HOW TO USE REAL-VALUED SPARSE RECOVERY ALGORITHMS FOR COMPLEX-VALUED SPARSE RECOVERY?

Arsalan Sharif-Nassab[1], Milad Kharratzadeh[1], Massoud Babaie-Zadeh[1], and Christian Jutten[2]

[1]Department of Electrical Engineering, Sharif University of Tech., Iran,
{ars.sharif, milad.kharratzadeh, mbzadeh}@yahoo.com,
[2]GIPSA-Lab, Grenoble, and Institut Universitaire de France, France,
{Christian.Jutten@gipsa-lab.grenoble-inp.fr}

ABSTRACT

Finding the sparse solution of an underdetermined system of linear equations (the so called *sparse recovery* problem) has been extensively studied in the last decade because of its applications in many different areas. So, there are now many sparse recovery algorithms (and program codes) available. However, most of these algorithms have been developed for real-valued systems. This paper discusses an approach for using available real-valued algorithms (or program codes) to solve complex-valued problems, too. The basic idea is to convert the complex-valued problem to an equivalent real-valued problem and solve this new real-valued problem using any real-valued sparse recovery algorithm. Theoretical guarantees for the success of this approach will be discussed, too.

On the other hand, a widely used sparse recovery idea is finding the minimum ℓ^1 norm solution. For real-valued systems, this idea requires to solve a linear programming (LP) problem, but for complex-valued systems it needs to solve a second-order cone programming (SOCP) problem, which demands more computational load. However, based on the approach of this paper, the complex case can also be solved by linear programming, although the theoretical guarantee for finding the sparse solution is more limited.

1. INTRODUCTION

Consider the underdetermined system of linear equations

$$\mathbf{As} = \mathbf{x}, \qquad (1)$$

where the $n \times m$ matrix $\mathbf{A}$ and the $n \times 1$ vector $\mathbf{x}$ are known, $\mathbf{s}$ is the $m \times 1$ vector of unknowns, and $m > n$. Such a system has generally infinitely many solutions. The goal of a sparse recovery algorithm is then to find a solution for which $\|\mathbf{s}\|_0$ is minimum, where $\|\cdot\|_0$ stands for the ℓ^0 norm of a vector, that is, the number of its nonzero entries. In the case that $\mathbf{s}$ is block-sparse (i.e., where the nonzero elements of $\mathbf{s}$ occur in clusters), a block-sparse recovery algorithm aims to find a solution of (1) with minimum number of blocks having at least one nonzero element.

Sparse solutions of underdetermined linear systems have been extensively studied in the last decade due to their applications in many different problems such as compressed sensing [1], atomic decomposition on overcomplete dictionaries [2], image deconvolution [3], direction-of-arrival estimation [4], underdetermined sparse component analysis and source separation [5].

It has been shown [4, 6, 7] that the sparsest solution of (1) is *unique*, provided that it is sparse enough. More precisely, let $\mathit{Spark}(\mathbf{A})$ denote the minimum number of columns of $\mathbf{A}$ that are linearly dependent [7]. Then [6, 7]:

Theorem 1 (Uniqueness Theorem)**.** If (1) has a solution $\mathbf{s}^*$ for which $\|\mathbf{s}^*\|_0 < \frac{1}{2}\mathit{Spark}(\mathbf{A})$, it is its unique sparsest solution.

Although the sparsest solution of (1) may be unique, finding this solution requires a combinatorial search and is generally NP-hard. Hence, many algorithms have been developed so far trying to estimate the sparsest solution, for example, Basis Pursuit (BP) [8], Matching Pursuit (MP) [2], and Smoothed L0 (SL0) [9]. The idea of BP, as one of the most successful sparse recovery ideas, is replacing the ℓ^0 norm by ℓ^1 norm, that is, finding the solution of (1) for which $\|\mathbf{s}\|_1 \triangleq \sum_i |s_i|$ is minimized. It is then guaranteed that this solution is the sparsest solution provided that it is sparse enough [6, 7]. More precisely, assume that the columns of $\mathbf{A}$ are of unit ℓ^2 norm, and let $M(\mathbf{A})$ denote the *mutual coherence* of $\mathbf{A}$, *i.e.* the maximum correlation between its columns. Then:

Theorem 2. If (1) has a solution $\mathbf{s}^*$ for which $\|\mathbf{s}^*\|_0 < \frac{1}{2}(1 + M(\mathbf{A})^{-1})$, it is the (unique) solution of both ℓ^1 and ℓ^0 minimization problems.

However, many of existing sparse recovery algorithms have been developed for real-valued systems. But, there are many applications, *e.g.* where the sparsity is in the frequency domain, in which (1) is complex-valued (see for

This work has been partially funded by Iran Telecom Research Center (ITRC) and by Iran National Science Foundation (INSF).

example [10, 11, 12]). For the case of BP approach, although it works[1] also for complex domain [6, 7], it is more complicated, both in theory and in computational load. This is because minimizing ℓ^1 norm for real-valued systems requires only to solve a linear programming (LP) problem, while for complex-valued systems it needs to solve a second-order cone programming (SOCP) problem. A SOCP problem can be solved for example using "SeDuMi" toolbox [13], which is the solver used in [10, 12], and is also one of the solvers used by CVX MATLAB package [14].

In this paper, we introduce a simple approach for directly using a real-valued sparse recovery algorithm (or computer program) for solving the complex-valued case. The basic idea is to construct a real-valued system such that its sparse solution determines the sparse solution of the original complex-valued system, and then to use the real-valued sparse recovery algorithm. A problem with this approach is that the sparse solution of the complex system can be found only if its ℓ^0 norm is less than a limit which is half of the limit guaranteeing the uniqueness of the sparsest solution. So, we also present a modified approach that even if the ℓ^0 norm of the sparse solution is above this limit (but of course less than the limit guaranteeing the uniqueness), it lets to find it with probability one.

When used with BP, this approach will result to solving the complex-valued case by *linear programming* instead of SOCP. However, we will see that the sparsity limit guaranteeing that it finds the sparse solution is (at worst) half of the sparsity limit required by complex-valued BP (based on SOCP).

The paper is organized as follows. The proposed approach is presented in Section 2, and Section 3 focuses on solving complex-valued problems using linear programming.

2. THE APPROACH

2.1. Main Idea

Definition 1. We define the operator $\mathcal{R}$ for complex-valued matrices and column vectors as follows:

$$\mathcal{R}(\mathbf{A}) \triangleq \begin{bmatrix} \mathscr{Re}(\mathbf{A}) & -\mathscr{Im}(\mathbf{A}) \\ \mathscr{Im}(\mathbf{A}) & \mathscr{Re}(\mathbf{A}) \end{bmatrix} \quad \left(\begin{array}{c} \text{For non-vector} \\ \text{matrices} \end{array} \right),$$

$$\mathcal{R}(\mathbf{x}) \triangleq \begin{bmatrix} \mathscr{Re}(\mathbf{x}) \\ \mathscr{Im}(\mathbf{x}) \end{bmatrix} \quad \left(\begin{array}{c} \text{For column} \\ \text{vectors} \end{array} \right).$$

where $\mathscr{Re}$ and $\mathscr{Im}$ stand for the real and imaginary parts.

Note also that the operator $\mathcal{R}$ is *invertible*.

Lemma 1. For any $n \times m$ matrix $\mathbf{A}$ and $m \times k$ matrix $\mathbf{B}$ ($k = 1$ for column vectors), $\mathcal{R}(\mathbf{AB}) = \mathcal{R}(\mathbf{A})\mathcal{R}(\mathbf{B})$.

Proof. The proof is straightforward from the definition. □

Let now $\tilde{\mathbf{A}} \triangleq \mathcal{R}(\mathbf{A})$, $\tilde{\mathbf{s}} \triangleq \mathcal{R}(\mathbf{s})$, and $\tilde{\mathbf{x}} \triangleq \mathcal{R}(\mathbf{x})$. It is easy to see that $\mathbf{s}$ is a solution of the complex-valued system (1) *if and only if* $\tilde{\mathbf{s}}$ is a solution of the real-valued system

$$\tilde{\mathbf{A}}\tilde{\mathbf{s}} = \tilde{\mathbf{x}}. \tag{2}$$

Hence, the basic idea to find the sparse solution of (1) is:

Approach 1.
Step 1) Find the sparsest solution of the real-valued system (2) using a real-valued sparse recovery algorithm;
Step 2) The sparsest solution of (1) is $\mathbf{s} = \mathcal{R}^{-1}(\tilde{\mathbf{s}})$, where $\tilde{\mathbf{s}}$ is the vector obtained in the previous step.

Does this approach necessarily give the sparsest solution of (1)? The answer is positive provided that the sparse solution of (1) is sparse enough:

Theorem 3. If (1) has a solution $\mathbf{s}^*$ with $\|\mathbf{s}^*\|_0 < \frac{1}{4}\mathit{Spark}(\mathbf{A})$, then $\tilde{\mathbf{s}}^* \triangleq \mathcal{R}(\mathbf{s}^*)$ is the unique sparsest solution of (2).

Proof. First note that for any vector $\mathbf{s}$:

$$\|\mathbf{s}\|_0 \le \|\mathcal{R}(\mathbf{s})\|_0 \le 2\|\mathbf{s}\|_0. \tag{3}$$

Now, let $\tilde{\mathbf{s}}$ be any other solution of (2). We have to show that $\|\tilde{\mathbf{s}}^*\|_0 < \|\tilde{\mathbf{s}}\|_0$. We write $\|\tilde{\mathbf{s}}^*\|_0 \triangleq \|\mathcal{R}(\mathbf{s}^*)\|_0 \le 2\|\mathbf{s}^*\|_0 < \frac{1}{2}\mathit{Spark}\mathbf{A} \le \|\mathbf{s}\|_0 \le \|\mathcal{R}(\mathbf{s})\|_0 = \|\tilde{\mathbf{s}}\|_0$, where the inequalities are respectively due to (3), assumption, Theorem 1, and again (3). □

2.2. A modified approach

Note that the sparsity bound in Theorem 3 is half of the sparsity bound for uniqueness (Theorem 1). So, if the complex-valued system (1) has a solution with $\frac{1}{4}\mathit{Spark}(\mathbf{A}) \le \|\mathbf{s}^*\|_0 < \frac{1}{2}\mathit{Spark}(\mathbf{A})$, although it is its unique sparsest solution (Theorem 1), it is not guaranteed by Theorem 3 that Approach 1 can find it. In this subsection, we practically eliminate this limitation by modifying Approach 1 to guarantee that such a sparse solution can also be found with probability one.

[1] There are some inconsistencies in the literature about these points. In [11] it has been stated based on a counter-example that in the complex case, ℓ^1 norm minimization is no longer equivalent to ℓ^0 norm minimization (this claim has also been repeated in [12]). This is, however, not correct, because the proofs of [6] and [7] to Theorem 2 have both been stated for the complex case. In fact, in the counter-example of [11], the sparsest solution does not satisfy the condition of Theorem 2 (*i.e.* it does not satisfy $\|\mathbf{s}^*\|_0 < \frac{1}{2}(1 + M(\mathbf{A})^{-1})$, where columns of $\mathbf{A}$ are normalized). On the other hand, it has been mistakenly stated in both [6] and [7] that ℓ^1 minimization problem can be casted as a linear program, which is not true for the complex case.

Let's first call a vector *strictly complex* if none of its nonzero elements are either real or pure imaginary. If $\mathbf{s}$ is a strictly complex vector, then:

$$\|\mathcal{R}(\mathbf{s})\|_0 = 2\|\mathbf{s}\|_0. \tag{4}$$

Using this equation (instead of (3)) in the last inequality of the proof of Theorem 3 gives:

Lemma 2. Suppose that (1) has a solution $\mathbf{s}^*$ for which $\|\mathbf{s}^*\|_0 < \frac{1}{2}\mathit{Spark}(\mathbf{A})$, and let $\mathbf{s}$ be any other solution of (1). If $\mathbf{s}$ is strictly complex, then $\|\mathcal{R}(\mathbf{s}^*)\|_0 < \|\mathcal{R}(\mathbf{s})\|_0$.

Proof. We can write $\|\mathcal{R}(\mathbf{s}^*)\|_0 \le 2\|\mathbf{s}^*\|_0 < \mathit{Spark}(\mathbf{A}) \le 2\|\mathbf{s}\|_0 = \|\mathcal{R}(\mathbf{s})\|_0$, where the inequalities are respectively due to (3), assumption, Theorem 1, and (4). □

However, it is not guaranteed that a solution $\mathbf{s}$ of the deterministic system (1) is strictly complex. But, we can add some 'randomness' to solve this problem. Note that (1) is equivalent to $\mathbf{A} \cdot (\alpha\mathbf{s}) = (\alpha\mathbf{x})$ for every nonzero complex scalar α. In other words, $\mathbf{s}$ is a solution of (1) if and only if $\mathbf{s}_\alpha \triangleq \alpha\mathbf{s}$ is a solution of

$$\mathbf{A}\mathbf{s}_\alpha = \mathbf{x}_\alpha, \tag{5}$$

where $\mathbf{x}_\alpha \triangleq \alpha\mathbf{x}$. Now, if we choose a nonzero α randomly using a complex density (*e.g.*, uniformly on the unit circle), then for a fixed $\mathbf{s}$, $\mathbf{s}_\alpha$ would be strictly complex with probability one. Thus, by applying Approach 1 to system (5), we arrive to the idea of using the following modified approach:

Approach 2.

Step 1) Randomly choose a complex number α (*e.g.*, using a uniform distribution on the unit circle).

Step 2) Find the sparsest solution of the real-valued system

$$\tilde{\mathbf{A}}\tilde{\mathbf{s}}_\alpha = \tilde{\mathbf{x}}_\alpha, \tag{6}$$

where $\tilde{\mathbf{A}} = \mathcal{R}(\mathbf{A})$ and $\tilde{\mathbf{x}}_\alpha = \mathcal{R}(\alpha\mathbf{x})$.

Step 3) The sparsest solution of (1) is $\mathbf{s} = \mathcal{R}^{-1}(\tilde{\mathbf{s}}_\alpha)/\alpha$, where $\tilde{\mathbf{s}}_\alpha$ is the vector obtained in the previous step.

Note that the above reasoning was only a first justification to obtain the modified approach, not a proof. The problem is that although for a *fixed* $\mathbf{s}$, $\mathbf{s}_\alpha$ is strictly complex with probability one, the above reasoning requires that *for all* solutions $\mathbf{s}$ of (1), $\mathbf{s}_\alpha$ is strictly complex, which is not true, because (6) has infinitely many solutions $\mathbf{s}_\alpha$ that are not strictly complex (for example, fix one of the entries of $\mathbf{s}_\alpha$ to a real number and solve the system for the remaining unknowns). So, we need to state a formal proof for validity of the above approach. The following theorem guarantees for Approach 2 the same bound of uniqueness as Theorem 1.

Theorem 4. If (1) has a solution $\mathbf{s}^*$ for which $\|\mathbf{s}^*\|_0 < \frac{1}{2}\mathit{Spark}(\mathbf{A})$ (that is the whole uniqueness range), then $\mathcal{R}(\mathbf{s}^*_\alpha)$, where $\mathbf{s}^*_\alpha \triangleq \alpha\mathbf{s}^*$, is the unique sparsest solution of the real-valued system (6) with probability one.

The main point for proving the above theorem is that we will not need that $\alpha\mathbf{s}$ is strictly complex *for all* solutions of (1); we will need this property only for solutions with $\|\mathbf{s}\|_0 < \mathit{Spark}(\mathbf{A})$), which is guaranteed by the following lemma.

Lemma 3. If α is a randomly chosen complex number, then for all solutions $\mathbf{s}$ of (1) satisfying $\|\mathbf{s}\|_0 < \mathit{Spark}(\mathbf{A})$, $\alpha\mathbf{s}$ is strictly complex with probability one.

Proof. The point is that (1) can have only a finite number of solutions that satisfy $\|\mathbf{s}\|_0 < \mathit{Spark}(\mathbf{A})$. To prove this claim, let $p \triangleq \mathit{Spark}(\mathbf{A}) - 1$, and note that for any set of indices $I = \{i_1, i_2, \cdots, i_p\}$, $1 \le i_k \le m$, there is at most one solution of (1) whose support is a subset of I. This can be shown by contradiction: suppose that $\mathbf{s}_1$ and $\mathbf{s}_2$ are two solutions of (1), and their supports both lie in I. Then $\mathbf{A}(\mathbf{s}_1 - \mathbf{s}_2) = \mathbf{x} - \mathbf{x} = 0$, while the support of $\mathbf{s}_1 - \mathbf{s}_2$ is in I. In other words, we have found less than or equal p columns of $\mathbf{A}$ that are linearly dependent, which contradicts the definition of $\mathit{Spark}(\mathbf{A})$. Now, since only $\binom{m}{p}$ number of such I's exist, (1) has at most $\binom{m}{p}$ number of solutions with $\|\mathbf{s}\|_0 < \mathit{Spark}(\mathbf{A})$.

Now, since the number of such solutions is finite, multiplying them by a randomly chosen complex number α, all of the products would be strictly complex with probability one. □

Proof of Theorem 4. Let $\tilde{\mathbf{s}}_\alpha = \mathcal{R}(\mathbf{s}_\alpha)$ be any other solution of (6). Then, $\mathbf{s}_\alpha$ is a solution of the complex-valued system (5) and $\mathbf{s} = \mathbf{s}_\alpha/\alpha$ is a solution of (1). There are two possibilities (note that $\forall\mathbf{s}$, $\|\alpha\mathbf{s}\|_0 = \|\mathbf{s}\|_0$):

- $\|\mathbf{s}\|_0 \ge \mathit{Spark}(\mathbf{A})$: In this case, $\|\mathcal{R}(\mathbf{s}^*_\alpha)\|_0 \le 2\|\mathbf{s}^*_\alpha\|_0 < \mathit{Spark}(\mathbf{A}) \le \|\mathbf{s}_\alpha\|_0 \le \|\mathcal{R}(\mathbf{s}_\alpha)\|_0$, where the inequalities are respectively due to (3), assumption, assumption, and (3).
- $\|\mathbf{s}\|_0 < \mathit{Spark}(\mathbf{A})$: Here, according to Lemma 3, $\mathbf{s}_\alpha$ is strictly complex with probability one, so, applying Lemma 2 to (5) gives $\|\mathcal{R}(\mathbf{s}^*_\alpha)\|_0 < \|\mathcal{R}(\mathbf{s}_\alpha)\|_0$. □

3. COMPLEX-VALUED SPARSE RECOVERY BASED ON LINEAR PROGRAMMING

Approaches 1 and 2 aim to obtain the sparse solution of a complex-valued system from the sparse solution of a real-valued system. However, the true sparse solution of this new real-valued system is not itself known, and real-valued sparse recovery algorithms are providing only *estimations* of it. In this section, we focus on the combination of Approach 1 (or 2) and BP for estimating the sparse solution of (2), as an approach for estimating the sparse solution of (1), and we study the condition that this estimator gives the true sparse solution of (1). It is interesting to note that this approach estimates the sparse solution of the complex-valued system (1) using *linear programming*, not SOCP.

Remark 1. One can also exploit the equivalence of the real-valued system (2) (or (6)) and the complex-valued system (1) by looking at $\tilde{\mathbf{s}}$ (or $\tilde{\mathbf{s}}_\alpha$) as a *block sparse* vector, with blocks of size 2, and then to use block sparse recovery algorithms for solving the real-valued system. This approach, however, does not use real-valued 'sparse recovery' algorithms, and requires more complicated real-valued 'block sparse recovery' algorithms. For example, the mixed l_2/l_1 minimization block sparse recovery [15] needs again to solve a SOCP.

Note that the proposed method in the previous sections can utilize any real-valued sparse recovery algorithm such as OMP and SL0. However, these two algorithms have sophisticated complex-valued versions [16, 17] (i.e., with the same complexity as their real-valued versions), and there is no need to use the proposed approach to handle the complex case. Note also that the dimension of the real-valued system (2) to be solved is twice as the dimension of the original complex system.

3.1. Theoretical guarantee

In this section, we assume that the columns of $\mathbf{A}$ have unit ℓ^2 norms. So, Theorem 2 guarantees that BP can find the sparsest solution if its ℓ^0 norm is less than $(1+M(\mathbf{A})^{-1})/2$. The next lemma states a relation between $M(\mathbf{A})$ and $M(\mathcal{R}(\mathbf{A}))$.

Lemma 4. Let $\mathbf{A}$ be a complex-valued $n \times m$ matrix ($n \leq m$), and let $\tilde{\mathbf{A}} \triangleq \mathcal{R}(\mathbf{A})$. Then $M(\tilde{\mathbf{A}}) \leq M(\mathbf{A})$.

Proof. It can be easily seen that the columns of $\tilde{\mathbf{A}}$ have also unit ℓ^2 norms. Let $\mathbf{C} \triangleq \mathbf{A}^H\mathbf{A}$. Then, according to Lemma 1:

$$\tilde{\mathbf{C}} \triangleq \mathcal{R}(\mathbf{C}) = \mathcal{R}(\mathbf{A}^H)\mathcal{R}(\mathbf{A}) = \mathcal{R}(\mathbf{A})^H\mathcal{R}(\mathbf{A}) = \tilde{\mathbf{A}}^H\tilde{\mathbf{A}}$$

$$\Rightarrow |c_{ij}|^2 = \tilde{c}_{ij}^2 + \tilde{c}_{i,j+m}^2 = \tilde{c}_{i+m,j}^2 + \tilde{c}_{i+m,j+m}^2$$

$$\Rightarrow |\tilde{c}_{k,l}| \leq |c_{k \ (\mathrm{mod}\ m),\ l \ (\mathrm{mod}\ m)}|, \quad 1 \leq k,l \leq 2m, \quad (7)$$

where c_{ij} and $\tilde{c}_{ij}$ stand for the (i,j)'th elements of $\mathbf{C}$ and $\tilde{\mathbf{C}}$, respectively. The mutual coherences, $M(\mathbf{A})$ and $M(\tilde{\mathbf{A}})$, are the largest magnitude non-diagonal elements of $\mathbf{C}$ and $\tilde{\mathbf{C}}$, respectively. According to (7), the largest magnitude non-diagonal element of $\mathbf{C}$ is not less than the largest magnitude non-diagonal element of $\tilde{\mathbf{C}}$, and hence $M(\tilde{\mathbf{A}}) \leq M(\mathbf{A})$. □

The fact that the mutual coherence of $\tilde{\mathbf{A}}$ is smaller than that of $\mathbf{A}$ is a desirable property, because it increases the $(1+M^{-1})/2$ bound. However, $\|\mathcal{R}(\mathbf{s})\|_0$ can be as large as $2\|\mathbf{s}\|_0$, which decreases the sparsity bound guaranteeing sparse recovery:

Theorem 5. If the complex-valued system (1) has a sparse solution $\mathbf{s}^*$ satisfying $\|\mathbf{s}^*\|_0 < \frac{1}{4}(1+M(\mathbf{A})^{-1})$, then it is guaranteed that the minimum ℓ^1 norm solution of the real-valued system (2) gives this sparse solution.

Proof. Let $\tilde{\mathbf{s}}^* \triangleq \mathcal{R}(\mathbf{s}^*)$. Then $\|\tilde{\mathbf{s}}^*\|_0 \leq 2\|\mathbf{s}^*\|_0 < \frac{1}{2}(1+M(\mathbf{A})^{-1}) \leq \frac{1}{2}(1+M(\tilde{\mathbf{A}})^{-1})$, where the inequalities are respectively due to (3), assumption, and Lemma 4. Therefore, Theorem 2 guarantees that the minimum ℓ^1 norm solution of (2) is equal to $\tilde{\mathbf{s}}^*$, which gives $\mathbf{s}^* = \mathcal{R}^{-1}(\tilde{\mathbf{s}}^*)$. □

3.2. Experiment

The bound given in Theorem 5 to guarantee finding the sparse solution by applying LP-based BP on the real-valued system (2) is half of the bound given in Theorem 2 to guarantee this by applying SOCP-based BP directly on the complex-valued system (1). However, note that these are theoretical 'guarantees' and do not reflect necessarily the performance in a typical scenario. This is both because the bound of Theorem 2 does not itself show the typical behavior, and because Lemma 4 does not show how much $M(\tilde{\mathbf{A}})$ is smaller than $M(\mathbf{A})$ in a typical case.

In this subsection, a simple experiment is done for comparing the bounds that these two approaches fail to estimate the sparsest solution. We used CVX MATLAB software [14] (Version 1.21) based on SeDuMi for both applying SOCP-based BP on (1) and applying LP-based BP on (2). As the performance measure, we computed the Percentage Of Failures (POF) of the algorithms, where a failure means $\|\hat{\mathbf{s}}_0 - \mathbf{s}^*\|_\infty > \epsilon$, in which $\mathbf{s}^*$ is the true sparsest solution, $\hat{\mathbf{s}}_0$ is its estimation, and ϵ is a fixed threshold.

In our simulation, a random matrix $\mathbf{A}$ of size 50×100 and an original sparse vector $\mathbf{s}^*$ (with random support) were generated with elements whose real and imaginary parts are independently drawn from uniform distribution on $(-0.5, 0.5)$. Then, we calculated $\mathbf{x} \triangleq \mathbf{A}\mathbf{s}^*$, and gave $\mathbf{A}$ and $\mathbf{x}$ to both sparse recovery algorithms to solve (1) and produce an estimation of $\mathbf{s}^*$. Fig. 1 shows the resulted POF's (using $\epsilon = 0.01$) and the run-times of both algorithms on the same machine (as a rough comparison of complexities) versus $\|\mathbf{s}^*\|_0$ (each point is averaged over 500 runs of the simulation with different $\mathbf{A}$ and $\mathbf{s}^*$). It is seen that the sparsity range in which the real-valued approach has failed is not as small as half of the sparsity range of the complex-valued approach. It is also noticeable that although the real-valued approach has to solve a problem with double size, because of the simplicity of LP compared to SOCP, it has performed essentially faster[2].

4. CONCLUSIONS

In this paper we introduced an approach which enables us to use any real-valued sparse recovery algorithm (or available computer program) for solving complex-valued sparse

[2] One may argue that the speed advantage obtained here is not worth a new algorithm. Note however that: 1) This speed advantage has been obtained by using an even simpler algorithm, not by using a more complex algorithm (LP instead of SOCP); 2) the main objective of the paper is not to obtain a 'faster' algorithm, it is to obtain an approach for using available real-valued sparse recovery algorithms and program codes to solve complex-valued problems.

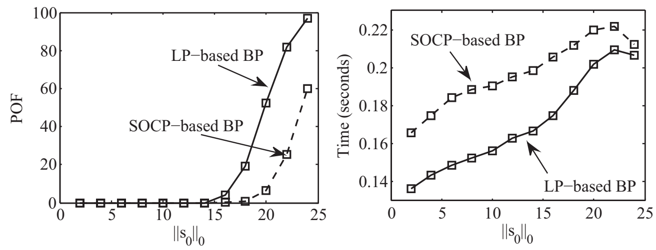

Fig. 1. Comparison of applying SOCP-based BP on the complex system (1) and applying LP-based BP on the real system (2). The figures depict Percentage of Failure (POF) and estimation time versus sparsity.

recovery, too. We saw that with a simple randomization trick, this approach can be applied for the whole sparsity range that guarantees the uniqueness of the sparsest solution. It was interesting that where BP is used to estimate the solution of the new real-valued system, the final sparse recovery algorithm would solve a complex-valued problem using linear programming, not SOCP which is usually needed. However, the guarantee that this LP-based BP can find the true sparsest solution is more limited compared to the usual SOCP-based BP.

5. REFERENCES

[1] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, April 2006.

[2] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. on Signal Proc.*, vol. 41, no. 12, pp. 3397–3415, 1993.

[3] M. A. T. Figueiredo and R. D. Nowak, "An EM algorithm for wavelet-based image restoration," *IEEE Transactions on Image Processing*, vol. 12, no. 8, pp. 906–916, 2003.

[4] I. F. Gorodnitsky and B. D. Rao, "Sparse signal reconstruction from limited data using FOCUSS, a reweighted minimum norm algorithm," *IEEE Transactions on Signal Processing*, vol. 45, no. 3, pp. 600–616, March 1997.

[5] R. Gribonval and S. Lesage, "A survey of sparse component analysis for blind source separation: principles, perspectives, and new challenges," in *Proceedings of ESANN'06*, April 2006, pp. 323–330.

[6] R. Gribonval and M. Nielsen, "Sparse decompositions in unions of bases," *IEEE Trans. Inform. Theory*, vol. 49, no. 12, pp. 3320–3325, Dec. 2003.

[7] D. L. Donoho and M. Elad, "Optimally sparse representation in general (nonorthogonal) dictionaries via ℓ^1 minimization," *Proc. Nat. Aca. Sci.*, vol. 100, no. 5, pp. 2197–2202, March 2003.

[8] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 43, no. 1, pp. 129–50, 2001.

[9] H. Mohimani, M. Babaie-Zadeh, and Ch. Jutten, "A fast approach for overcomplete sparse decomposition based on smoothed ℓ^0 norm," *IEEE Transactions on Signal Processing*, vol. 57, no. 1, pp. 289–301, January 2009.

[10] P. Bofill, "Underdetermined blind separation of delayed sound sources in the frequency domain," *Neurocomputing*, vol. 55, no. 3–4, pp. 627–641, 2003.

[11] S. Winter, H. Sawada, and Sh. Makino, "On real and complex valued ℓ_1-norm minimization for overcomplete blind source separation," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, NY, October 2005, pp. 86–89.

[12] R. Saab, Ö. Yilmaz, M. J. McKeown, and R. Abugharbieh, "Underdetermined anechoic blind source separation via ℓ^q-basis-pursuit with $q < 1$," *IEEE Transactions on Signal Processing*, vol. 55, no. 8, pp. 4004–4017, August 2007.

[13] J. F. Sturm, "Using SeDuMi 1.02, a matlab toolbox for optimization over symmetric cones," *Optimization. Methods Software*, vol. 11–12, pp. 625–653, 1999, Available at `http://sedumi.ie.lehigh.edu/`.

[14] M. Grant, S. Boyd, and Y. Ye., "Disciplined convex programming," in *Global Optimization: from Theory to Implementation*, L. Liberti and N. Maculan, Eds., pp. 155–210. Springer, 2006, Available at `http://cvxr.com/cvx/`.

[15] M. Stojnic, F. Parversh, and B. Hassibi, "On the reconstruction of block-sparse signals with an optional number of measurements," *IEEE Trans. On Sig. Proc.*, 2009.

[16] Y. Pati, R. Rezaiifar, and P. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proceedings of the 27 th Annual Asilomar Conference on Signals, Systems, and Computers*, Nov. 1993.

[17] H. Mohimani, M. Babaie-Zadeh, and C. Jutten, "Complex-valued sparse representation based on smoothed ℓ^0 norm," in *Proceedings of* ICASSP2008, Las Vegas, April 2008, pp. 3881–3884.