NP-hardness of ℓ_0 minimization problems: revision and extension to the non-negative setting

Vu Thi Hai Yen[1] and Abdel Wedoud Mohamed[2]

Master Datascience 2019-2020

[1]thi-hai-yen.vu.m2@polytechnique.edu, [2]mohamed.abdel-wedoud@polytechnique.edu

April 20, 2021

Abstract

This report aims to present the work in [1], where the NP-hardness of ℓ_0 minimization problems is studied. They revisit previous analyses of ℓ_0, point out the defects in some existing works, and propose new results to the NP-hardness of the penalized ℓ_0 problem based on previous works using reducibility from the X3C problem. They also propose new results on the NP-hardness of non-negative ℓ_0 minimization problems. Following this work, we finally propose a complete analysis on the NP-hardness of the penalized ℓ_0 problem.

1 Introduction

In compressed sensing, we are always interested in recovering exactly or approximately a signal in a high dimensional space from a small number of measurements so that the signals can have a much simpler and smaller representations. This has various applications, for example in photography, audio compression or finance, and is very important especially in this world where all types of data is being digitalized and is stored numerically.

More formally, the above problem can be represented mathematically as follows: Given a vector of measures (or signal data/observations) $\mathbf{y} \in \mathbb{R}^m$, and a matrix (or dictionary) $A \in \mathbb{R}^{m \times n}$ consisting of m measurement vectors (in $\mathbb{R}^n$, with $m \ll n$), the objective is to find (or reconstruct) the signal vector $\mathbf{x} \in \mathbb{R}^n$ that satisfies $\mathbf{y} = A\mathbf{x}$ under the conditions that $\mathbf{x}$ is sparse (*i.e.*, having a small number of non-zero coefficients). This aims in solving the following *linearly constrained* minimization problem :

$$\min_{\mathbf{x} \text{ s.t. } \mathbf{y}=A\mathbf{x}} \|x\|_0 \qquad (\ell_0 LC)$$

However, when the signal contains noises and we are only interested in approximating the signal, the linear constraint can be transformed into to an approximation inequality constraint :

$$\min_{\mathbf{x} \text{ s.t. } \|\mathbf{y}-A\mathbf{x}\|_2 \leq \epsilon} \|x\|_0 \tag{$\ell_0 C$}$$

where $\epsilon > 0$ is related to the noise standard deviation. Another formulation of the above reconstruction problem is the following problem, where we aim to find the best approximation given a sparsity constraint on the signal :

$$\min_{\mathbf{x} \text{ s.t. } \|x\|_0 \leq K} \|\mathbf{y} - A\mathbf{x}\|_2^2 \tag{$\ell_0 C'$}$$

where $K > 0$ is the sparsity level. Finally, a *penalized* (or regularized) version of the ℓ_0 minimization problem can be formulated as follows :

$$\min_{\mathbf{x}} \|\mathbf{y} - A\mathbf{x}\|_2^2 + \lambda \|x\|_0 \tag{$\ell_0 P$}$$

where λ refers to the regularization strength.

The three problems ($\ell_0 C$), ($\ell_0 C'$), ($\ell_0 P$) formulated above are closely related to each other and we can show (by Lagrangian duality) that the solution of a problem can be obtained by solving the others for a *certain* parameter. For example, if x^* is a solution of ($\ell_0 C'$), then there exists λ^* such that x^* is also a solution of ($\ell_0 P$) for $\lambda = \lambda^*$. However, in general we cannot say that the three problems are equivalent, and therefore solving them may require different time complexities.

Naively, the problem ($\ell_0 LC$) (and similarly (ℓ_0C), ($\ell_0 C'$)) can be solved by iterating through all subsets $J \subset \{1, \ldots, n\}$ with $|J| = s$ with $s = 1, 2, \ldots$ until we find a solution $\mathbf{x}$ to the linear equation $\mathbf{y} = A_J \mathbf{x}$, where $A_J \in \mathbb{R}^{m \times s}$ is the matrix containing the columns of A with the indices in J. However, this algorithm requires at each step s a total number of $\binom{n}{s}$ iterations (which is the number of s-element subsets of J), making the execution complexity until $s = S$ (with the assumption that we can solve the linear equation in constant time) :

$$\sum_{i=1}^{S} \binom{n}{i} \geq \sum_{i=1}^{S} \left(\frac{n}{i}\right)^i \geq \left(\frac{n}{S}\right)^S \tag{1}$$

which is exponential in S. This algorithm is therefore very inefficient and as a result, we may want to know if there exists any algorithm to the above problems that can be solved within polynomial time.

Unfortunately, as the essence of this work, all the aforementioned problems are shown (or at least partly-shown) to be fundamentally difficult as other classic problems that cannot (or not yet) be solved in polynomial time. We will study in the next section the existing analyses of complexity for the given ℓ_0 minimization problems as well as some defects in the analyses of ($\ell_0 P$). In section 3, we focus on the main contributions of [1] on the complexity analysis of ($\ell_0 P$) and new results on the non-negative minimization problems. On the light of theses results, we further extend the analysis of ($\ell_0 P$) to show the NP-hardness of ($\ell_0 P$). We'll then give some algorithmic discussions in the section 5 before concluding our work.

2 Previous complexity analyses on ℓ_0 minimization problems

2.1 Revision of complexity theory

Before discussing the complexities of ℓ_0 minimization problems, we will need nevertheless to recall some elementary notions of complexity analysis.

A problem is said to be in class P if there exists an algorithm to solve it that executes within polynomial time. The class NP is a generalisation of class P that consists of problems for which solutions can be verified (but not necessarily solved) in polynomial time. It's worth noting that P is a subset of NP.

To compare the problems in NP, we'll need the notion of reducibility. We say that a problem X can be reduced to problem Y (in polynomial time) if we can first map each instance of X to an instance of Y in polynomial time, and then solving Y leads to solving X. In that case, Y is said to be at least as hard as X since any algorithm that solves Y can also solve X.

Consequently, we can define the class of NP-complete problems, which is the set of problems in NP to which all other problems in NP can be reduced in polynomial time. In other words, any problem in NP-complete is at least as hard other problems in NP, and therefore they are the most difficult problems in NP. It is generally believed (though not proven) that the set of NP-complete problems is disjoint with the set of P problems. Finally, a problem is called NP-hard if there exists a NP-complete problem that can be reduced to it in polynomial time. NP-hard problems are therefore at least as hard as NP-complete problems, and do not need to be in NP (if it is, then it's NP-complete).

There are a lot of known NP-complete problems which comprises a variety of domains such as graph theory, mathematical programming, language theory, etc. ([2]). In this work, we will introduce a well-known NP-complete problem named *exact cover by 3-set (X3C)* ([2, p. 221]) that is mainly used to show the NP-hardness of the ℓ_0 minimization problems.

The X3C problem is announced as follows: Given a set X and a collection C of 3-element subsets of X. Does C contains an exact cover of X, *i.e.* a subcollection $C' \subseteq C$ such that C' exactly covers X ?

This problem has been shown to be NP-complete by reduction from the *3-dimensional matching (3DM)* problem, which is reduced from the *3-SAT* problem, which is finally obtained from the *Boolean satisfiability (SAT)* problem, the first problem that was proven to be NP-complete ([3]).

2.2 NP-hardness of $(\ell_0 LC)$, $(\ell_0 C)$ and $(\ell_0 C')$

Given the notion of NP-hardness, we can now return to the complexity analysis of our ℓ_0 minimization problems. Note that n and K are often depend on m when one considers the size of the problems.

Firstly, the problem ($\ell_0 LC$) has been shown to be NP-hard by reduction from the X3C problem. The construction is based on the NP-hardness proof of the *minimum weight solutions to linear systems* problem in [2, p. 246], which can be found in [4, Theorem 3.2]. The sketch of the proof is as follows: Each instance of X3C is first mapped (in polynomial time) to an instance of ($\ell_0 LC$) by constructing a matrix A whose columns are the indicator vectors of the subsets C and the vector $\mathbf{y}$ is taken to be $\mathbb{1}$-vector. Then the reduction is justified by showing that each solution of X3C is equivalent to a solution of ($\ell_0 LC$). Similarly, the problem ($\ell_0 C$) has also been shown to be NP-hard ([5]) with the similar construction from the X3C problem.

The problem ($\ell_0 C'$) has later been shown to be NP-hard in [6, Theorem 2.1], under the name *M-optimal approximation*. They showed that it is at least as hard as the (ϵ, M)-*approximation* problem, which is shown to be NP-complete by reduction from X3C.

2.3 NP-hardness of $(\ell_0 P)$ and existing defects

It is worth noting in this section that the NP-hardness of ($\ell_0 P$) is considered for each fixed λ, otherwise we can easily deduce from the Lagrangian duality above that ($\ell_0 P$) is NP-hard.

The penalized problem $(\ell_0 P)$ was claimed to be NP-hard as a particular case of more general complexity analyses ([7, 8]).

Indeed, in [7], it was shown that a greater class of problems denoted by $(\ell_q\text{-}\ell_p)_{\{\lambda>0,\ 0\leq p<1,\ q\geq 2\}}$:

$$\min_{\mathbf{x}} \|\mathbf{y} - A\mathbf{x}\|_q^q + \lambda \|\mathbf{x}\|_p^p \tag{ℓ_q-ℓ_p}$$

are all NP-hard problems. The idea of the prove consists of two main steps : they set an invertible transformation that maps all problems of that class to a subset S of the same class such that $\lambda = \frac{1}{2}$ (within a polynomial time), then they managed to find an equivalence between S and the partition problem, another well-known NP-complete problem ([2, p. 224]). However, the employed transformation :

$$\tilde{\mathbf{x}} = (2\lambda)^{1/p}\mathbf{x}, \quad \tilde{A} = (2\lambda)^{-1/p} A \tag{2}$$

is not well-defined for $p = 0$. As a consequence, the result cannot be applied to the case when $p = 0$ and $\lambda \neq 1/2$.

Alternatively, [8] proposes a proof of NP-hardness for the following *penalized least-squares* problem :

$$\min_{\mathbf{x}} \|\mathbf{y} - A\mathbf{x}\|_2^2 + \lambda \sum_{i=1}^{n} \phi(|x_i|) \tag{PLS}$$

where $\phi : \mathbb{R}^+ \to \mathbb{R}^+$ is a penalty function satisfying the following conditions: (C1) $\phi(0) = 0$ and $\phi(\tau_1) \leq \phi(\tau_2),\ \ \forall 0 \leq \tau_1 < \tau_2$. (C2) $\exists \tau_0 > 0, d > 0$ such that $\phi(\tau) \geq \phi(\tau_0) - d(\tau_0 - \tau)^2,\ \ \forall 0 \geq \tau < \tau_0$. (C3) $\phi(\tau_1) + \phi(\tau_2) \geq \phi(\tau_1 + \tau_2),\ \ \forall \tau_1, \tau_2 < \tau_0$ for the aforementioned τ_0. (C4) $\phi(\tau) + \phi(\tau_0 - \tau) > \phi(\tau_0),\ \ \forall 0 \leq \tau < \tau_0$. They claimed that the ℓ_0 function satisfies the conditions C1-C4 for $\tau_0 = d = 1$. However, the condition (C4) is itself contradictory with the condition (C1) when taking $\tau = 0$, resulting in $\phi(\tau_0) > \phi(\tau_0)$.

In addition, [8] also mentions another proof of NP-hardness of ($\ell_0 P$) that establishes the equivalence between ($\ell_0 P$) and (ℓ_0C) using Lagrangian duality. However, as we mentioned earlier, this equivalence is only up to certain parameter and therefore is not guaranteed to be true. In addition, the mentioned transformation between ($\ell_0 P$) and (ℓ_0C) is not polynomial.

3 Main results

In this section, we'll provide the main results of [1], which concentrates on proving the NP-hardness of ($\ell_0 P$) for $0 < \lambda < 3$ as well as giving a new complexity analysis on non-negative ℓ_0 minimization problems.

3.1 New analysis on penalized ℓ_0 minimization problems

The following theorem makes use of the construction from an instance of the X3C problem as in [5] so as to provide the NP-hardness of the penalized ℓ_0 minimization problem for a certain λ.

Theorem 1. *The problem* $(\ell_0 P)$ *is NP-hard for* $0 < \lambda < 3$.

Proof. To prove this result we will proceed in three classical main steps :

- First we will try to construct an instance of $(\ell_0 P)$ from an instance of the X3C problem. To do so, let S be a set of m distinct elements say $\{s_1, ..., s_m\}$ and $C = (c_1, ..., c_n)$ a collections of triplets included in S. Similar to what is described in Section 2.2, we set $y = (1, .., 1)^T$ and define $A = (a_{ij})_{1 \leq i \leq m, 1 \leq j \leq n}$ such that $a_{ij} = \mathbb{1}_{\{s_i \in c_j\}}$. We get minimization problem (instance of $(\ell_0 P)$) of the following function :

$$F(x) := \|\mathbf{y} - A\mathbf{x}\|_2^2 + \lambda \|x\|_0 \tag{3}$$

- To be able to assume existence of solutions for the X3C, we assume that the setting above is subject to m being multiple of 3 otherwise it does not admit any solution. Now, we construct a solution for the $(\ell_0 P)$ problem from a solution of X3C. Let's say that we have a perfect recovery of S. We denote the solution by $C^* \subset C$ and define x^* such that $x_i^* = \mathbb{1}_{\{c_i \in C^*\}}$. We'll now show that x^* is a solution to the minimization of F (above) problem. To do that, we proceed by an absurdity argument : assume there exits x_1 such that $F(x_1) < F(x^*) = \frac{\lambda m}{3}$. If such thing happens to be true and as $F(x_1) \geq \lambda \|x_1\|_0$, it follows that $\|x_1\|_0 = \frac{m}{3} - b < \frac{m}{3}$ with $b \in \mathbb{N}^*$. As Ax_1 identifies with the s_j's covered by x_1, we get that $\|y - Ax_1\|_2^2 > 3b$ and then that $F(x_1) \geq \frac{\lambda m}{3}$ which leads to a contradiction because $F(x_1)$ is supposed to be strictly smaller than $\frac{\lambda m}{3}$.
- The last thing, that remains to prove is that $(\ell_0 P)$ is NP-hard for $0 < \lambda < 3$, is the construction of a solution of X3C from a solution of $(\ell_0 P)$: which we denote by $\overline{x}$. And consider x^* as defined in the previous step. We proceed by analyzing 3 cases :
  1. If $\|\overline{x}\|_0 \neq \frac{m}{3}$ and X3C admits a solution we will have that $F(\overline{x}) > F(x^*)$ which contradicts the fact that $\overline{x}$ is the solution of $(\ell_0 P)$.
  2. If $\|\overline{x}\|_0 = \frac{m}{3}$, $y \neq A\overline{x}$ and X3C admits a solution we can also show that $F(\overline{x}) > F(x^*)$ which leads to a contradiction
  3. Finally if $\|\overline{x}\|_0 = \frac{m}{3}$, $y = A\overline{x}$ and X3C admits a solution, we will get that $F(\overline{x}) = F(x^*)$ and thus $\overline{x}$ is fully descriptive of a solution of of X3C.

$\square$

Note that the proof of the above theorem is still applicable if we change the squared ℓ_2 norm by the ℓ_p norm to power p ($\|\cdot\|_p^p$), which leads to this theorem :

Theorem 2. *The following problem (ℓ_p-$\ell_0 P$) :*

$$\min_{\boldsymbol{x}} \|\boldsymbol{y} - A\boldsymbol{x}\|_p^p + \lambda \|x\|_0 \tag{ℓ_p-$\ell_0 P$}$$

is NP-hard for $p \geq 1$ and $0 < \lambda < 3$.

Proof. The proof lays on the same steps as in the theorem 1 proof:

- The setting (instance construction) is the same as before
- If one were to construct a solution for the (ℓ_p-$\ell_0 P$) problem from a solution of X3C: he or she may have to prove that $\|y - Ax_1\|_p^p > 3b$ where $F(x_1)$ is supposed to be strictly smaller than $F(x^*)$. Indeed, that's actually the case as $y - Ax_1$ has at least $3b$ components that are not null.
- Finally, we prove that if X3C admits a solution then the solution $\overline{x}$ is such that $\|\overline{x}\|_0 = \frac{m}{3}$, $y = A\overline{x}$ and $F(\overline{x}) = F(x^*)$.

□

3.2 Extensions to non-negative ℓ_0 minimization problems

In many applications, the signal vector is forced to be non-negative, which transforms the ℓ_0 minimization problems into the following problems :

$$\min_{\mathbf{x}\geq 0 \text{ s.t. } \|\mathbf{y}-A\mathbf{x}\|_2 \leq \epsilon} \|x\|_0 \tag{$\ell_0 C+$}$$

$$\min_{\mathbf{x}\geq 0 \text{ s.t. } \|x\|_0 \leq K} \|\mathbf{y} - A\mathbf{x}\|_2^2 \tag{$\ell_0 C'+$}$$

$$\min_{\mathbf{x}\geq 0} \|\mathbf{y} - A\mathbf{x}\|_2^2 + \lambda \|x\|_0 \tag{$\ell_0 P+$}$$

As we'll show shortly, these problems are also NP-hard and therefore are as difficult as normal ℓ_0 minimization problems.

Theorem 3. *The problems ($\ell_0 C+$), ($\ell_0 C'+$) are NP-hard. The problem ($\ell_0 P+$) is NP-hard for $0 < \lambda < 3$.*

Proof. In fact, the proof of NP-hardness for the given problems is conducted exactly the same ways as presented in [5, 6] and in Theorem 1, based on the fact that all these solutions rely on reducing the X3C problem to the desired problem. In all the solutions, the transformed instance $\mathbf{x}$ is always 0 or 1, hence is non-negative. As a result, the transformed instance is also an instance of the non-negative version of the corresponding problem. Hence, we can always reduce the X3C problem to the corresponding non-negative problem, which ensures the NP-hardness of these problems. □

Remark. *The NP-hardness of these problems can also be established by reducing from the corresponding normal version, which has been shown to be NP-hard. In that case, the transformation from the normal problems to the non-negative ones is as follows :*

$$\tilde{\mathbf{y}} = \mathbf{y}, \quad \tilde{A} = [A, -A], \quad \tilde{\mathbf{x}} = \begin{bmatrix} \mathbf{x}^+ \\ \mathbf{x}^- \end{bmatrix} \tag{4}$$

where $\mathbf{x}^+ = \max\{\mathbf{x}, \mathbf{0}\}$ *and* $\mathbf{x}^- = \max\{-\mathbf{x}, \mathbf{0}\}$ *which ensures that* $\tilde{\mathbf{x}} \geq \mathbf{0}$.

Similar to what we've done in Section 3.1, we obtain the following theorem :

Theorem 4. *The following problem (ℓ_p-ℓ_0P+) :*

$$\min_{x \geq 0} \|\boldsymbol{y} - A\boldsymbol{x}\|_p^p + \lambda \|x\|_0 \tag{ℓ_p-ℓ_0P+}$$

is NP-hard for $p \geq 1$ and $0 < \lambda < 3$.

4 Extending to the case $\lambda \geq 3$

As we have seen previously, the Theorem 1 only deals with the case when $0 < \lambda < 3$, and [1] says that whether ($\ell_0 P$) is NP-hard for $\lambda \geq 3$ is still an open problem. However, under the guidance of the proof of Theorem 1, we can look for an extension in the case when $\lambda \geq 3$.

We first note that the constant 3 comes from the fact that ($\ell_0 P$) is reduced from the X3C problem. A straight forward intuition is that if we have the NP-completeness of the XkC (Exact cover by k-set) problem for $k \in \mathbb{N}, k \geq 3$, then we can show the NP-hardness of the problem ($\ell_0 P$) for $0 < \lambda < k$. This, once again, provokes the k-SAT problem which (might be) known as NP-complete.

However, we realize that in order to provide the solution for the case $\lambda > 3$, it is sufficient to show that the XkC problem is NP-complete for certains $k = k_1, k_2, \ldots$ such that $k_n \to +\infty$. This idea leads us to the following lemma that only makes use of the X3C problem :

Lemma 1. *The exact cover by k-set (XkC) problem is NP-complete for all $k = 3^n, n \in \mathbb{N}$.*

Proof. The proof is conducted by induction. For $n = 1$, we know that X3C is NP-complete.

Now suppose that XkC is NP-complete for $k = 3^n$, we'll show that X(3k)C is also NP-complete.

First, note that we can always check if a collection of subsets is an exact cover in polynomial time, we deduce that X(3k)C is in NP. We'll now show that X(3k)C is NP-hard, which then guarantees that it is NP-complete.

Now, let us introduce the problem XkC′ as follows: Given a set X such that $|X| = 3kq$ and a collection C of k-element subsets of X. Does C contains an exact cover of X, *i.e.* a subcollection $C' \subseteq C$ such that C' exactly covers X ?

We'll show that XkC′ can be reduced from XkC, and X(3k)C can be reduced from XkC′, which ensures the NP-hardness of X(3k)C. Firstly, the reduction from XkC to XkC′ can be performed as follows: For a collection C of $X = \{1, 2, \ldots, N\}$ we generate an instance of XkC′ with $X' = \{1, 2, \ldots, 3N\}$ and $C' = C \cup (C + N) \cup (C + 2N)$ where $C + j = \{S + j : S \in C\}$. Then, we can easily show that a solution of XkC implies a solution of XkC′ and vice-versa. Thus XkC′ is reduced from XkC. Now, let the set X be such that $|X|$ is a multiple of $3k$, and C is a collection of k-element subsets of X. We now define an instance of X(3k)C with the same set X and the collection $C^* = \{S_1 \cup S_2 \cup S_3 | S_1, S_2, S_3 \in C \text{ pairwise different }\}$. We see that if C contains an exact cover of X, then as the number of elements in X is divisible by $3k$, the number of subsets that exactly covers X must be divisible by 3, so C^* must contain an exact cover of X. Inversely, if C^* contains an exact cover of X, then obviously C contains an exact cover of X. This concludes the reducibility of X(3k)C and finally we deduce that X(3k)C is NP-hard, which then completes our proof. □

From the above Lemma, we finally can extend all the above theorems to the following theorem :

Theorem 5. *The problems ($\ell_0 P$) and ($\ell_0 P+$) are NP-hard for $\lambda > 0$. The problems (ℓ_p-$\ell_0 P$) and (ℓ_p-$\ell_0 P+$) are NP-hard for $p \geq 1,\ \lambda > 0$.*

5 Some approximation algorithms discussions

As ($\ell_0 P$) is an NP-hard problem one's question would be how to figure out a way to approximate the solution. One may think that finding a solution must be as easy as finding a solution for X3C. Unfortunately, even though we assume that all polynomial time is accepted, it will remain to verify the equivalence for every instance (e.g for $y \notin \mathbb{N}^m$). In this section, we will discuss the complexity and efficiency for some algorithms for the X3C (polynomial-time reduction of ($\ell_0 P$) in the given instance above) as well as the ($\ell_0 P$) in a general case.

- There are many possible ways used in practice to solve the X3C problem. As the X3C is NP-complete, we have that ($\ell_0 P$) is at least as hard as the X3C is. We can easily deduce a boundary for the instance defined in theorem 1 proof which is equal equivalent at the infinity to the polynomial instance construction time added to the X3C problem complexity time. There are many approximations for X3C namely :
  1. A greedy algorithm with a logarithmic efficiency ratio $H(n) \sim \ln(n)$.
  2. As an integer linear problem, the cutting methods, as well as brunch and bounds, are other methods used into approximation.

  The ratio is generally decaying very slowly. However, this ratio is tending to 0 as the instance size goes to the infinity.
- As long as the NP=P is out there waiting for answers, we would not be able to be more precise on whether a polynomial time is a boundary in general cases or not. With this being said, we suggest the following randomized algorithm for the ($\ell_0 P$) solution approximation :
  1. Choose uniformly M distinct subsets S_i of $[1, ..., n] = [n]$, $1 \leq i \leq M$
  2. For i in range $[1, ..., M]$: Solve ($\ell_0 P$) such that $x_j = 0$ if $j \in S_i$, and $\|x\|_0 = |S_i|$, and get the solution x_i^* and its associated value $f(x_i^*)$
  3. Return the x_i^* with the smallest value $f(x_i^*)$

  The algorithm above would be very efficient with to respect to quantile order given by f : That means, the probability that the solution x^* given by the algorithm being better than a proportion $1 - \alpha$ (i.e $|\{x_i^*, \text{s.t } s_i \subset [n], f(x_i^*) \leq f(x^*)\}| \geq |\{s, s \subset [n]\}| \times (1 - \alpha)$) is at least $1 - (1 - \alpha)^M$.

6 Conclusions

In this work, we have revisited the previous complexity analyses of ℓ_0 minimization problems. We must recall once more that ($\ell_0 P$) is defined $\forall \lambda$. This means the goal of article [1] is to prove that for all fixed λ, ($\ell_0 P$) is NP-hard. Indeed, if λ is an input of the problem that would lead into an equivalence between $(\ell_0 P)$ and $(\ell_0 C)$ or $(\ell_0 C')$ for a given value of λ (by Lagrange transformation) and the NP-hardness would be trivial. The authors managed to find fatal defects in ℓ_0 minimization literature and to give a very easy scheme to demonstrate the results included in section 3. However, they could not extend their results to more general cases. Thus we presented our attempt to solve the problem as detailed in section 4, which is our main theoretical result: the extension stems from lemma 1 of section 4. Finally, in section 5, and as part of our algorithmic exploration, we proposed a randomized algorithm that is quantile efficient with high probabilities.

If our results do not include many mistakes, a natural following step would be examining the link between a problem T_1 being NP-hard and a T_2, consisting of a generalization (by Lagrange duality) of T_1, being NP-hard as well. Aside from pure theoretical exploration, finding a heuristic that is efficient regarding the problem value and not the quantile order would make it perfect.

References

[1] Thi Thanh Nguyen, Charles Soussen, Jérôme Idier, and El-Hadi Djermoune. NP-hardness of $\ell 0$ minimization problems: revision and extension to the non-negative setting. In *13th International Conference on Sampling Theory and Applications, SampTA 2019*, 13th International Conference on Sampling Theory and Applications, SampTA 2019, Bordeaux, France, July 2019.

[2] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness (Series of Books in the Mathematical Sciences)*. W. H. Freeman, first edition edition, 1979.

[3] Stephen A. Cook. The complexity of theorem-proving procedures. In *IN STOC*, pages 151–158. ACM, 1971.

[4] Guillaume Lecué. Introduction au Compressed sensing. Notions de complexité algorithmique et relaxation convexe. https://lecueguillaume.github.io/assets/lecture_ell_0_cs.pdf, 2015.

[5] B. K. Natarajan. Sparse approximate solutions to linear systems. *SIAM Journal on Computing*, 24(2):227–234, 1995.

[6] G. Davis, Stéphane Mallat, and Marco Avellaneda. Greedy adaptive approximation. *Constr Approx*, 13:57–98, 03 1997.

[7] Xiaojun Chen, Dongdong Ge, Zizhuo Wang, and Yinyu Ye. Complexity of unconstrained l2-lp minimization. February 2014.

[8] Xiaoming Huo and Jie Chen. Complexity of penalized likelihood estimation. *Journal of Statistical Computation Simulation*, 21:1–16, 01 2008.