Information Theory, IEEE Transactions on*, vol. 53, no. 12, pp. 4655–4666, 2007.
- [25] Wei Dai and Olgica Milenkovic, “Subspace pursuit for compressive sensing signal reconstruction,” *Information Theory, IEEE Transactions on*, vol. 55, no. 5, pp. 2230–2249, 2009.
- [26] Deanna Needell and Joel A. Tropp, “Cosamp: Iterative signal recovery from incomplete and inaccurate samples,” *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301–321, 2009.
- [27] Hui Zhang, Wotao Yin, and Lizhi Cheng, “Necessary and sufficient conditions of solution uniqueness in l1 minimization,” *arXiv preprint arXiv:1209.0652*, 2012.
- [28] George Bernard Dantzig, *Linear programming and extensions*, Princeton university press, 1998.
- [29] Ingrid Daubechies, Ronald DeVore, Massimo Fornasier, and C. Sinan Güntürk, “Iteratively reweighted least squares minimization for sparse recovery,” *Communications on Pure and Applied Mathematics*, vol. 63, no. 1, pp. 1–38, 2010.