Constrained L2-L0 optimization and its application to Single-Molecule

Arne Bechensteen

UNIVERSITÉ CÔTE D'AZUR

ÉCOLE DOCTORALE
SCIENCES ET TECHNOLOGIES DE L'INFORMATION ET DE LA COMMUNICATION

THÈSE DE DOCTORAT

Optimisation L_2-L_0 contrainte et application à la microscopie à molécule unique

Arne Henrik Bechensteen

INRIA Sophia Antipolis/Laboratoire I3S

Présentée en vue de l'obtention du grade de docteur en Automatique, Traitement du Signal et des Images
d'Université Côte d'Azur
Dirigée par : Laure Blanc-Féraud/ Sébastien Schaub
Co-encadrée par : Gilles Aubert
Soutenue le : 24 novembre 2020

Devant le jury, composé de :
Jean-François Aujol, Université de Bordeaux
Caroline Chaux, CNRS, Aix-Marseille Université
Christine De Mol, Université Libre de Bruxelle
Jérôme Idier, CNRS, Ecole Centrale de Nantes
Olivier Meste, Université Côte d'Azur

OPTIMISATION $\ell_2 - \ell_0$ CONTRAINTE ET APPLICATION À LA MICROSCOPIE À MOLÉCULE UNIQUE

JURY:
Reviewers
Jean-François Aujol, Professeur, Université de Bordeaux
Jérôme Idier, Directeur de recherche, CNRS, École Centrale de Nantes

Examiners
Caroline Chaux, Chargé de recherche, CNRS, Aix-Marseille Université
Christine De Mol, Professeur émérite, Université Libre de Bruxelle
Olivier Meste, Professeur, Université Côte d'Azur

Directors of the thesis
Laure Blanc-Féraud, Directeur de recherche, CNRS, Université Côte d'Azur
Sébastien Schaub, Ingénieur de recherche, CNRS, Université Côte d'Azur

Co-supervisor
Gilles Aubert, Professeur émérite, Université Côte d'Azur

RESUMÉ

L'optimisation parcimonieuse est cruciale dans la société d'aujourd'hui, car elle est utilisée dans de nombreux domaines, tels que le débruitage, la compression, l'apprentissage et la sélection de caractéristiques. Cependant, obtenir une bonne solution parcimonieuse d'un signal est un défi de calcul.

Cette thèse se concentre sur l'optimisation d'un terme des moindres carrés en norme ℓ_2 sous une contrainte de k-parcimonie sur la solution exprimée avec la pseudo-norme ℓ_0 (le problème $\ell_2 - \ell_0$ contraint). Nous étudions également la somme de la fonction de perte des moindres carrés et d'un terme de pénalité ℓ_0 (le problème $\ell_2 - \ell_0$ pénalisé). Les deux problèmes sont non convexes, non continus et NP-durs. Nous proposons trois nouvelles approches d'optimisation parcimonieuse.

Nous présentons d'abord une relaxation continue du problème contraint et présentons une méthode pour minimiser la relaxation proposée. Deuxièmement, nous reformulons la pseudo-norme ℓ_0 comme un problème de minimisation convexe. Ceci est fait en introduisant une variable auxiliaire, et nous présentons une reformulation exacte du problème contraint (CoBic) et pénalisé (PeBic). Enfin, nous présentons une méthode pour minimiser le produit du terme de fidélité des données et du terme de régularisation. Ce dernier est un travail de recherche en cours.

Nous appliquons les trois premières méthodes proposées (relaxation, CoBic et PeBic) à la microscopie par molécule unique. Les résultats des algorithmes proposés sont à l'état de l'art des méthodes basées sur la grille. De plus, fixer la constante de contrainte de parcimonie est généralement plus intuitif que fixer le paramètre de pénalité, ce qui rend l'approche contrainte attractive pour les applications.

Mots clés: Traitement d'images, Problème inverse, Parcimonie

ABSTRACT

Sparse optimization is crucial in today's society, as this is used in multiple domains, such as denoising, compression, machine learning. Sparse optimization is also vital in single-molecule localization microscopy, a microscopy method widely used in biology. However, obtaining a good sparse solution of a signal is computationally challenging.

This thesis focuses on sparse optimization in the form of minimizing the least square loss function under a k-sparse constraint with an ℓ_0 pseudo-norm (the constrained $\ell_2 - \ell_0$ problem). We also study the sum of the least square loss function and an ℓ_0 penalty term (the penalized $\ell_2 - \ell_0$ problem). Both problems are non-convex, non-continuous, and NP-hard. We propose three new approaches to sparse optimization.

We present first a continuous relaxation of the constrained problem and present a method to minimize the proposed relaxation. Secondly, we reformulate the ℓ_0 pseudo-norm as a convex minimization problem. This is done by introducing an auxiliary variable, and we present an exact biconvex reformulation of the constrained (CoBic) and penalized (PeBic) problems. Finally, we present a method to minimize the product of the data fidelity term and the regularization term. The latter is still an ongoing research work.

We apply the three proposed methods (relaxation, CoBic, and PeBic) to single-molecule localization microscopy and compare them with other commonly used algorithms in sparse optimization. The proposed algorithms' results are as good as the state-of-the-art in grid-based methods. Furthermore, fixing the sparsity constraint constant is usually more intuitive than fixing the penalty parameter, making the constraint approach attractive for applications.

Keywords: Image processing, sparse optimization, inverse problem, regularization, super-resolution

To my loving family.

PUBLICATIONS

[BBFA18a] Arne Bechensteen, Laure Blanc-Féraud, and Gilles Aubert. "Single molecule localization by $\ell_2 - \ell_0$ constrained optimization." In: *iTWIST'18, Paper-ID: 13, Marseille, France, November, 21-23, 2018*. 2018.

[BBFA18b] Arne Bechensteen, Laure Blanc-Féraud, and Gilles Aubert. "Towards a continuous relaxation of the ℓ_0 constrained problem." 2018.

[BBFA19] Arne Bechensteen, Laure Blanc-Féraud, and Gilles Aubert. "New methods for $\ell_2 - \ell_0$ minimization and their applications to 2D Single-Molecule Localization Microscopy." In: *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*. IEEE. 2019, pp. 1377–1381.

[BBFA20a] Arne Bechensteen, Laure Blanc-Féraud, and Gilles Aubert. "A continuous relaxation of the constrained L_2- L_0 problem." In: *To appear in Journal of Mathematical Imaging and Vision* (2020).

[BBFA20b] Arne Bechensteen, Laure Blanc-Féraud, and Gilles Aubert. "New $\ell_2 - \ell_0$ algorithm for single-molecule localization microscopy." In: *Biomedical optics express* 11.2 (2020), pp. 1153–1174.

"There are some things you can't share without ending up liking each other, and knocking out a twelve-foot mountain troll is one of them."

— Harry Potter and the Philosopher's Stone

ACKNOWLEDGEMENTS

All Ph.D. tips and tricks suggested waiting with the acknowledgments till after all the rest is finished. I'm not following this suggestion, mainly because, while writing, I am overcome with gratitude because I would not have been here without my friends and colleagues.

First and foremost, I would thank Laure Blanc-Féraud, who first accepted me for an internship and then a thesis. Thank you for trusting me, for all the fruitful discussions, and for all the help. I know that my line of thought is not always easy to follow when we discuss, but you always take the time to figure out what I'm saying.

Many thanks to Gilles Aubert! Without you, I would still be stuck with the problem of sorting the vector. Thank you for all the Wednesdays and Fridays discussion. Thank you for teaching me (and I know I still have a lot to learn) the rigorous way of proofs. Furthermore, thank you for all good tips for hiking and running.

Thank you, Sébastien Schaub, for introducing me to the fascinating world of microscopy.

Furthermore, I would like to thank everyone in the Morpheme team and my fellow Ph.D students (for the quote above is for). Special thanks to Anca, Cyprien, Sarah, Gael, Diana, Somia, Cédric, Vasilina, Bastien, and many more at Morpeheme and Mediacoding. We may not have knocked a troll out together, but I feel writing a thesis is on the same level of difficulty and creates equally strong bonds.

Even though a thesis is hard work, I would not be able to finish it without having my friends outside of work, Doriane, Jérémy, and Clément. Thanks for all the great moments together. Furthermore, I would like to the sports group Groupe Azur Inter Sport for all the great activities. A special shout-out to Ronnie, my amazing swimming coach!

Although I did my thesis in Nice, I would like to thank my Norwegian friends back home, Harald and Lene, Elizabeth, Yan, Linn and Ellen, and many more.

I would not have been here without some fantastic teachers I have had throughout the years. Thanks to: Pierre, Alain, Claude, Geir, and Anita.

My family has always been there for me and I want to thank them from the bottom of my heart.

Finally, thank you, Clément, for all the love and support.

CONTENTS

LIST OF TABLES

ACRONYMS

BP Basis Pursuit

BPDN BP De-Noising

CEL0 Continuous Exact ℓ_0

C-IHT Constrained Iterative Hard Thresholding

CoBic Constrained Biconvex method

DC Difference of Convex

FBS Forward-Backward Splitting

FISTA Fast Iterative Shrinkage-Thresholding Algorithm

FPALM Fluorescence photoactivation localization microscopy

FWHM Full Width at Half Maximum

GMC Generalized Minimax-Concave

IHT Iterative Hard Thresholding

K-L Kurdyka-Łojasiewicz

LASSO Least Absolute Shrinkage And Selection Operator

MCP Minimax Concave Penalty

MIP Mixed Integer Problem

MP Matching Pursuit

nmAPG Nonmonotone Accelerated Proximal gradient algorithm

OMP Orthogonal Matching Pursuit

OLS Orthogonal Least Squares

PALM Photoactivated Localization Microscopy

PAM Proximal Alternating Minimization

PeBic Penalized Biconvex method

P-IHT Penalized Iterative Hard Thresholding

PSF Point Spread Function

RIP Restricted Isometry Property

SBR Single Best Replacement

SMLM Single Molecule Localization microscopy

STORM Stochastic Optical Reconstruction Microscopy

$< \cdot, \cdot >$ Euclidean scalar product

$\#$ The cardinality of a set

$\chi_X(x)$ The indicator function, $\chi_X(x) = +\infty$ if $x \notin X$, and 0 if $x \in X$

$\mathbb{R}$ Real numbers

$\mathbb{R}_{>0}$ Real positive numbers

$\mathbb{R}_{\geqslant 0}$ Real non-negative numbers

$\mathbf{1}_X$ The function $\mathbf{1}_X(x) = 1$ if $x \notin X$, and 0 if $x \in X$

$\| \cdot \|_p$ The p-norm, defined as $\| x \|_p^p = \sum_i |x_i|^p$

$\| A \|_2 = \sigma(A)$ The matrix norm of A. Equal to the largest singular value of A

$\| \cdot \|$ The ℓ_2-norm if the norm is not defined

$\| \cdot \|_0$ The ℓ_0-pseudo norm defined by $\| x \|_0 = \#\{x_i, i = 1, \cdots N : x_i \neq 0\}$

$\sigma(A)$ The largest singular value of A

$|x|$ The vector x where the absolute value is applied to each element.

$A \preceq B$ $B - A$ is a positive semidefinite matrix, i.e., $x^T(B - A)x \geqslant 0 \ \forall x \in \mathbb{R}^N$

$A \in \mathbb{R}^{M \times N}$ A is a $M \times N$ matrix.

A_ω The restriction of A to the columns indexed by the elements of ω be denoted as $A_\omega = (a_{\omega[1]}, \ldots, a_{\omega[\#\omega]}) \in \mathbb{R}^{M \times \#\omega}$

a_i The ith column of A

e The vector $(1, 1, \ldots, 1)^T \in \mathbb{R}^N$

$P^{(y)} \in \mathbb{R}^{N \times N}$ A permutation matrix such that $P^{(y)}y = y^\downarrow$

$x \in \mathbb{R}^N$ x is vector of length N

$x^{\downarrow y}$ We denote the vector $x^{\downarrow y} = P^{(y)}x$

$x^\downarrow$ The vector $x^\downarrow \in \mathbb{R}^N$ is the vector x where its components are sorted by their magnitude i.e. $|x_1^\downarrow| \geqslant |x_2^\downarrow| \geqslant \cdots \geqslant |x_N^\downarrow|$.

A Lipschitz function A function f is said to be L-Lipschitz if $|f(x) - f(y)| \leqslant L \| x - y \|$. L is independent of x and y.

Part I

CONSTRAINED $\ell_2 - \ell_0$ OPTIMIZATION

"Numquam ponenda est pluralitas sine necessitate"

—William of Ockham

1 INTRODUCTION TO SPARSE OPTIMIZATION

Contents

1.1 INTRODUCTION TO INVERSE PROBLEMS

MRI-scanners, CT-scanners, modern telescopes microscopes, and many other instruments, solve inverse problems. They have in common that they cannot always observe the object of interest directly. A CT-scanner does not directly observe a patient's brain but measures the intensity of multiple X-rays taken from different perspectives around the head. With the measurements and solving an inverse problem, the CT-scanner can generate an image. Digital telescopes also use inverse problems to obtain clear and precise images. In acquiring an astronomical photo, the object of interest is both far away and does not emit much light. Therefore the sensor will be sensitive to light pollution, and the acquisition contains most likely noise. Removing noise is an inherent difficulty in inverse problems.

Even though the study and applications of inverse problems are relatively new, they have been present for a long since ancient Greece [Luc94; Ber60]: Aristotle solved the inverse problem of the form of a 3D object when projected on to a 2D surface. The object he studied was the earth, and the observation was the shadow the earth cast during lunar eclipses. He concluded that the earth must have a spherical shape.

Let us define the forward and the inverse problems mathematically. Let say we have a function $F \in \mathcal{L}(X, Y)$, where X and Y are some space. We can write the equation as:

$$d = F(x). \tag{1.1}$$

The forward problem is to find $d \in Y$ for a given $x \in X$. The inverse problem is to find x given d. In the case of CT-tomography, d is the set of measurements of the X-ray intensity, F is the function that models the intensity after an X-ray has passed through an object, and x is a slice of the brain.

F can model countless applications. Another well-known inverse problem is deblurring, where d is a blurred image, F is a blurring operator (more on this operator later), and x is the "clean" image.

The data the sensors capture are discrete. The sensors may be a CCD image sensor where each pixel detects a signal proportional to light intensity. Moreover, even though we live in a continuous world, we observe the CDD sensor's images on a screen made up of pixels. Thus we reconstruct often in the discrete setting. Furthermore, many inverse model applications use a function F, which can be linearized in the discrete setting. In a discrete setting, we can now write the model as

$$d = Ax \tag{1.2}$$

where A is a matrix in $\mathbb{R}^{M\times N}$ which performs as the linear function F, $x \in \mathbb{R}^N$ and $d \in \mathbb{R}^M$.

The world we live in is far from perfect, and so are the sensors that capture the data. Noise is introduced. Thus a more fitting acquisition model is

$$d = Ax + \eta \tag{1.3}$$

where $\eta \in \mathbb{R}^N$ is some small additive noise. Depending on the application, the noise can follow different statistics such as Gaussian, or it does not even need to be additive, like Poisson noise. Furthermore, the noise can be a mix of different statistics.

To obtain x, we search to maximize the likelihood to recover x, given the data d. This likelihood is equal to the conditional probability of d knowing x, denoted $P(d|x)$. Supposing η to be additive white Gaussian noise, to maximize the likelihood is equivalent to minimize the following term:

$$\underset{x\in\mathbb{R}^N}{\arg\min}\ \frac{1}{2}\|Ax - d\|_2^2. \tag{1.4}$$

The above term is referred to as the data fidelity term or the ℓ_2-term.

In the case of Poisson noise, we have to minimize

$$\arg\min_{x\in\mathbb{R}^N} \sum_i [(Ax+\eta)_i - d_i \ln((Ax+\eta)_i)]\,. \tag{1.5}$$

In many cases, the ℓ_2 data fidelity term (1.4) is used, mostly because it is easy to analyze and has an intuitive form.

1.1.1 *Ill-posed problems*

Hadamard presented in 1902 the term well-posed problems. They have the three following properties:

- a solution exists,
- the solution is unique,
- the solution's behavior changes continuously with the initial conditions.

The term "well-posed" was meant to guide researches to make good mathematical models for forward-problems. Unfortunately, most inverse problems are ill-posed. In the presence of noise, the behavior of the solution explodes. This can be shown quite easily. The optimality condition for problem (1.4) is

$$0 = \nabla\left(\frac{1}{2}\|A\hat{x}-d\|^2\right) \Leftrightarrow$$

$$0 = A^TA\hat{x} - A^Td \Leftrightarrow \hat{x} = (A^TA)^{-1}A^Td.$$

Replacing d with (1.3), and we obtain

$$\hat{x} = (A^TA)^{-1}A^T(A\hat{x}_{exact}+\eta) \leftrightarrow \hat{x} = \hat{x}_{exact} + (A^TA)^{-1}A^T\eta$$

If A have small non-zero singular values, the inversion of A^TA will lead to an explosion of noise such that $\hat{x}$ will be far from $\hat{x}_{exact}$.

1.1.2 *Regularization*

The solution of (1.4) is not acceptable in the presence of noise as the problem is most likely ill-posed. *A priori* information of the signal is used to avoid this problem, i.e., we impose a priori some properties on the sought solution. A great example of *a priori* information is total variation [AV94], which is used in denoising. Total variation is designed to promote smoothness in the image while preserving edges. This is natural in images. Just look out of the window; there is a sharp edge between the window frame and the blue sky, and the blue sky is smooth. Total variation was quite revolutionary at the time, as average filters were mostly used in denoising, but these filters

smoothened the edges. With time, variations of TV regularizations has been proposed and is still used in many denoising algorithms [Pad+20].

With the a priori information, denoted $R(x) : \mathbb{R}^N \to \mathbb{R}$, an optimization problem can be written as one of these three

$$\arg\min_{x\in\mathbb{R}^N} \frac{1}{2}\|Ax-d\|_2^2 \text{ s.t. } R(x) \leqslant k, \tag{1.6}$$

$$\arg\min_{x\in\mathbb{R}^N} R(x) \text{ s.t. } \|Ax-d\|_2^2 \leqslant \epsilon, \tag{1.7}$$

$$\arg\min_{x\in\mathbb{R}^N} \frac{1}{2}\|Ax-d\|_2^2 + \lambda R(x). \tag{1.8}$$

The choice of which form of minimization depends on the knowledge of the problem. The minimization problem (1.6) can be used if we have information on the boundedness, k, of the a priori information. Examples of this are presented later in this thesis.

(1.7) can be used when the user has information on the noise level, η, such that $\epsilon \approx \eta$. The most common one is the regularization method (1.8), as the $\lambda \in \mathbb{R}_{\geqslant 0}$ serves as a weight between the data fidelity term and the a priori information $R(x)$.

The choice of the regularization term depends on the application and should represent properties of the wished-for reconstructed signal.

1.2 PROBLEM FORMULATION

Sparse optimization is to reconstruct a signal with few non-zero components from an acquired observation. A natural way of choosing a way of measuring sparsity is by using the ℓ_0-pseudo-norm , which will, by abuse of terminology, referred to as the ℓ_0-norm:

$$\|x\|_0 = \#\{x_i, i = 1, \cdots N : x_i \neq 0\} \tag{1.9}$$

with #S defined as the cardinality of S. The ℓ_0-norm is only a pseudo-norm as it is invariant by multiplication, i.e., $\lambda \neq 0, \|\lambda x\|_0 = \|x\|_0$.

Following the previous section, we can write sparse optimization as one of the three following ways:

$$\arg\min_{x\in\mathbb{R}^N} G_k(x) := \frac{1}{2}\|Ax-d\|_2^2 \text{ s.t. } \|x\|_0 \leqslant k \tag{1.10}$$

$$\arg\min_{x\in\mathbb{R}^N} \|x\|_0 \text{ s.t. } \|Ax-d\|_2^2 \leqslant \epsilon \tag{1.11}$$

$$\arg\min_{x\in\mathbb{R}^N} G_{\ell_0}(x) := \frac{1}{2}\|Ax-d\|_2^2 + \lambda\|x\|_0 \tag{1.12}$$

We refer problem (1.10) to as the *constrained* $\ell_2 - \ell_0$ optimization problem. The problem (1.12) is referred to as the *penalized* $\ell_2 - \ell_0$ problem, and (1.11) is referred to as the constrained $\ell_0 - \ell_2$ problem. The different minimization problems can be applied depending on the knowledge of the data the user has beforehand. If the user has an idea of the solution's sparsity, then the constrained $\ell_2 - \ell_0$ formulation may be preferable to use. However, if the user has information on the level of noise the data has been corrupted by, then the constrained $\ell_0 - \ell_2$ formulation is advantageous. Lastly, if the user has no information about the solution's sparsity, neither the noise, then the penalized $\ell_2 - \ell_0$ formulation is preferable.

Due to the combinatorial nature of the ℓ_0-norm, the problems are non-convex. Thus, it is important to note that these three problems are similar, but not equivalent [Nik16].

The global minimum of (1.12) (respectively (1.10)) could be calculated as follows: For each possible combination of support of x (respectively, such that the cardinal of the support is less or equal than k), noted ω, we can minimize $\frac{1}{2}\|A_\omega x_\omega - d\|^2$ and take the minimum. However, it is expensive to investigate all the possible supports, as the number of calculations to do is 2^N for the penalized problem ($\sum_{i=0}^{N} \binom{N}{i} = 2^N$). In the *constrained* $\ell_2 - \ell_0$ case, supposing that the global minimum has an exact k-support, the possible number of supports to test is $\binom{N}{k}$. This could be calculated in small numerical examples, but in the second part of this thesis, we are working on images going up to 512×512. Say that $k = 3$, then the number of possible supports to search for are 3×10^{15}. Furthermore, k is much larger in the applications, and this method is not possible with standard computers.

The main focus of the thesis is centered around the constrained problem.

1.3 APPLICATIONS OF SPARSE OPTIMIZATION

Sparse optimization is found in many applications and in this section, we introduce problems where efficient sparse optimization is needed.

1.3.1 *Sparse deconvolution*

The acquisition model for a standard deconvolution problem is as follows

$$d = h * x + \eta,$$

where h is the impulse response of the system, η is the noise, and d the observed signal. x is the signal we want to reconstruct.

For some acquisitions, the user knows the solution to be sparse, i.e., the observed signal is a result of convolution to a few "spikes." Applications to this can be found everywhere. In geophysics, the spikes

are reflectivity, which indicates the transit of geological layers. A vibrational source is located at the surface and sends a wave down in the earth. Sensors observe waves reflected from the ground at different layers, which can be modeled as convolution with the source wavelet and the reflectivity [Men13].

Ultrasonic nondestructive evaluation in nuclear reactors uses sparse deconvolution to estimate the reactor's pressure tube life. Hydrogen can shorten the tubes' life, and a method to do so is to measure the percentage of a particular molecule in the tube. An ultrasonic pulse propagates in the material, and the echos captured by the sensors are from internal flaws [OSK94]. This can be modeled as a sparse optimization problem when we suppose that the internal flaws are a few "spikes."

In applications described above, we can use a regularization term that promotes sparsity, such as the ℓ_0-norm. Furthermore, assuming Gaussian noise, this can be described as sparse regularization problem such as (1.10), (1.11) or (1.12).

1.3.2 *Dictionary learning*

A sparse representation of a signal is to represent a "full" signal with few components. The search for sparse representations of signals grew as the data captured grew and grew. A signal with few components uses less space when stored, and thus sparse representation was initially inspired by data compression. What more, noise is by nature random, and sparse representation of something random is problematic. Thus a sparse representation of a signal can be used to denoising as well. Sparse signals are more prone to interpretation since only a few atoms contribute to the full signal. There are many methods to obtain sparse representations of signals, as using the Haar transformation [Haa10].

More recent methods search for a dictionary $D \in \mathbb{R}^{M\times N}$, such that a full signal d can be represented by Dx. where x is a sparse signal. The method is called dictionary learning. Using a learned dictionary in opposition to predefined dictionaries, such as wavelets, is more efficient in signal processing, such as denoising [EA06].

Dictionary learning aims to find sparse representations for a set of signals and the minimization problem can be written as

$$\min_{D,x} \|Dx - d\|^2 + R(x),$$

where R is a regularization term that promotes sparsity, such as penalized ($R(x) = \lambda\|x\|_0$) or constrained ($R(x) = \chi_{\|\cdot\|_0\leqslant k}$).

In order to perform the above minimization, alternating methods may be applied [Xu+17]. Alternating methods consist of minimizing

first with respect to one variable while fixing the other, then minimizing the other variable. Thus one step of the minimization is

$$\min_x \|Dx - d\|^2 + R(x).$$

Thus efficient sparse optimization is needed.

Dictionary learning can be applied to many image processing problems such as denoising, inpainting [She+09], and classification and face recognition [Xu+17].

1.3.3 *And much more!*

In the previous two sections, we have seen two classical optimization problems that apply to a large scale of problems. That is only a small part of possible applications where sparse optimization is needed. In this thesis, we limit the applications to Single-Molecule localization microscopy, which is deconvolution and upsampling problem, see Chapter 5.

1.4 STATE OF THE ART

Sparse optimization is a rapidly growing area of research. This section tries to give a small overview of existing methods and advances to solve these problems.

This section is divided into subsections, each for the different methods of optimization. We start with the well known ℓ_1-relaxation and introduce other relaxations and reformulations. We end the chapter by presenting algorithms designed to minimize the initial problems.

1.4.1 *Relaxations*

The ℓ_0-norm is non-convex and non-continuous. Thus it is interesting to replace the norm by another function.

1.4.1.1 *ℓ_1-relaxation*

A natural choice is the ℓ_1-norm, which is both convex and continuous. In fact, the ℓ_1-norm is the convex envelope of the ℓ_0-norm when restricted to the unit ball. The ℓ_1-relaxation was first introduced in the article [CD94] and is named Basis Pursuit (BP), but the ℓ_1-relaxation has a long history of promoting sparsity. Instead of finding a minimum of the following problem

$$\min_x \|x\|_0 \text{ s. t. } Ax = d, \tag{1.13}$$

they propose to solve the relaxed form

$$\min_x \|x\|_1 \text{ s. t. } Ax = d. \tag{1.14}$$

Simplex methods from linear programming, such as [CDS01], can solve problem (1.14).

The above problem does not take into account noise, and BP De-Noising (BPDN) [CDS01], and Least Absolute Shrinkage And Selection Operator (LASSO) [Tib96] propose algorithms to take noise into account. Thus the problem formulations are

$$\min_x \|x\|_1 \text{ s. t. } \|Ax - d\| \leqslant \eta, \tag{1.15}$$

and

$$\min_x \frac{1}{2}\|Ax - d\|^2 + \lambda\|x\|_1. \tag{1.16}$$

There is an extensive literature concerning equivalence between the minimizer of the ℓ_1-and the ℓ_0 problem. Among others, the Restricted Isometry Property (RIP) is used as a criterion.

Definition 1.1. *Let A be an $M \times N$ matrix and let $1 \leqslant k \leqslant N$ be an integer. Suppose that there exists a constant $\delta_s \in (0,1)$ such that, for every $M \times k$ submatrix A_k of A and for every k-dimensional vector y*

$$(1-\delta_k)\|y\|_2^2 \leqslant \|A_k y\|_2^2 \leqslant (1+\delta_k)\|y\|_2^2.$$

Then, the matrix A is said to satisfy the k-restricted isometry property with the Restricted Isometry Constant (RIC) δ_k, δ_k being the smallest value that satisfies the above inequalities.

If A verifies RIP of order $2k$ with constant $\delta_{2k} < 1$, then ℓ_0 problem has a unique k-sparse solution. Furthermore, in [Can+08], if $\delta_{2k} < \sqrt{2}-1$ then the minimum of the ℓ_1-relaxation is the same as the minimum of the initial function. This holds for the noiseless (1.14) and in the case with noise (1.15) and (1.16).

The RIP condition is hard to verify since it is needed to test all submatrices A_k, and other definitions such as Spark [DE03], Null Space Condition [CDD09], and Coherence has been used. This is very much used in Compressed Sensing, a topic we do not go further into in this thesis, but there are excellent source material on the subject, see for example [Dav+12]

The Forward-Backward Splitting (FBS) algorithm [CW05] is suitable to solve the penalized problem (1.16). The algorithm uses the proximal operator.

Definition 1.2. *The proximal operator of a function g is defined as*

$$prox_{\frac{g}{\gamma}}(y) = \arg\min_x g(x) + \frac{\gamma}{2}\|x-y\|_2^2 \tag{1.17}$$

The FBS is designed to work on problems on the form

$$\hat{x} \in \arg\min_x J(x) := f(x) + g(x) \tag{1.18}$$

where f is a differential function, ∇f is L-Lipschitz, and the proximal operator of g can be calculated.

The algorithm for solving problems of the form (1.18) is as follows:

Algorithm 1 : Forward-Backward Splitting

Input :

$\gamma \in [L/2, +\infty[$;

$x^0 \in \mathbb{R}^N$;

Repeat :

$x^{(p+1)} \in \text{prox}_{\frac{g}{\gamma}}(x^{(p)} - \frac{1}{\gamma}\nabla f(x^{(p)}))$;

Until : Convergence

Output : $x^{(p+1)}$

In the penalized case, we have that $f(x) = \frac{1}{2}\|Ax - d\|^2$ and $g(x) = \lambda\|x\|_1$ The proximal operator of the ℓ_1-norm is the shrinkage operator (also known as soft thresholding) defined as

$$\text{prox}_{\lambda\|\cdot\|_1}(y)_i = \begin{cases} y_i - \lambda \text{ if } y_i \geqslant \lambda \\ y_i + \lambda \text{ if } y_i \leqslant -\lambda \\ 0 \text{ if } y_i \in [-\lambda, \lambda]. \end{cases}$$

Accelerations of the above algorithm have been proposed, such as the Fast Iterative Shrinkage-Thresholding Algorithm (FISTA) [BT09].

We have seen that ℓ_1-relaxation has the same minimizer as the initial problem under certain conditions. However, if these strict conditions are not verified, we have no assurance of the equivalence. Furthermore, contrary to the ℓ_0-norm, the ℓ_1 norm penalizes large coefficients. Adaptive methods of the penalized problem were introduced in [CWB08; Zou06] to increase the accuracy of the ℓ_1-minimization.

They propose

$$\|x\|_0 \approx \sum_i w_i |x_i|,$$

where w_i adapts from x_i, and is inversely proportional to x_i. For example, the algorithm presented in [CWB08] adapts for each step of its algorithm w_i such that $w_i^{(p+1)} = \frac{1}{|x_i^{(p)}| + \delta}$, (p) being the pth iteration of the algorithm, and δ a small constant. Thus elements with a large magnitude will be less penalized.

Similarly, but with another approach, [ZK10; Hug+17], rewrites the ℓ_0-norm as

$$\|x\|_0 = \sum_i \frac{x_i^2}{x_i^2} \approx \frac{x_i^2}{\tilde{x_i}^2}$$

where $\tilde{x}_i$ is an approximation of x_i. Thus the problem can be written as

$$\frac{1}{2}\|Ax - d\|^2 + \sum_i w_i x_i^2,$$

where w_i is updated at each iteration.

Most of the research seems to focuse on the penalized $\ell_2 - \ell_1$ or the constrained $\ell_1 - \ell_2$ problem.

1.4.1.2 *Non-convex relaxations*

Non-convex regularizers avoids penalizing the magnitude x, such as the ℓ_1-norm does. The idea is to use a function that is "similar" to the ℓ_0-norm. Many regularizers are separable, i.e.,

$$R(x) := \sum_i \phi(x_i). \tag{1.19}$$

There exist non-separable regularizers, which are presented later in this section. First, the capped-ℓ_1 penalty does not penalize large values by using

$$\phi(x_i) = \lambda \min(\theta|x|, 1),$$

and the method is still used with success [Wan+19].

The ℓ_p-pseudo-norm, with $0 < p < 1$, was studied in [Cha07; CXY10], first with the observation matrix A verifying the RIP condition, and later more generally. Under certain conditions, using the RIP condition, this relaxation is exact [Cha07]. We refer the ℓ_p pseudo-norm, by abuse of terminology, as the ℓ_p-norm. The penalty term is

$$R(x) = \|x\|_p^p = \sum_i |x_i|^p.$$

The authors state that the ℓ_p-norm can be seen as intermediate between the ℓ_0-norm and the ℓ_1-norm as the ℓ_0-norm can be written as

$$\|x\|_0 = \sum_{\substack{i \\ x_i \neq 0}} |x_i|^0.$$

More generally, [FL01] introduced some conditions to define what a "good" relaxation of the ℓ_0-norm is. First of all, a regularizer should be *unbiased* in order not to penalize large values. Secondly, the regularizer should *promote sparsity*, with a thresholding rule to set small values to zero. At last, they state that a regularizer should be *continuous* to avoid instability. They proposed the Smoothly clipped absolute deviation [FL01].

In the same idea, [Cho+13] studied regularizers that approach the ℓ_0-norm asymptotically. The relaxations are on the form

$$R(\delta; x) = \sum_i \phi(\delta; x_i),$$

where

$$\lim_{\substack{\delta \to 0 \\ \delta > 0}} \phi(\delta; x) = |x|_0.$$

There are many examples of relaxations in this form. In the article they give, among other, the example of Truncated quadratic potential [Vekoo]:

$$\phi(\delta, x) = \min(\frac{x^2}{2\delta^2}, 1).$$

Their main results state that a minimizer of the initial problem G_{ℓ_0} can be approximated by choosing a small enough δ.

An important relaxation that is also a source of inspiration for this thesis is the Continuous Exact ℓ_0 (CEL0) penalty, introduced in [SBFA15]. The authors calculate, using the Legendre transformation, the convex envelope of the penalized $\ell_2 - \ell_0$ function in one dimension.

$$G_{\ell_0}(x) = \frac{1}{2}(ax - d)^2 + \lambda \|x\|_0$$

The convex envelope of the above equation is

$$G^{**}_{\ell_0} = \frac{1}{2}(ax - d)^2 + \lambda - \left(\max \left\{ \sqrt{\lambda} - \frac{|a||x|}{\sqrt{2}}, 0 \right\} \right)^2.$$

To expand a higher dimension, the authors suppose that $A^T A$ is a diagonal matrix and calculates the initial problem's convex envelope. Since $A^T A$ is a diagonal matrix, the problem becomes a sum of one-dimensional problems. For $A \in \mathbb{R}^{M \times N}$ and $x \in \mathbb{R}^N$, he obtains:

$$G^{**}_{\ell_0} = \frac{1}{2}\|Ax - d\|^2 + \sum_{i=1}^{N} \phi(\|a_i\|, \lambda; x_i)$$

where $\|a_i\|_2$ is the ℓ_2 norm of the ith column of the matrix A and

$$\phi(\|a_i\|, \lambda; x_i) = \lambda - \frac{\|a_i\|^2}{2} \left(|x_i| - \frac{\sqrt{2\lambda}}{\|a_i\|} \right)^2 \mathbf{1}_{|x_i| \leqslant \frac{\sqrt{2\lambda}}{\|a_i\|}}. \tag{1.20}$$

They define the penalty term

$$\Phi_{CEL0}(A, \lambda; x) = \sum_{i=1}^{N} \phi(\|a_i\|, \lambda; x_i). \tag{1.21}$$

Further, they assume that A is not necessarily orthogonal, and define G_{CEL0}.

$$G_{CEL0}(x) = \frac{1}{2}\|Ax - d\|^2 + \Phi_{CEL0}(A, \lambda; x). \tag{1.22}$$

The authors propose to minimize G_{CEL0} rather than G_{ℓ_0}. The function G_{CEL0} is not convex, but continuous. They prove three important properties concerning the connection between the minimizers of the G_{CEL0} and G_{ℓ_0}.

- The global minimizers of G_{CEL0} contains the global minimizers of G_{ℓ_0}.
- From each minimizers (global or local) of G_{CEL0} one can identify a minimiser (global or local) of G_{ℓ_0}.
- Some local minimizers of G_{ℓ_0} are not local minimizers of G_{CEL0}.

G_{CEL0} can eliminate some local minimizers while preserving the global ones. Furthermore, G_{CEL0} is continuous while G_{ℓ_0} is not. Thus, minimizing G_{CEL0} may lead to better results than G_{ℓ_0}.

Following their work, [SBFA17] gives strong conditions for relaxations of the ℓ_0-norm for the penalized formulation. The authors were interested in exact relaxations. For a relaxation to be exact, it has to verify the two following conditions

- The relaxation has the same global minimizers as the initial function.
- The relaxation does not add any local minimizers to the problem.

As seen, the ℓ_1-norm is an exact relaxation, but only under strict assumptions on A. In [SBFA17], they work without any assumptions on A.

The authors prove that relaxations such as Capped-ℓ_1 conserve the global minimizers for certain choices of θ, but it does not verify the second condition of exactness. The same holds for SCAD. The well-known Minimax Concave Penalty (MCP) [Zha+10] verifies the two conditions if the parameters are chosen correctly.

To end this section, we give an example of non-separable relaxation. These are relaxations that cannot be written as (1.19). [Sel17] proposes a non-convex regularization, Generalized Minimax-Concave (GMC) Penalty. The penalty is based on the Generalized Huber function, defined below.

Definition 1.3. *Let* $B \in \mathbb{R}^{M\times N}$. *The Generalized Huber function* $S_B : \mathbb{R}^N \to \mathbb{R}$ *is defined as*

$$S_B(x) = \inf_{v\in\mathbb{R}^N} \|v\|_1 + \frac{1}{2}\|B(x-v)\|_2^2$$

The GMC penalty is defined as

$$\Psi_B(x) = \|x\|_1 - S_B(x)$$

The author proposes the following cost function

$$\frac{1}{2}\|Ax-d\|^2 + \lambda\Psi_B(x).$$

They show that when B is chosen such that $\lambda B^TB \preceq A^TA$, then the above cost function is convex.

It is important to note that we are not aware of any non-convex relaxations of the constrained $\ell_2 - \ell_0$ problem (1.10).

1.4.2 *Reformulations*

In this section, we choose to include Difference of Convex (DC) programming. This could be seen as a method of minimizing approximations of the ℓ_0-norm, but it has strong connections to reformulations of the ℓ_0-norm as well. The idea is to write an approximation or a relaxation of the ℓ_0-norm as the difference of two convex functions. In [LT+15], they study the exactness of the relaxations.

The authors of [GTT18] propose to rewrite the k-sparse constraint as a difference of two convex functions. They states the equivalence between $\|x\|_0 \leqslant k$ and $\|x\|_1 - |||x|||_k = 0$. $|||\cdot|||_k$ is defined as $\sum_{i=1}^{k} |x_i^{\downarrow}|$, where $x^{\downarrow}$ is such that $|x_1^{\downarrow}| \geqslant |x_2^{\downarrow}| \geqslant \dots |x_N^{\downarrow}|$. Further, the authors relax the constraints, and proposes to minimize the following expression using DC methods:

$$\frac{1}{2}\|Ax - d\|_2^2 + \rho(\|x\|_1 - |||x|||_k). \tag{1.23}$$

They prove that the algorithm converges to a critical point of the initial constrained $\ell_2 - \ell_0$ function (1.10) as long as $\rho > \|A^T d\|_2 + \frac{3}{2}LC$, with L being the L-Lipschitz gradient of the ℓ_2 data fidelity term, and C is a value that bounds, by the ℓ_2-norm, the optimal argument $\hat{x}$ of (1.23) $\forall\rho$.

In the article [Asp03], they reformulated the ℓ_0-norm by introducing an auxiliary variable. The author proposes:

$$\|x\|_0 = \min_{\nu} \sum_i \nu_i \text{ s.t. } (\nu_i - 1)x_i = 0 \text{ and } \nu_i \geqslant 0 \ \forall i.$$

They suggest to use the above formulation to solve the constrained $\ell_0 - \ell_2$ problem (1.11). In the same idea, [Bou+16] reformulates the constrained problem as a Mixed Integer Problem (MIP). They propose to use the reformulation for the all three formulations (1.10), (1.11), and (1.12). The authors suppose that the solution $\hat{x}$ to the problem can be bounded in the infinity norm by M.

$$\|x\|_0 \leqslant k \Leftrightarrow \begin{cases} \exists b \in \{0,1\}^N \\ \text{s.t} \\ \sum_i^N b_i \leqslant k \\ -Mb \leqslant x \leqslant MB. \end{cases}$$

Furthermore, they use an algorithm that finds a global minimum. However, the algorithm is limited to small problems; when, $x \in \mathbb{R}^N$, $N \approx 100$, the algorithm may use more than 1000 seconds to find the solution.

Other methods offer different reformulations of the ℓ_0-norm. The work of Liu and Bi [BLP14; LBP18] focuses on a reformulation of ℓ_0-norm on the following form:

$$\|x\|_0 = \min_{0 \leqslant v \leqslant e} < e, e - v > \text{ s.t. } < v, |x| >= 0, \tag{1.24}$$

where $\mathbf{0} \leqslant \mathbf{v} \leqslant e$ is a component-wise comparison, and $|x|$ is the vector $x \in \mathbb{R}^N$ where the absolute value is applied to each component. This reformulation is similar to one proposed later by [YG16]:

$$\|x\|_0 = \min_{\substack{-\mathbf{1} \leqslant u \leqslant \mathbf{1} \\ u \in \mathbb{R}^N}} \|u\|_1 \text{ s.t } \|x\|_1 =< u, x > . \tag{1.25}$$

Let $u = e - v$, and use Eq. (1.24):

$$\|x\|_0 = \min_{\mathbf{0} \leqslant \mathbf{u} \leqslant \mathbf{1}} < e, u > \text{ s.t. } < e - u, |x| >= 0 \tag{1.26}$$

$$\Leftrightarrow \|x\|_0 = \min_{\mathbf{0} \leqslant \mathbf{u} \leqslant \mathbf{1}} \sum u_i \text{ s.t. } \|x\|_1 - < u, |x| >= 0 \tag{1.27}$$

We observe that (1.25) is similar but not equivalent to the one above. However, [BLP14; LBP18] focus mainly on the following minimization problem

$$\min_x \|x\|_0 \text{ s.t. } \|Ax - d\| \leqslant \delta.$$

Note that their data fidelity term is not squared, compared to the problem (1.11).

1.4.3 *Algorithms*

We have in the previous sections presented different methods to facilitate solving the initial ℓ_0-problems (1.10), (1.11), and (1.12) by relaxing or reformulating them. However, some algorithms minimize the initial problem. We enumerate some of it below.

The Iterative Hard Thresholding (IHT) was first introduced in [BD08]. It is an algorithm that, under the assumption that $\|A\|_2 < 1$, converges to a critical point. The algorithm can be viewed as an FBS (see Algorithm 1), with a step size $\gamma = 1$. The FBS algorithm's convergence was later ensured for a non-convex function J that satisfies the Kurdyka-Łojasiewicz (K-L) property, see [ABS13]. The steps size γ should be in $\gamma \in [L, +\infty[$ where L is the Lipschitz constant of ∇f.

Definition 1.4. *A locally Lipshitz function $f : \mathbb{R}^N \to \mathbb{R}$ satisfies the Kurdyka-Łojasiewicz inequality at $x^* \in \mathbb{R}^N$ iff there exists $0 < \eta < \infty$, a neighborhood U of x^*, and a concave function $\kappa : [0, \eta] \to [0, +\infty[$ such that*

1. $\kappa(0) = 0$,
2. *κ is of class C^1 on $]0, \eta[$,*
3. *$\kappa' > 0$ on $]0, \eta[$,*
4. *For every $x \in U$ with $f(x^*) < f(x) < f(x^*) + \eta$ we have*

$$\kappa'(f(x) - f(x^*)) \operatorname{dist}(0, \partial^L(f(x)) \geqslant 1$$

The ∂^L denotes the limiting-subdifferential [Moro6]. Further note that a large number of functions satisfy the K-L property, including $\|x\|_0$ and $\chi_{\|\cdot\|_0 \leqslant k}(x)$ [ABS13].

The hard thresholding is simply the proximal operator of the regularization term. In the case of the penalized formulation, the proximal operator is defined as

$$\text{prox}_{\lambda\|\cdot\|_0}(y)_i = \begin{cases} 0 \text{ if } |y_i| < \sqrt{2\lambda} \\ \{0, y_i\} \text{ if } |y_i| = \sqrt{2\lambda} \\ y_i \text{ else.} \end{cases}$$

In the constrained case, the proximal operator is

$$\text{prox}^{\downarrow}_{\chi_{\|\cdot\|_0 \leqslant k}}(y) = \begin{cases} y_i^{\downarrow} \text{ if } i \leqslant k \\ 0 \text{ else.} \end{cases} \tag{1.28}$$

The operator $x^{\downarrow}$ sort the elements from their magnitude, i.e., $|x_1^{\downarrow}| \geqslant |x_2^{\downarrow}| \geqslant \cdots \geqslant |x_k^{\downarrow}|$. Thus, the proximal operator keeps only the k largest elements by their magnitude.

Several greedy algorithms are designed to obtain sparse solutions. The idea behind greedy algorithms is to start with a zero initialization, and for each iteration, add an element to x just until a convergence criterium. A generic algorithm is shown in Algorithm 2.

Algorithm 2 : Standard Greedy algorithm

Input :

$A \in \mathbb{R}^{M \times N}$, $d \in \mathbb{R}^M$;

$x^{(0)} \in 0^N$, $R^{(0)} = d$;

Repeat :

Choose a subset i; $i \in Res(A, R^{(p)})$
Merge the support: $\omega^{(p+1)} = NewSupport(\omega^{(p)}, A, i)$
Update $x^{(p+1)} = newX(A, \omega^{(p)}, x^{(p)})$
Update residue:
$R^{(p+1)} = NewResidue(R^{(p)}, A, x^{(p+1)}, \omega^{(p+1)})$

Until : Convergence

Output : x^n

The Matching Pursuit (MP) algorithm was proposed in [MZ93]. The algorithm is designed to choose one support, i, at each iteration. The support is chosen to minimize the residual, where the residual is defined as $R^{(p)} = d - Ax^{(p)}$.

Let A be normalized, i.e., $\|a_i\| = 1$, then we search for an index i, such that the following problem has the smallest value:

$$\min_{\beta} \|R^{(p)} - \beta A_i\|^2$$

First, we observe that for a given i, β must be equal to $< A_i^T, R >$. Then, by developing the above expression, we have

$$\|R^{(p)}\|^2 - < A_i^T, R >$$

Thus, we search i such that $< A_i^T, R >$ is maximal. x is updated by $x^{(p+1)} = x^{(p)} + < A_i^T, R > e_i$. The algorithm convergences either when the residual is sufficiently small or when the sparsity constraint is saturated.

The Orthogonal Matching Pursuit (OMP) [PRK93] is identical in the choice of support; however, x and the residual is updated differently. Given at iteration $p+1$, we note $\omega^{(p+1)}$ the support obtained after $p+1$ iterations. Contrary to MP which only adds one value to the vector x, OMP additionally optimizes the value of x_ω based on the data fidelity term. That yields

$$x^{p+1} \in \arg\min \|Ax - d\|^2 \text{ s.t. } x_i = 0 \quad \forall i \notin \omega.$$

The residual is updated $R^{(p+1)} = d - Ax^{(p+1)}$.

Another cornerstone of sparse greedy algorithms is the Orthogonal Least Squares (OLS) [CBL89]. The algorithm differs from MP and OMP in the choice of support. While MP and OMP choose the support from the residual, OLS searches for the component that decreases the data fidelity term.

$$i \in \arg\min_j \|A_{\omega \cup \{j\}} x_{\omega \cup \{j\}} - d\|^2$$

where ω is the support found in previous iterations. Even though methods exist to make the above minimization problem less costly, it is still computational expensive compared to MP and OMP.

Based on these three methods, MP, OMP, and OLS, many new and more advanced methods have been introduced. Without going into too many details, there are methods to select and deselect components, such as the Single Best Replacement (SBR) [Sou+11]. This algorithm can minimize the penalized $\ell_2 - \ell_0$ problem, and at each iteration, it updates the support by either adding or subtracting a component. Further studies of minimizing the penalized problem have been done, which includes finding the optimal lambda [Sou+15]. Methods, adding a non-negative constraint, have been studied, see [Ngu+19], and references therein.

At last, it must be noted that countless other methods exist to solve sparse problems. For more in-depth reviews, see [Zha+15; Mar+18; Sou16].

1.5 CONTRIBUTION

This thesis focuses on the constrained $\ell_2 - \ell_0$ minimization

$$\min_x G_k(x) = \frac{1}{2}\|Ax - d\|^2 + \chi_{\|\cdot\|_0 \leqslant k}(x).$$

We have presented in the previous section, the state of the art in different approaches to sparse optimization. However, it seems that the relaxation literature is more focused on the penalized problem:

$$G_{\ell_0}(x) = \frac{1}{2}\|Ax - d\|^2 + \lambda\|x\|_0.$$

Inspired by recent developments in sparse optimization, we propose in Chapter 2, to study a continuous relaxation of the constrained problem:

$$\min_x G_Q(x) = \frac{1}{2}\|Ax - d\|^2 + Q(x),$$

where $Q : \mathbb{R}^N \to \mathbb{R}_{\geqslant 0}$, and is defined in (2.11). To our knowledge, this is the first time an explicit form of a relaxation of the constrained formulation is proposed. G_Q has many favorable properties, and most importantly, a minimizer of G_Q that satisfies the sparsity constraint is a minimizer of G_k.

We present the proximal operator of the new regularizer. Thus, standard minimization schemes such as FBS can be applied.

Numerically, we are not sure to always converge towards a solution that satisfies the sparsity constraint. Thus, we implement a "Fail-Safe" strategy into the algorithm. The algorithm always converges to a solution of the initial problem.

In chapter 3, we propose an exact reformulation of the constrained and penalized formulation, by introducing an auxiliary variable. We obtain a biconvex expression. We name the two methods Constrained Biconvex method (CoBic) and Penalized Biconvex method (PeBic). We show that they can be defined as

$$G_\rho(x, u) = \frac{1}{2}\|Ax - d\|^2 + I(u) + \chi_{\cdot \geqslant 0}(x) + \rho(\|x\|_1 - < x, u >).$$

where $I(u)$ is a convex function defined in (3.6) for CoBic and (3.7) for PeBic. Compared to the methods introduced in Section 1.4.2, we have tighter results concerning the formulation's exactness. To compute the solution we use an homotopy continuation algorithm. The algorithm is easy to implement as it is built on already well-known minimization problems.

Finally, in Chapter 4, we study the multiplicative criterion; more specifically, we search to minimize

$$J(x) = \frac{1}{2}\|Ax - d\|^2 R(x),$$

where $R(x)$ is a regularization term that promotes sparsity. Based on the optimality conditions of the problem, we show that the numerical minimization of $J(x)$ can be realized by minimizing the following additive functional

$$\frac{1}{2}\|Ax - d\|^2 + \|\Lambda_{\sqrt{\beta}} x\|^2,$$

where $\Lambda_{\sqrt{\beta}}$ is a diagonal matrix with $\sqrt{\beta}$ on the diagonal. β_i is a parameter that is updated at each iteration, and thus the method can be viewed as an algorithm with an adaptive penalization parameter. The approach is interesting as, in theory, it could be parameter-free. This is a work in progress.

We apply the methods to Single Molecule Localization microscopy (SMLM). This is a deconvolution and upsampling problem. We first create simulated data to compare and show the advantage of minimizing G_Q, CoBic, and PeBic compared to minimizing the constrained or penalized formulation directly. We further apply the methods to data from the 2013 ISBI SMLM challenge, and compare the methods. Not only do the methods perform as well as other state-of-the-art methods in $\ell_2 - \ell_0$ minimization, but the sparsity constraint in G_Q and CoBic may be easier to adjust than the penalizing parameter λ.

2

A CONTINUOUS RELAXATION OF THE CONSTRAINED $\ell_2 - \ell_0$ PROBLEM

Contents

In this chapter, we present a continuous relaxation of the constrained $\ell_2 - \ell_0$ problem. The inspiration for our work is the CEL0 relaxation, presented in Section 1.4.1.2. This work aims to investigate if an equivalent relaxation of the constrained problem can be constructed. Will the relaxation conserve the global minima, such as the CEL0 method does? We answer the question in this chapter.

Following the CEL0 method, we compute the convex envelope of the constrained $\ell_2 - \ell_0$ problem when the observation matrix A is orthogonal. We obtain a relaxation function $Q(x)$, and we use this as a continuous relaxation of the ℓ_0 constraint. We investigate the properties of the new functional, named G_Q, by calculating the generalized subgradient and give numerical examples in two dimensions. We calculate the proximal operator of Q, and show that classical minimization schemes, such as Forward-Backward splitting methods, can be applied. The chapter is based on the article [BBFA20a].

First, in this chapter, we assume that the columns of A are normalized.

Proposition 2.1. *We can suppose that* $\|a_i\|_2 = 1$, $\forall\, i$ *without loss of generality.*

Proof. The proof is based on the fact that ℓ_0-norm is invariant to a multiplication factor. Let $\Lambda_{\|a_i\|}$ and $\Lambda_{\frac{1}{\|a_i\|}}$ be diagonal matrices with the norm of a_i (respectively $1/\|a_i\|$) on its diagonal, and let $z = \Lambda_{\|a_i\|} x$, then $\|\Lambda_{\frac{1}{\|a_i\|}} z\|_0 = \|z\|_0 = \|x\|_0$, and thus

$$\arg\min_x \frac{1}{2}\|Ax - d\|_2^2 + \chi_{\|\cdot\|_0 \leqslant k}(x) = \Lambda_{\frac{1}{\|a_i\|}} \arg\min_z \frac{1}{2}\|A_n z - d\|_2^2 + \chi_{\|\cdot\|_0 \leqslant k}(z)$$

where A_n is a matrix deduced from A where the norm of the columns are 1. □

2.1 THE CONVEX ENVELOPE OF THE CONSTRAINED PROBLEM WHEN A IS ORTHOGONAL

In this section, we are interested in the case where A is an orthogonal matrix, i.e. $< a_j, a_i >= 0, \forall i \neq j$. In contrast to the penalized $\ell_2 - \ell_0$ problem (1.12), G_k (1.10) with A orthogonal is not separable. Thus the computation of the convex envelope in the N dimensional case cannot be reduced to the sum of N *one* dimensional cases (as in the case of CEL0 described in section 1.4.1.2). The constrained $\ell_2 - \ell_0$ problem (1.10) can be written as

$$G_k(x) = \frac{1}{2}\|Ax - d\|^2 + \chi_{\|\cdot\|_0 \leqslant k}(x) \tag{2.1}$$

where χ is the indicator function defined in Notations. Before calculating the convex envelope, some preliminary results are needed.

Proposition 2.2. *Let $x \in \mathbb{R}^N$. There exists $T_k(x) \in \mathbb{N}$ such that $0 < T_k(x) \leqslant k$ and*

$$|x^{\downarrow}_{k-T_k(x)+1}| \leqslant \frac{1}{T_k(x)} \sum_{i=k-T_k(x)+1}^{N} |x^{\downarrow}_i| \leqslant |x^{\downarrow}_{k-T_k(x)}| \tag{2.2}$$

where the left inequality is strict if $T_k(x) \neq 1$, and where $x_0 = +\infty$. Furthermore, $T_k(x)$ is defined as the smallest integer that verifies the double inequality.

The proof of existence is given in the Appendix A.1. We will also use the Legendre-Fenchel transformation which is essential in the calculation of the convex envelope.

Definition 2.1. *The Legendre-Fenchel transformation of a function $f : \mathbb{R}^N \to \mathbb{R} \cup \{+\infty\}$ is defined as*

$$f^*(u^*) = \sup_{u \in \mathbb{R}^N} < u, u^* > -f(u).$$

The biconjugate of a function, that is applying the Legendre-Fenchel transformation twice, is the convex envelope of the function.

Following [SBFA15], we present the convex envelope of G_k (2.1) when A is orthogonal.

Theorem 2.3. *Let $A \in \mathbb{R}^{M \times N}$ be such that $A^T A = I$. The convex envelope of $G_k(x)$ is*

$$G_k^{**}(x) = \frac{1}{2}\|Ax - d\|_2^2 + Q(x) \tag{2.3}$$

where

$$Q(x) = -\frac{1}{2}\sum_{i=k-T_k(x)+1}^{N} x_i^{\downarrow 2} + \frac{1}{2T_k(x)}\left(\sum_{i=k-T_k(x)+1}^{N} |x_i^{\downarrow}|\right)^2 \quad (2.4)$$

and where $T_k(x)$ *is defined as in Proposition 2.2.*

Proof. Since $A^TA = I$, the function G_k (2.1) can be rewritten as

$$G_k(x) = \chi_{\|\cdot\|_0 \leqslant k}(x) + \frac{1}{2}\|d-b\|_2^2 + \frac{1}{2}\|x-z\|_2^2 \quad (2.5)$$

where $b = AA^Td$ and $z = A^Td$. This reformulation allows us to decompose the data-fitting term into a sum of 1-dimensional functions. We apply the Legendre transformation on the functional (2.5):

$$G_k^*(y) = \sup_{x\in\mathbb{R}^N} < x, y > -\chi_{\|\cdot\|_0 \leqslant k}(x) - \frac{1}{2}\|d-b\|_2^2 - \frac{1}{2}\|x-z\|_2^2.$$

We leave out the terms that are not depending on x.

$$G_k^*(y) = -\frac{1}{2}\|d-b\|_2^2 + \left(\sup_{x\in\mathbb{R}^N} < x, y > -\chi_{\|\cdot\|_0 \leqslant k}(x) - \frac{1}{2}\|x-z\|_2^2\right).$$

Writing differently the expression inside the supremum we get

$$G_k^*(y) = -\frac{1}{2}\|d-b\|_2^2 + \left(\sup_{x\in\mathbb{R}^N} -\chi_{\|\cdot\|_0 \leqslant k}(x) - \frac{1}{2}\|x-(z+y)\|_2^2 + \frac{1}{2}\|z+y\|_2^2 - \frac{1}{2}\|z\|_2^2\right).$$

We develop further

$$G_k^*(y) = -\frac{1}{2}\|d-b\|_2^2 - \frac{1}{2}\|z\|_2^2 + \frac{1}{2}\|z+y\|_2^2 + \left(\sup_{x\in\mathbb{R}^N} -\chi_{\|\cdot\|_0 \leqslant k}(x) - \frac{1}{2}\|x-(z+y)\|_2^2\right).$$

The supremum is reached when $x_i = (z+y)_i^{\downarrow}$, $i \leqslant k$, and $x_i = 0$, $\forall i > k$. The Legendre transformation of G_k is therefore

$$G_k^*(y) = -\frac{1}{2}\|d-b\|_2^2 - \frac{1}{2}\|z\|_2^2 + \frac{1}{2}\sum_{i=1}^{k}(z+y)_i^{\downarrow 2}.$$

To obtain the convex envelope of the function G_k, we compute the Legendre transformation of G_k^*.

$$G_k^{**}(x) = \sup_y < x, y > +\frac{1}{2}\|d-b\|_2^2 + \frac{1}{2}\|z\|_2^2 - \frac{1}{2}\sum_{i=1}^{k}(z+y)_i^{\downarrow 2}.$$

We add and subtract $\frac{1}{2}\|x\|^2$ and $< x, z >$ in order to obtain an expression that is easier to work with.

$$\begin{aligned} G_k^{**}(x) &= \sup_y < x, y > + \frac{1}{2}\|d - b\|_2^2 + \frac{1}{2}\|z\|_2^2 + \frac{1}{2}\|x\|^2 - \frac{1}{2}\|x\|^2 \\ &\qquad + < x, z > - < x, z > - \frac{1}{2}\sum_{i=1}^{k}(z+y)_i^{\downarrow 2} \\ &= \sup_y < x, z+y > + \frac{1}{2}\|d - b\|_2^2 \\ &\qquad + \frac{1}{2}\|x - z\|_2^2 - \frac{1}{2}\|x\|^2 - \frac{1}{2}\sum_{i=1}^{k}(z+y)_i^{\downarrow 2}. \end{aligned}$$

Noticing that $\frac{1}{2}\|d - b\|_2^2 + \frac{1}{2}\|x - z\|_2^2 = \frac{1}{2}\|Ax - d\|_2^2$, using the notation $w = z + y$, and given the definition of $w^{\downarrow}$, this is equivalent to

$$G_k^{**}(x) = \frac{1}{2}\|Ax - d\|_2^2 - \frac{1}{2}\|x\|^2 + \sup_{w \in \mathbb{R}^N} < x, w > - \frac{1}{2}\sum_{i=1}^{k} w_i^{\downarrow 2}. \quad (2.6)$$

The above supremum problem can be solved by using Lemma 2.4, which is presented after this proof. This yields

$$\begin{aligned} G_k^{**}(x) = \frac{1}{2}\|Ax - d\|_2^2 &- \frac{1}{2}\sum_{i=k-T_k(x)+1}^{N} x_i^{\downarrow 2} \\ &+ \frac{1}{2T_k(x)}\left(\sum_{i=k-T_k(x)+1}^{N} |x_i^{\downarrow}|\right)^2. \quad (2.7) \end{aligned}$$

□

The following Lemma is necessary in the proof of the convex envelope.

Lemma 2.4. *Let $x \in \mathbb{R}^N$. Consider the following supremum problem*

$$\sup_{y \in \mathbb{R}^N} -\frac{1}{2}\sum_{i=1}^{k} y_i^{\downarrow 2} + < y, x > . \quad (2.8)$$

This problem is concave and the value of the supremum problem (2.8) is

$$\frac{1}{2}\sum_{i=1}^{k-T_k(x)} x_i^{\downarrow 2} + \frac{1}{2T_k(x)}\left(\sum_{i=k-T_k(x)+1}^{N} |x_i^{\downarrow}|\right)^2.$$

$T_k(x)$ is defined in Proposition 2.2. The supremum argument is given by

$$y = P^{(x)^{-1}}\hat{y}$$

where $\hat{y}$ *is*

$$\hat{y}_j(x) = \begin{cases} \operatorname{sign}(x_j^{\downarrow}) \frac{1}{T_k(x)} \sum_{i=k-T_k(x)+1}^{N} |x_i^{\downarrow}| & \text{if } k \geqslant j \geqslant k - T_k(x) + 1 \\ & \text{or if } j > k \text{ and } x_j^{\downarrow} \neq 0 \\ [-1, 1] \frac{1}{T_k(x)} \sum_{i=k-T_k(x)+1}^{N} |x_i^{\downarrow}| & \text{if } j > k \text{ and } x_j^{\downarrow} = 0 \\ x_j^{\downarrow} & \text{if } j < k - T_k(x) + 1. \end{cases} \tag{2.9}$$

The proof can be found in Appendix A.2.

Remark 2.1. $\hat{y}$ *is such that* $\hat{y} = \hat{y}^{\downarrow}$.

The expression of the convex envelope (2.3) may be hard to grasp since the expression is on a non-closed form. To understand better $Q(x)$ we have the following properties.

Property 2.1. $Q(x) : \mathbb{R}^N \to [0, \infty[$.

Proof. Let us show that $Q(x) \geqslant 0$, $\forall x$. We use equation (2.4) as starting point.

$$\begin{aligned} Q(x) &= -\frac{1}{2} \sum_{i=k-T_k(x)+1}^{N} x_i^{\downarrow 2} + \frac{1}{2T_k(x)} \left(\sum_{i=k-T_k(x)+1}^{N} |x_i^{\downarrow}| \right)^2 \\ &\geqslant -\frac{1}{2} |x_{k-T_k(x)+1}^{\downarrow}| \sum_{i=k-T_k(x)+1}^{N} |x_i^{\downarrow}| + \frac{1}{2T_k(x)} \left(\sum_{i=k-T_k(x)+1}^{N} |x_i^{\downarrow}| \right)^2 \\ &\geqslant -\frac{1}{2} |x_{k-T_k(x)+1}^{\downarrow}| \sum_{i=k-T_k(x)+1}^{N} |x_i^{\downarrow}| + \frac{1}{2} |x_{k-T_k(x)+1}^{\downarrow}| \sum_{i=k-T_k(x)+1}^{N} |x_i^{\downarrow}| \\ &= 0. \end{aligned}$$

We used the fact that $|x_{k-T_k(x)+1}^{\downarrow}| \geqslant |x_i^{\downarrow}|$, $\forall i \geqslant k - T_k(x) + 1$ for the first inequality. For the second inequality, we used the inequality in the definition of $T_k(x)$ (see Proposition 2.2) to go from the second to third line. Note that for $T_k(x) > 1$ the last inequality is strict. □

Property 2.2. *The function* $Q(x)$ *is continuous on* $\mathbb{R}^N$.

Proof. By definition we have that $G_k^{**}(x) = \frac{1}{2}\|Ax - d\|^2 + Q(x)$ when A is orthogonal, and G_k^{**} is lower semi-continuous, and continuous in the interior of its domain. From [RW09, Corollary 3.47] for coercive functions, $\operatorname{dom}(\operatorname{co}(f)) = \operatorname{co}(\operatorname{dom}(f))$, where co is the convex envelope of a function and dom is the domain of the function. First, G_k is coercive when A is orthogonal since we have $\|Ax\|^2 = (Ax)^T Ax = x^T A^T Ax = \|x\|^2$. G_k^{**} is continuous on $\mathbb{R}^N$. Since $\operatorname{dom}(G_k)$ is made up of all different supports where $\|x\|_0 \leqslant k$, so its convex envelope is

$\mathbb{R}^N$. Thus $\mathrm{dom}(G_k^{**}) = \mathbb{R}^N$, and G_k^{**} is continuous on $\mathbb{R}^N$. Moreover, $Q(x) = G_k^{**}(x) - \frac{1}{2}\|Ax - d\|^2$, so $Q(x)$ is the difference between a continuous function and a continuous function, and is independent of A, and thus continuous. □

Property 2.3. *Let $\|x\|_0 \leqslant k$. Then $T_k(x)$ as defined in Proposition 2.2 is such that $T_k(x) = 1$. The inverse it not necessarily true.*

Proof. From Proposition 2.2 we know that $T_k(x)$ satisfies

$$|x^{\downarrow}_{k-T_k(x)+1}| \leqslant \frac{1}{T_k(x)} \sum_{i=k-T_k(x)+1}^{N} |x^{\downarrow}_i| \leqslant |x^{\downarrow}_{k-T_k(x)}|.$$

First, note that for all x such that $\|x\|_0 \leqslant k$, we have $\forall j > k$, $x^{\downarrow}_j = 0$, and in this case the inequalities are clearly satisfied for $T_k(x) = 1$. Furthermore, $T_k(x)$ is defined as the smallest possible integer, and thus $T_k(x) = 1$.

An example to prove the inverse is not true: Let $x = (6, 3, 2, 1)^T$. Let $k = 2$, then

$$\sum_{i=k}^{N} |x^{\downarrow}_i| = 6 \leqslant |x^{\downarrow}_{k-1}| = 6.$$

$T_k(x) = 1$ but the constraint $\|x\|_0 \leqslant 2$ is clearly not satisfied. □

Property 2.4. $Q(x) = 0$ ***if and only if*** $\|x\|_0 \leqslant k$.

Proof. From Property 2.1, $Q(x) \geqslant 0$ and the inequality is strict if $T_k(x) > 1$. Thus, it suffices to investigate $T_k(x) = 1$. In that case, the expression of $Q(x)$ (2.4) can be written as:

$$Q(x) = \sum_{j=k+1}^{N} \sum_{i=k}^{j-1} |x^{\downarrow}_i||x^{\downarrow}_j|$$

which is equal to 0 only if at least $\forall j$, $j > k$, $x^{\downarrow}_j = 0$. □

In the next section, we will investigate the use of $Q(x)$ when A is not orthogonal.

2.2 A NEW RELAXATION

From now on, we suppose $A \in \mathbb{R}^{M \times N}$ with A not necessarily orthogonal.

We are interested in a continuous relaxation of G_k defined as

$$G_k = \frac{1}{2}\|Ax - d\|^2 + \chi_{\|\cdot\|_0 \leqslant k}(x).$$

Following the CEL0 approach, we propose the following relaxation of G_k:

$$G_Q(x) = \frac{1}{2}\|Ax - d\|^2 + Q(x) \tag{2.10}$$

with

$$Q(x) = -\frac{1}{2} \sum_{i=k-T_k(x)+1}^{N} x_i^{\downarrow 2} + \frac{1}{2T_k(x)} \left(\sum_{i=k-T_k(x)+1}^{N} |x_i^{\downarrow}| \right)^2 \quad (2.11)$$

where $T_k(x)$ is the function defined in Proposition 2.2:

$$|x^{\downarrow}_{k-T_k(x)+1}| \leqslant \frac{1}{T_k(x)} \sum_{i=k-T_k(x)+1}^{N} |x_i^{\downarrow}| \leqslant |x^{\downarrow}_{k-T_k(x)}| \quad (2.12)$$

where, by definition, the left inequality in (2.12) is strict if $T_k(x) > 1$.

Remark that, from its definition (see Eq. (2.6)), $Q(x)$ can be written as

$$Q(x) = -\frac{1}{2} \sum_{i=1}^{N} x_i^2 + \sup_{w \in \mathbb{R}^N} -\frac{1}{2} \sum_{i=1}^{k} w_i^{\downarrow 2} + < w, x > . \quad (2.13)$$

Note that the properties of $Q(x)$ proved in Section 2.1 are valid for any A.

The exactness of a relaxation means that the relaxation has the same global minimizers as the initial function. Furthermore, it does not add any minimizers that are not minimizers of the initial function. The CEL0 relaxation [SBFA15] is an exact relaxation of the penalized functional (1.12). The proposed relaxation G_Q of the constraint functional G_k (2.1) is not exact as a counterexample later in this chapter shows. We can prove, however, some partial results.

Remark 2.2. *From Property 2.4, we have* $Q(x) = 0 \; \forall \, x$ *such that* $\|x\|_0 \leqslant k$. *Thus* $G_Q(x) = G_k(x) \; \forall \, x$ *such that* $\|x\|_0 \leqslant k$.

Theorem 2.5. *Let* $\hat{x}$ *be a local (respectively global) minimizer of* G_Q. *If* $\|\hat{x}\|_0 \leqslant k$, *then* $\hat{x}$ *is a local (respectively global) minimizer of* G_k.

Proof. Let $\mathscr{S} := \{x : \|x\|_0 \leqslant k\}$. Let $\hat{x}$ be a local minimizer of G_Q, such that $\|\hat{x}\|_0 \leqslant k$ and let $\mathscr{N}(\hat{x}, \gamma)$ denote the γ-neighborhood of $\hat{x}$. By contradiction assume that $\exists \bar{x} \in \mathscr{N}(\hat{x}, \gamma) \bigcup \mathscr{S}$ s.t. $G_k(\bar{x}) < G_k(\hat{x})$. From Remark 2.2, $G_Q(\bar{x}) = G_k(\bar{x})$ and $G_Q(\hat{x}) = G_k(\hat{x})$, which means $\exists \bar{x} \in \mathscr{N}(\hat{x}, \gamma) \cup \mathscr{S}$ s.t. $G_Q(\bar{x}) < G_Q(\hat{x})$ which is a contradiction since $\hat{x}$ is a minimizer of G_Q. The same reasoning can be applied in the case of global minimizers. □

Thus, if a minimizer of the relaxed functional satisfies the sparsity constraint, then it is a minimizer of the initial problem. Furthermore, the relaxation is a mix of absolute values and squares and promotes therefore sparsity.

Further note that we could have applied the quadratic envelope [Car19] to obtain the relaxation Q. The quadratic envelope can be

defined as applying twice the S_υ transformation on a function f. The S_υ transformation is defined as:

$$S_\upsilon(f)(y) := \sup_x -f(x) - \frac{\upsilon}{2}\|x-y\|^2.$$

If we apply the quadratic envelope to the constrained ℓ_0 indicator function, we obtain υQ. Further, the author proposes to either choose $\upsilon I \prec A^T A$, where I is the identity matrix, or $\upsilon I \succ A^T A$. It is important to note that if we have a υ such that $\upsilon I \not\succ A^T A$, does not mean that $\upsilon I \prec A^T A$. When υ is such that $\upsilon I \succ A^T A$ the relaxation is *exact*. However, numerically, we found this condition far too strong, and it did not perform better than minimizing the initial hard constraint function G_k (2.1). For a normalized matrix A, Q can be found by taking $\upsilon = 1$ in $S_\upsilon(S_\upsilon(\chi_{\|\cdot\|_0 \leqslant k}))$. However, we do not have necessarily $I \succ A^T A$. Nevertheless we show in this chapter, some exact relaxation properties for G_Q.

Furthermore, what is hidden in our proposed method is the fact that each column of A is normalized. Without this assumption, *each* element x_i would be weighted by $\|a_i\|^2$, which is finer than multiplying the same constant to the whole the regularization term. Again, we can compare with the CEL0 relaxation. When applying the quadratic envelope to the ℓ_0 penalization term, we obtain CEL0, but instead of $\|a_i\|^2$ in the expression, there is a υ.

However, we are obliged to normalize A to calculate the proximal operator of the regularization term.

2.2.1 *The subgradient*

In this section, we calculate the subgradient of G_Q. Since G_Q is neither smooth nor convex, we cannot calculate the gradient nor the subgradient in the sense of convex analysis. We calculate the generalized subgradient (or Clarke subgradient). The obtained expression shows the difficulties to give optimal necessary conditions for the relaxation.

To calculate the generalized subgradient, we must first prove that $Q(x)$ is locally Lipschitz.

Definition 2.2. *A function $f : \mathbb{R}^N \to \mathbb{R}$ is locally Lipschitz at point x if*

$$\exists (L, \epsilon), \forall (y, y') \in \mathcal{N}(x, \epsilon)^2, |f(y) - f(y')| \leqslant L\|y - y'\|$$

where $L \in \mathbb{R}_{\geqslant 0}$, and $\mathcal{N}(x, \epsilon)$ is a ϵ neighborhood of x.

Lemma 2.6. *$Q(x)$ is locally Lipschitz, $\forall x \in \mathbb{R}^N$.*

Proof. First, it is well-known that the supremum of locally Lipschitz functions is locally Lipschitz. Let us use the definition of $Q(x)$ from (2.13). The function defined as $x \to \sup_w -\frac{1}{2}\sum_{i=1}^k w_i^{\downarrow 2} + < w, x >$ is

locally Lipschitz since $\forall i$ the functions $x \to -\frac{1}{2}\sum_{i=1}^{k} w_i^{\downarrow 2} + < w, x >$ are locally Lipschitz. Furthermore, the sum of two locally Lipschitz functions is locally Lipschitz. □

Since $Q(x)$ is locally Lipschitz, we can search for the generalized subgradient, denoted ∂.

Definition 2.3. *The generalized subgradient [Cla90] of a function* $f : \mathbb{R}^N \to \mathbb{R}$ *(which is locally Lipschitz) is defined by*

$$\partial f(x) := \{\xi \in \mathbb{R}^N : f^0(x, v) \geqslant < v, \xi >, \forall v \in \mathbb{R}^N\}$$

where $f^0(x, v)$ *is the generalized directional derivative in the direction* v*,*

$$f^0(x, v) = \limsup_{\substack{y \to x \\ \eta \downarrow 0}} \frac{f(y + \eta v) - f(y)}{\eta}.$$

Theorem 2.7. *Let* $x \in \mathbb{R}^N$*, and let* $T_k(x)$ *be as defined in Proposition 2.2. The subgradient of* $G_Q(x)$ *is*

$$\partial G_Q(x) = A^*(Ax - d) - x + y(x) \tag{2.14}$$

where $y(x)$ *is the argument where the supremum is reached in Lemma 2.4.*

Proof. G_Q is sum of three functions, $\sup_w -\frac{1}{2}\sum_{i=1}^{k} w_i^{\downarrow 2} + < w, x >$, $\frac{1}{2}\|Ax - d\|^2$ and $-\frac{1}{2}\|x\|^2$. From [Cla90, Proposition 2.3.3 and Corollary 1] and since the two last functions are differentiable, we can write the generalized subgradient of G_Q as the sum of the gradient of the two last functions and the generalized subgradient of the first, i.e.

$$\begin{aligned}\partial G_Q = \nabla[\frac{1}{2}\|A \cdot -d\|^2](x) - \nabla[\frac{1}{2}\|\cdot\|^2](x) \\ + \partial[\sup_{w \in \mathbb{R}^N} -\frac{1}{2}\sum_{i=1}^{k} w_i^{\downarrow 2} + < w, \cdot >](x).\end{aligned} \tag{2.15}$$

Thus, the difficulty is to calculate $\partial[\sup_w -\frac{1}{2}\sum_{i=1}^{k} w_i^{\downarrow 2} + < w, \cdot >](x)$.

From [MN13, Theorem 2.93], the subgradient of the supremum is the convex envelop of the subgradients where the supremum is reached. We define $g(w, x) = -\frac{1}{2}\sum_{i=1}^{k} w_i^{\downarrow 2} + < w, x >$. The subgradient of g with respect to x is $\partial(g(w, \cdot))(x) = w$. Now, we need to find the supremum in $\sup_w -\frac{1}{2}\sum_{i=1}^{k} w_i^{\downarrow 2} + < w, x >$. From Lemma 2.4, we know that the supremum is reached at $y(x)$, given in (2.9). We insert $y(x)$ into (2.15) and this concludes the proof. □

2.2.2 *Numerical examples*

In order to obtain a clearer view of what is gained with the proposed relaxation, we study two numerical examples in two dimensions. We set $k = 1$ and the initial problem is

$$G_k(x) := \frac{1}{2}\|Ax - d\|^2 + \chi_{\|\cdot\|_0 \leqslant 1}(x).$$

In two dimensions, finding the minimum of $G_{k=1}$ is a simple problem to solve. The solution is either when the first component, $\hat{x}_1$ is 0, or when the second component $\hat{x}_2 = 0$, or both. For $k = 1$, we have necessarily that $T(x) = 1$, and the relaxed formulation is then

$$G_Q(x) = \frac{1}{2}\|Ax - d\|^2 + |x_1||x_2|.$$

We consider first two cases where $A \in \mathbb{R}^{2\times 2}$ is an orthogonal matrix.

$$A = \begin{pmatrix} 1 & 2 \\ -2 & 1 \end{pmatrix} \Lambda_{1/\|a_i\|} \quad \text{and} \quad d = \begin{pmatrix} 2 \\ 1 \end{pmatrix} \tag{2.16}$$

$$A = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \text{and} \quad d = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \tag{2.17}$$

where $\Lambda_{1/\|a_i\|}$ is a diagonal matrix with $\frac{1}{\|a_i\|}$ on its diagonal, and $\|a_i\|$ is the norm of the ith column of A. Figure 2.1 presents the contour lines of G_k and G_Q. The red semi-transparency layer over the contour line of the G_k represents the infinite value, and the blue semi-transparency layer over the relaxation highlights the axes. The green line in the lower level right figure represents the global minima. The Figure 2.1 presents the initial and the convex envelope of the initial problem. We observe that the relaxations are convex, and example 2.16 has one global minimum. Example 2.17 shows when the global minima are on a line. The two extremums of the line are on the axis, and are the global minima of the intial function.

In the two following cases $A \in \mathbb{R}^{2\times 2}$ is not orthogonal.

$$A = \begin{pmatrix} 3 & 2 \\ 1 & 3 \end{pmatrix} \Lambda_{1/\|a_i\|} \quad \text{and } d = \begin{pmatrix} 1 \\ 2 \end{pmatrix} \tag{2.18}$$

$$A = \begin{pmatrix} -3 & -2 \\ 1 & 3 \end{pmatrix} \Lambda_{1/\|a_i\|} \quad \text{and } d = \begin{pmatrix} 1 \\ 2 \end{pmatrix} \tag{2.19}$$

The figures (2.2) show the advantages of using G_Q as relaxation. The relaxation is continuous, and in Example (2.18), the relaxation is exact. This can be observed in the upper row in Fig. 2.2. Example (2.19) gives an example when the relaxation is not exact. In the lower row of Fig. 2.2 we observe the effect of the relaxation, as it is a product of the absolute value of x_1 and x_2. The global minima for the relaxation in this case is situated in $(-0.086, 1.0912)$ and the two minima for G_k are $(-0.3162, 0)$ and $(0, 1.094)$.

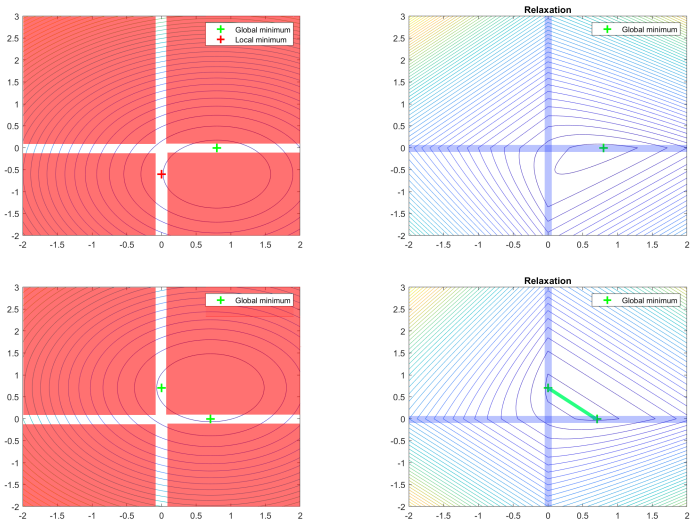


Figure 2.1: Top: Level lines of the function G_k and G_Q for the example (2.16). Bottom: Level lines of the function G_k and G_Q for the example (2.17).

2.3 ALGORITHMS TO DEAL WITH THE RELAXATION

In this section, we present an algorithm to use for the minimization of G_Q. The analysis of the relaxation shows that it promotes sparsity. The function G_Q is non-convex and non-smooth, but in comparison to the initial function G_k, G_Q is continuous. During the thesis, we had two main difficulties concerning the numerical aspects of the minimization: First, the calculation of the proximal operator of Q. Secondly, a non-negativity constraint is used in many image processing applications, so the algorithm must minimize the functional plus a non-negativity constraint.

Both problems were solved. To calculate the proximal operator of Q, we had to normalize the columns of A, and use Proposition 2.9. Furthermore, the proximal operator of the sum of Q and the non-negativity constraint remains unknown. To avoid this problem, we use a penalty term of negative values, defined as

$$\mathrm{dist}^2_{\mathbb{R}^n_{\geqslant 0}}(x) := \inf_{y \geqslant 0} \frac{1}{2}\|x-y\|^2$$

Finally, we use an acceleration of the FBS algorithm (see Algorithm (1)), the Nonmonotone Accelerated Proximal gradient algorithm (nmAPG) algorithm [LL15], which is used in the numerical experiences of this thesis (see Appendix A.4 for the steps of the algorithm)[1]. The algo-

1 Note that we searched for algorithms that can minimize the sum of G_Q and the non-negativity constraint. Alternative methods such as SDMM [CP11; FBD10; SST10] and inexact proximal methods [Gu+18; Bon+17] were studied, but they are not used in numerical applications, due to the computational time. One could also implement a subgradient method by using gradient bundle methods (see [Bur+18] for

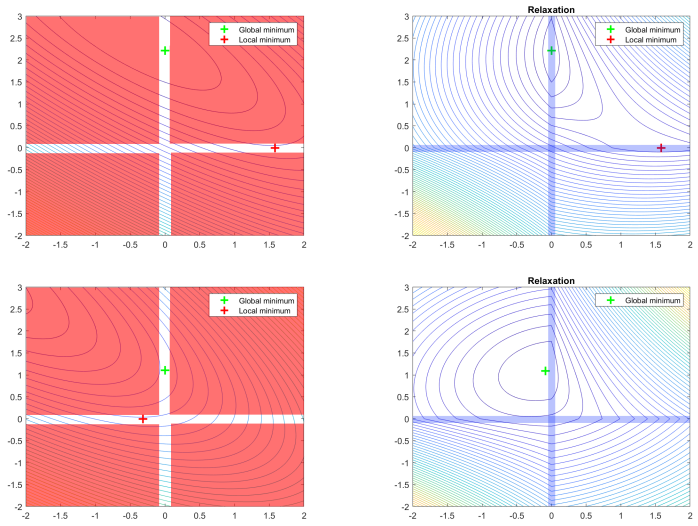


Figure 2.2: Top: Level lines of the function G_k and G_Q for the example (2.18). Bottom: Level lines of the function G_k and G_Q for the example (2.19).

rithm is proved to converge when the cost function satisfies the K-L property.

We split the G_Q into f and g.

$$f(x) := \frac{1}{2}\|Ax - d\|^2 + \alpha \mathrm{dist}^2_{\mathbb{R}_{\geqslant 0}}(x)$$
$$g(x) := Q(x),$$

where $\alpha \in \mathbb{R}_{\geqslant 0}$. To use nmAPG, we need the gradient of $f(x)$, and the proximal operator of $Q(x)$, which is presented in this section.

Lemma 2.8. *G_Q satisfies the K-Ł property.*

Proof. $\frac{1}{2}\|Ax - d\|^2$ is semi-algebraic. Using the definition of $Q(x)$ in (2.13) we can prove that $Q(x)$ is semi-algebraic. $\|x\|_2^2$ is semi-algebraic. Since

$$\sum_{i=1}^{k} x_i^{\downarrow 2} = \sup_y g(x,y) := -\chi_{\|\cdot\|_0 \leqslant k}(y) - \frac{1}{2}\|x - y\|^2$$

and $g(x,y)$ is semi-algebraic [BST14], then $\sum_{i=1}^{k} x_i^{\downarrow 2}$ is semi-algebraic. Thus, $f(x,y) := -\sum_{i=1}^{k} y_i^{\downarrow 2} + < x, y >$ is semi-algebraic, and the supremum as well. We can conclude that $Q(x)$ is semi-algebraic, and thus G_Q satisfies the K-L property. □

Remark 2.3. *Since G_Q does not always have a minimum that corresponds to the k-sparsity, we can add a "fail-safe" strategy to ensure that the algorithm always converges to a solution that satisfies the sparsity constraint.*

an overview) or classical subgradient methods. However, there are no convergence guarantees for the latter.

A simple projection of the minimum obtained from G_Q *to the constraint* $\|x\|_0 \leqslant k$ *using the proximal of the constraint (see (1.28)) and then the calculation of the optimal intensity for the given support would suffice. In mathematical terms, let* ω *be the support found after a projection on to the* k*-sparse space. Then, we can minimize* $\frac{1}{2}\|A_\omega x_\omega - d\|^2$*. Since* $A_\omega \in \mathbb{R}^{M\times \#\omega}$*, and* $\#\omega < M$*, the problem is overdetermined and easily solved.*

The expression of $Q(x)$ in (2.4) is not on a closed-form expression because of the function $T_k(x)$ and calculating the proximal operator directly from this expression is difficult. The following proposition facilitates the calculation of prox_Q. The proposition is inspired by [Car16, Proposition 3.3], and the proof is available in Appendix A.3.

Proposition 2.9. *Let* $\gamma > 1$ *and* $z = prox_{-(\frac{\gamma-1}{\gamma})\sum_{i=k+1}^{N}(\cdot)^{\downarrow 2}}(y)$*. We have*

$$prox_{\frac{Q}{\gamma}}(y) = \frac{\gamma y - z}{\gamma - 1}. \tag{2.20}$$

Thus, it suffices to calculate the proximal operator of

$$\zeta(x) := -(\frac{\gamma-1}{\gamma})\sum_{i=k+1}^{N} x_i^{\downarrow 2}.$$

This is done in Lemma A.7 in the Appendix A.3. The following theorem presents the proximal operator of Q

Theorem 2.10. *The proximal operator of* Q *for* $\gamma > 1$ *is such that*

$$prox_{\frac{Q}{\gamma}}(y)_i^{\downarrow y} = \begin{cases} \frac{\gamma y_i^{\downarrow} - \text{sign}(y_i^{\downarrow})\max(|y_i^{\downarrow}|,\tau)}{\gamma-1} & \text{if } i \leqslant k \\ \frac{\gamma y_i^{\downarrow} - \text{sign}(y_i^{\downarrow})\min(\tau,\gamma|y_i^{\downarrow}|)}{\gamma-1} & \text{if } i > k \end{cases}$$

or, equivalently

$$prox_{\frac{Q}{\gamma}}(y)_i^{\downarrow y} = \begin{cases} y_i^{\downarrow} & \text{if } i \leqslant k^* \\ \frac{\gamma y_i^{\downarrow} - \text{sign}(y_i^{\downarrow})\tau}{\gamma-1} & \text{if } k^* < i < k^{**} \\ 0 & \text{if } k^{**} \leqslant i. \end{cases}$$

where k^* *is the first index such that* $\tau > |y_i^{\downarrow}|$ *and* k^{**} *is the first index such that* $\gamma|y_i^{\downarrow}| < \tau$*.* τ *is a value in the interval* $[|y_k^{\downarrow}|, \gamma|y_{k+1}^{\downarrow}|[$*, and is defined as*

$$\tau = \frac{\gamma\sum_{i\in n_1}|y_i^{\downarrow}| + \gamma\sum_{i\in n_2}|y_i^{\downarrow}|}{\gamma\#n_1 + \#n_2} \tag{2.21}$$

where n_1 *and* n_2 *are two groups of indices such that* $\forall i \in n_1, y_i^{\downarrow} < \tau$ *and* $\forall i \in n_2, \tau \leqslant \gamma|y_i^{\downarrow}|$ *for an* $\#n_1$ *and* $\#n_2$ *are the sizes of* $n1$ *and* $n2$*. To go from* $prox_{\frac{Q}{\gamma}}(y)^{\downarrow y}$ *to* $prox_{\frac{Q}{\gamma}}(y)$ *we apply the inverse permutation that sort* y *to* $y^{\downarrow}$*.*

Proof. The result is direct by applying Proposition 2.9 and Lemma A.7 which present the proximal operator of $\text{prox}_{\zeta}(y)$, the latter is presented in Appendix A.3. □

Note that the proximal operator of Q is only a relaxation of the proximal operator of $\|x\|_0 \leqslant k$, which keeps the k largest values of x. Further note that the search for τ can be done iteratively by sorting in descending order all values of $y_i^{\downarrow}$ $i \leqslant k$ and $\gamma y_i^{\downarrow}$ $i > k$ that are (with respect to their absolute value) in the interval $[|y_k^{\downarrow}|, \gamma|y_{k+1}^{\downarrow}|]$. The elements in the interval are sorted, and denoted p_i. n_1,n_2 must calculated for each interval $[p_{i+1}, p_i]$. The search is over if τ is $\in [p_{i+1}, p_i]$.

The codes to compute the proximal operator and the cost function are available online:
https://github.com/abechens/SMLM-Constraint-Relaxation

2.3.1 *Numerical examples of the proximal operator*

The proximal operator of Q may not be easy to grasp. The main difficulty is to find tau. It is on a non-closed form, so this section gives a step-by-step explanation of the proximal operator.

We start with a vector $y \in \mathbb{R}^{11}$ already sorted by its magnitude, i.e., $y = y^{\downarrow}$.

$$y = (8,\ 7.5,\ 7,\ 6.5,\ 6,\ 5.5,\ 5,\ 4.5,\ 4,\ 3.5,\ 3)^T$$

We let γ be 1.5, and $k = 6$. From Theorem 2.10, we know that τ is situated in $[|y_k^{\downarrow}|, \gamma|y_{k+1}^{\downarrow}|]$, and we define $w \in \mathbb{R}^{11}$:

$$w_i = \begin{cases} y_i \text{ if } i \leqslant k \\ \gamma y_i \text{ else} \end{cases} \quad \Leftrightarrow \tag{2.22}$$

$$w = (8,\ 7.5,\ 7,\ 6.5,\ 6,\ 5.5,\ 7.5,\ 6.75,\ 6,\ 5.25,\ 4.5)^T.$$

We refer to w as the *unconstrained minimizers*, as w would be the proximal operator of $(\frac{\gamma-1}{\gamma})\sum_{i=k+1}^{N}(x_i)^{\downarrow 2}$ if we did not have that $x^{\downarrow}$ is the vector x sorted.

The w is plotted in Fig. 2.3, where the horizontal axis is the coordinate index of the vectors. The right plot highlights in gray the area in which we search for τ, as stipulated in Theorem 2.10, i.e., $\tau \in [5.5, 7.5]$. τ is calculated for each element interval. For example, to determine if τ is in the interval $[6.75, 7]$, we identify the elements of y that are less than 7 for $i \leqslant k$, and the elements of γy larger than 6.75 for $i > k$. This yields n_1 and n_2, respectively. We apply the formula (2.21) from Theorem 2.10.

$$\tau = \gamma\frac{6.5 + 6 + 5.5 + 5}{3\gamma + 1} \approx 6.27$$

Since $\tau = 6.27 \notin [6.75,\ 7]$, it is not the optimal value we search.

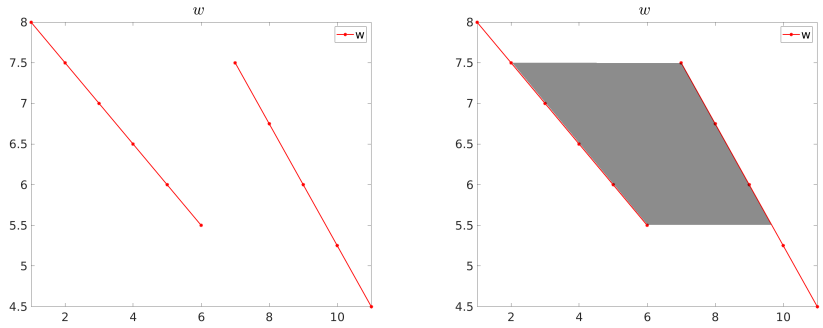


Figure 2.3: Left: The plot of the unconstrained minimizers, w. Right: The interval in which we search τ is highlighted in grey.

The left plot of Fig. 2.4 presents values of τ for each possible interval. Note that τ must be in the predefined interval. In the plot, only the value 6.30 corresponds to that criteria, since the interval is $[6, 6.5]$. In the right figure is the τ plotted. The arrows emphasize the projection of each element. The calculation of the τ is simply a minimization of a specific cost-of-projection.

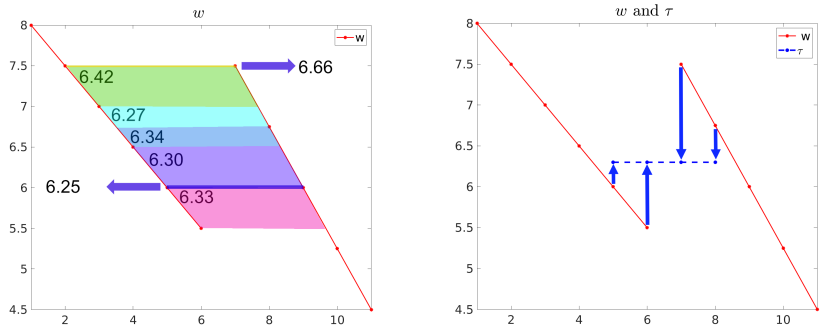


Figure 2.4: Left: The value of τ for each interval. Right: τ plotted and projection.

Finally, we have the $\text{prox}_{\zeta}(y)$, and, by applying Proposition 2.9 we have the proximal operator of Q. Both visible in Fig. 2.5. The right figure compares the hard threshold with the proximal of the relaxation Q. The proximal operator is, in fact, a relaxation, as it does not enforce k-sparsity, but still penalizes the smallest components. This may play a role in not getting stuck in a local minimum too fast when using the FBS algorithm.

Note that the hard threshold operator is not unique. If the kth and the $k+1$th largest element of y are equal, then the hard threshold chooses either the kth or the $k+1$th element. In this example we have $\gamma = 1.2$, $k = 6$ and

$$y = (8, 7.5, 7, 6.5, 6, 6, 6, 6, 6, 5.5, 5, 4.5, 4, 3.5)^T.$$

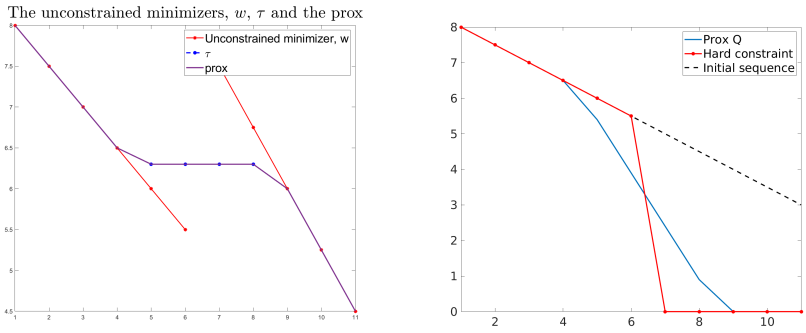


Figure 2.5: Left: The $\text{prox}_{\zeta}(y)$. Right: The proximal operator of Q (blue), initial vector y (black) and the hard threshold (red).

Note that the hard threshold of y could be:

$$\text{prox}_{\frac{\chi_{\|\cdot\|_0 \leqslant k}}{\gamma}}(y) = \begin{cases} (8, 7.5, 7, 6.5, 6, 6, 0, 0, 0, 0, 0, 0, 0, 0)^T, \text{ or} \\ (8, 7.5, 7, 6.5, 0, 6, 6, 0, 0, 0, 0, 0, 0, 0)^T, \text{ or} \\ (8, 7.5, 7, 6.5, 0, 0, 6, 6, 0, 0, 0, 0, 0, 0)^T, \text{ or} \\ (8, 7.5, 7, 6.5, 6, 0, 6, 0, 0, 0, 0, 0, 0, 0)^T, \text{ or} \\ (8, 7.5, 7, 6.5, 6, 0, 0, 6, 0, 0, 0, 0, 0, 0)^T. \end{cases}$$

The proximal operator of Q does not have this behaviour, and treats identical entries identically. This is shown in the following figures. Figure 2.6 plots the unconstrained minimizers, w and the value τ. Figure 2.7 shows the $\text{prox}_{\zeta}(y)$ and the proximal operator of Q. Both visible in Fig. 2.5. The right figure plots the hard threshold and the proximal operator of Q. We observe that the hard threshold keeps only the k-largest, and the computer chooses afterwards by lexicographical order. The proximal operator of Q does treat however these element *equally*.

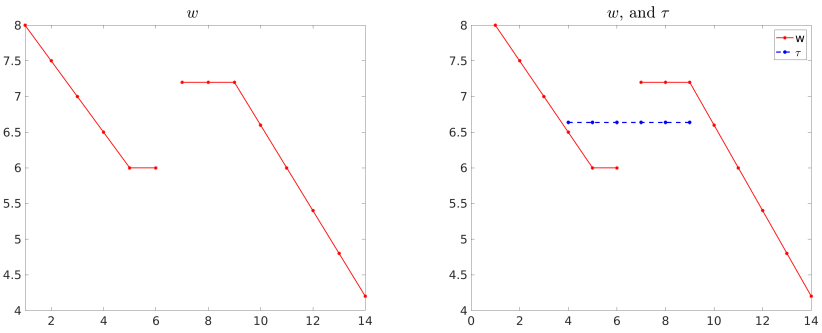


Figure 2.6: Left: The unconstrained minimizers, w. Right: The unconstrained minimizers, w, and τ.

As seen, the Hard thresholding acts differently than the relaxation. The effect of this can be shown in a small numerical example. We use

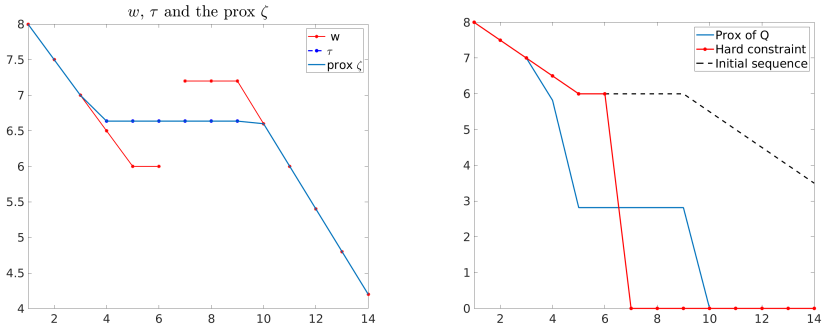


Figure 2.7: Left: The $\text{prox}_{\zeta}(y)$. Right: The proximal operator of Q (blue), initial vector y (black) and the hard threshold (red).

the same example as found in section 2.2.2, with A and d as in (2.18). We use the algorithm nmAPG to minimize G_Q and initial function G_k. In Fig. 2.8, each step of the minimization is plotted over the level

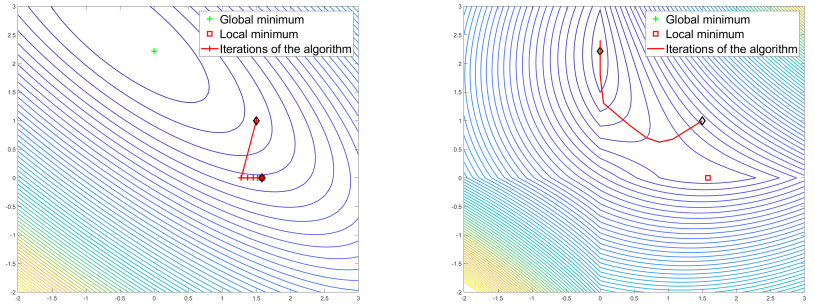


Figure 2.8: Left: The minimization of the initial function G_k. Right: The minimization of G_Q.

lines. The black diamond spots are the initial and finish points. Both algorithms converge to a minimum, but only the relaxation converges to the global minimum. It can be observed that the C-IHT takes a step towards the global minima, but the hard threshold projects the step towards the closest axis. In this case, this leads to the local minima.

2.4 CONCLUSION

We have investigated in this chapter a continuous relaxation of the constrained $\ell_2 - \ell_0$ problem. We compute the convex hull of G_k when A is orthogonal. We further propose to use the same relaxation for any A and name this relaxation G_Q. This is the same procedure as the authors used to obtain CEL0 [SBFA15]. The question that has driven us has been answered, the proposed relaxation, G_Q, is not exact for every observation matrix A. However, it promotes sparsity and is continuous. We propose an algorithm to minimize the relaxed function.

We further add a "fail-safe" which ensures convergence to a critical point of the initial functional.

3

EXACT BICONVEX MINIMIZATION TO SPARSE OPTIMIZATION

Contents

This chapter introduces CoBic and PeBic, two new methods to minimize the constrained and penalized problem. An ArXiv article inspired us, and we resume the article in the introduction. After introducing the inspiration, we start with the reformulation of the ℓ_0-norm, which is introduced into the constrained and penalized problem. The problems are then relaxed. We name this CoBic and PeBic. In Section 3.2.1, we prove the exactness of the reformulations and propose an algorithm to minimize the functionals.

3.1 INSPIRATION

The results in [YG16] seemed promising. The authors study problems of the form

$$\min f(x) \text{ s.t. } \|Bx\|_0 \leqslant k, \tag{3.1}$$

where $f(x)$ is a convex and *L-Lipschitz*, and $B \in \mathbb{R}^{M\times N}$, and $\mathrm{rank}(B) = M$. Note that B can be the identity matrix. The authors proposes a reformulation of the ℓ_0-norm, see Eq. (3.4). They introduce the reformulation into problem (3.1), and further relax the problem such that the final problem is

$$\min_{x,u} f(x) + \chi_{\|\cdot\|_1 \leqslant k}(u) + \chi_{-1\leqslant\cdot\leqslant 1}(u) + \rho(\|Bx\|_1 - < Bx, u >). \tag{3.2}$$

They claim to prove the exactness of the reconstruction for $\rho > \frac{L}{\sigma_m(B)}$, where L is the Lipschitz constant of the function f, and $\sigma_m(B)$ is the

smallest singular value of B. However, in the case of $\ell_2 - \ell_0$ minimization the proposed method cannot be used as the ℓ_2-data fidelity term is *not* L-Lipschitz continuous, but gradient L-Lipschitz continuous. Other reformulation methods exist such as Liu and Bi [BLP14], and they study a reformulation of ℓ_0-norm on the following form:

$$\|x\|_0 = \min_{0 \leqslant v \leqslant e} < e, e - v > \text{ s.t. } < v, |x| >= 0. \tag{3.3}$$

However, they do not apply their reformulation to the constrained $\ell_2 - \ell_0$ problem. They study the reformulation with the data fidelity term $\frac{1}{2}\|Ax - d\|$. Note that it is not squared.

We choose to study the reformulation proposed by [YG16], since they apply their reformulations to the constrained $\ell_2 - \ell_0$ problem (1.10) with good results. However, the constrained $\ell_2 - \ell_0$ formulation does not verify their hypothesis. We concluded that further study of the reformulation was possible and were inspired to adapt this work to encompass the ℓ_2-data fidelity term.

3.2 EXACT BICONVEX FORMULATION OF THE $\ell_2 - \ell_0$ PROBLEMS

In this section, we focus on a reformulation of the ℓ_0-norm. [YG16] first introduced this formulation where they rewrite the ℓ_0-norm as convex minimization problem by adding an auxiliary variable. We can write the ℓ_0-norm of any $x \in \mathbb{R}^N$ as

$$\|x\|_0 = \min_{\substack{-1 \leqslant u \leqslant 1 \\ u \in \mathbb{R}^N}} \|u\|_1 \text{ s.t } \|x\|_1 =< u, x > . \tag{3.4}$$

Even though the introduction of the auxiliary variable u increases the dimension of the problem, the non-convex and non-continuous ℓ_0-norm can now be written as a *convex and continuous* minimization problem. In this chapter, we study the $\ell_2 - \ell_0$ constrained and penalized problems using the reformulation of the ℓ_0-norm. We also add a non-negativity constraint to the x variable as it is usually used as a priori in imaging problems. We can get an unified notation for the constrained and penalized forms

$$\min_{x,u} \frac{1}{2}\|Ax - d\|^2 + I(u) + \chi_{\cdot \geqslant 0}(x) \text{ s.t. } \|x\|_1 =< x, u >, \tag{3.5}$$

where $I(u)$ is, in the case of the constrained problem (1.10):

$$I(u) = \begin{cases} 0 \text{ if } \|u\|_1 \leqslant k \text{ and } \forall\, i, -1 \leqslant u_i \leqslant 1 \\ \infty \text{ otherwise.} \end{cases} \tag{3.6}$$

For the penalized problem (1.12):

$$I(u) = \begin{cases} \lambda \|u\|_1 \text{ if } \forall\, i,\ -1 \leqslant u_i \leqslant 1 \\ \infty \text{ otherwise.} \end{cases} \tag{3.7}$$

We note $\mathcal{S} = \{(x,u); \|x\|_1 =< x,u >\}$, and we define the functional G as

$$G(x,u) = \frac{1}{2}\|Ax - d\|^2 + I(u) + \chi_{\cdot \geqslant 0}(x) + \chi_{\mathcal{S}}(x,u). \tag{3.8}$$

The functional (3.8) is continuous and biconvex with respect to (x,u): the functional $G(x,u)$ in (3.8) is convex with respect to x while u is fixed, and conversely. However, globally, it is still non-convex due to of the space $\mathcal{S}$. We can relax this constraint by introducing a penalty term, $\rho(\|x\|_1 - < x,u >)$, which is based on the method of Lagrange Multipliers.

We define the Lagrangian cost function $G_\rho(x,u) : \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}$ as

$$G_\rho(x,u) = \frac{1}{2}\|Ax - d\|^2 + I(u) + \chi_{\cdot \geqslant 0}(x) + \rho(\|x\|_1 - < x,u >). \tag{3.9}$$

In this chapter, we are interested in exact reformulation methods. This means that any minimizer of (3.9) must be also a minimizer of (3.8) and conversely. We show that for $\rho > \sigma(A)\|d\|$, with $\sigma(A)$ the largest singular value of A, $G_\rho(x,u)$ is exact.

3.2.1 *Theoretical results*

The theoretical results of this section have been published in [BBFA20b].

In this section we present the theoretical foundation of our work. Theorem 3.1 and 3.4 show that minimizing (3.9) is equivalent in terms of minimizers as minimizing (3.8), given ρ is large enough.

This theorem differs from [LBP18, Corollary 3.2] as their ρ may be arbitrarily large in contrast to this work where ρ can be calculated precisely. Furthermore, they work with a slightly different reformulation of the ℓ_0-norm and not explicitly with the problem (5.3) since they assume their loss-function to be locally Lipschitzian.

Similarly, the DC method proposed in [GTT18] (presented in Section 1.4.2), proves that the algorithm converges to a critical point of the initial function as long as $\rho > \|A^T d\|_2 + \frac{3}{2}LC$, with L being the L-Lipschitz gradient, and C is a value that bounds the optimal argument $\|\hat{x}\|_2$ of (1.23) $\forall \rho$. Thus, even though they do not introduces an auxiliary variable, we recognize the idea of introducing a reformulation of the k-sparsity constraint and further relax the reformulation by introducing it as a penalty term. A drawback in their work is that ρ must be chosen arbitrarily large as C is unknown.

Theorem 3.1 (Constrained form). *Assume that* $\rho > \sigma(A)\|d\|_2$, *and* A *is full rank. Let* G_ρ *and* G *be defined respectively in (3.9) and (3.8) with* $I(u)$ *defined by (3.6). We have:*

1. *If* (x_ρ, u_ρ) *is a local (respectively global) minimizer of* G_ρ, *then* (x_ρ, u_ρ) *is a local (respectively global) minimizer of* G.
2. *If* $(\hat{x}, \hat{u})$ *is a global minimizer of* G, *then* $(\hat{x}, \hat{u})$ *is a global minimizer of* G_ρ.

Two lemmas are needed in order to proof Theorem 3.1. The complete proofs of these lemmas require three other lemmas (Lemma B.2, Lemma B.5, and Lemma B.6) stated in the Appendix B.

Lemma 3.2. *Let* $\rho > \sigma(A)\|d\|_2$. *Let* (x_ρ, u_ρ) *be a local or global minimizer of* $G_\rho(x,u) := \frac{1}{2}\|Ax - d\|^2 + I(u) + \rho(\|x\|_1 - < x, u >)$ *with* $I(u)$ *defined as in (3.6) or (3.7). Let* $\omega = \{i \in \{1,\dots,N\}; (u_\rho)_i = 0\}$. *Then* $(x_\rho)_i = 0 \, \forall i \in \omega$.

Proof. Let J denote the set of indices: $J = \{1,\dots,N\}\backslash\omega$. If (x_ρ, u_ρ) is a local or global minimizer of G_ρ then $\forall (x,u) \in \mathcal{N}((x_\rho, u_\rho), \gamma)$, where $\mathcal{N}((x_\rho, u_\rho), \gamma)$ denotes a neighborhood of (x_ρ, u_ρ) of size γ, we have

$$\frac{1}{2}\|Ax_\rho - d\|^2 + \chi_{\cdot \geqslant 0}(x_\rho) + I(u_\rho) + \rho(\|x_\rho\|_1 - < x_\rho, u_\rho >) \leqslant \frac{1}{2}\|Ax - d\|^2 + \chi_{\cdot \geqslant 0}(x) + I(u) + \rho(\|x\|_1 - < x, u >).$$

By choosing $u = u_\rho$ and $x = \tilde{x}$ with $\tilde{x}_J = (x_\rho)_J$ and $\tilde{x}_\omega = x_\omega$, with $(x_\omega, (u_\rho)_\omega) \in \mathcal{N}(((x_\rho)_\omega, (u_\rho)_\omega), \gamma)$, we have

$$\frac{1}{2}\|Ax_\rho - d\|^2 + \chi_{\cdot \geqslant 0}(x_\rho) + \rho\|(x_\rho)_\omega\|_1 \leqslant \frac{1}{2}\|A\tilde{x} - d\|^2 + \chi_{\cdot \geqslant 0}(\tilde{x}) + \rho\|x_\omega\|_1. \tag{3.10}$$

We want to show that $(x_\rho)_\omega$ is zero. We have

$$\begin{aligned}
\|Ax - d\|^2 &= \|Ax\|^2 + \|d\|^2 - 2 < Ax, d > \\
&= \sum_i (Ax)_i^2 + \|d\|^2 - 2\sum_i x_i (A^T d)_i \\
&= \sum_i \left[(\sum_{j \in J} A_{ij} x_j)^2 + (\sum_{j \in \omega} A_{ij} x_j)^2 \right] + \|d\|^2 - \\
&\quad 2\left[\sum_{i \in J} x_i (A^T d)_i + \sum_{i \in \omega} x_i (A^T d)_i \right]
\end{aligned}$$

Using the above decomposition simplifies (3.10), and we have $\forall\ x_\omega$:

$$\frac{1}{2}\sum_i \left(\sum_{j\in\omega} A_{ij}(x_\rho)_j\right)^2 - \sum_{i\in\omega}(x_\rho)_i(A^T d)_i + \rho\|(x_\rho)_\omega\|_1 + \chi_{\cdot\geqslant 0}(x_\rho)$$
$$\leqslant \frac{1}{2}\sum_i \left(\sum_{j\in\omega} A_{ij}x_j\right)^2 - \sum_{i\in\omega} x_i(A^T d)_i + \rho\|x_\omega\|_1 + \chi_{\cdot\geqslant 0}(x_\omega).$$

Thus $(x_\rho)_\omega$ is a solution of

$$\arg\min_{x_\omega} \frac{1}{2}\sum_i \left(\sum_{j\in\omega} A_{ij}x_j\right)^2 - \sum_{i\in\omega} x_i(A^T d)_i + \rho\|x_\omega\|_1 + \chi_{\cdot\geqslant 0}(x_\omega),$$

or, equivalently solution of

$$\arg\min_{x_\omega} \frac{1}{2}\|A_\omega x_\omega - d\|^2 + \rho\|x_\omega\|_1 + \chi_{\cdot\geqslant 0}(x_\omega) \tag{3.11}$$

where A_ω is the $P \times \#\omega$ submatrix of A composed by the columns indexed by ω of A. With Lemma B.2 , we have that $\sigma(A) \geqslant \sigma(A_\omega)$ and if $\rho > \sigma(A)\|d\|_2$ we can apply Lemma B.5 with w a vector composed of ρ. We conclude that $(x_\rho)_\omega = 0$. □

Lemma 3.3. *If $\rho > \sigma(A)\|d\|_2$, let (x_ρ, u_ρ) be a local or global minimizer of*

$$\arg\min_{x,u} \frac{1}{2}\|Ax - d\|^2 + \chi_{\cdot\geqslant 0}(x) + \rho(\|x\|_1 - < x, u >) + I(u),$$

with $I(u)$ defined as in (3.6), that is, the constrained form. Then

$$\|x_\rho\|_1 - < x_\rho, u_\rho >= 0.$$

Proof. From Lemma B.6 (see Appendix B), we have that $(u_\rho)_i(x_\rho)_i = |(x_\rho)_i| \forall\ i \in J$, and $(u_\rho)_i = 0\ \forall i \in \omega$. It suffices to prove $(x_\rho)_i = 0\ \forall i \in \omega$. For that we use Lemma 3.2 and conclude that $(x_\rho)_\omega = 0$. □

With the two above lemmas, we can prove Theorem 3.1.

Proof. We start by proving the first part of the theorem. Let (x_ρ, u_ρ) be a local minimizer of G_ρ, with $I(u)$ on the constrained form, that is, defined as in (3.6). Let $S = \{(x,u); \|x\|_1 =< x, u >\}$. If $\rho > \sigma(A)\|d\|_2$ then, from Lemma 3.3,

$$(x_\rho, u_\rho) \text{ verifies } \|x_\rho\|_1 =< x_\rho, u_\rho > .$$

Furthermore, from the definition of a minimizer, we have

$$G_\rho(x_\rho, u_\rho) \leqslant G_\rho(x,u)\ \forall (x,u) \in \mathcal{N}((x_\rho, u_\rho), \gamma),$$

and so we have

$$G_\rho(x_\rho, u_\rho) \leqslant G_\rho(x, u)\ \forall (x, u) \in \mathcal{N}((x_\rho, u_\rho), \gamma) \cap \mathcal{S}.$$

Since $\forall (x, u) \in \mathcal{S}, G_\rho(x, u) = G(x_\rho, u_\rho)$, we have

$$G(x_\rho, u_\rho) \leqslant G(x, u)\ \forall (x, u) \in \mathcal{N}((x_\rho, u_\rho), \gamma) \cap \mathcal{S} \tag{3.12}$$

By the definition, (x_ρ, u_ρ) is also a local minimizer of G.

Now we prove part 2 of Theorem 3.1.

Let $(\hat{x}, \hat{u})$ be a global minimizer of G. We necessarily have $\|\hat{x}\|_1 =< \hat{x}, \hat{u} >$. First, we show that

$$G_\rho(\hat{x}, \hat{u}) \leqslant \min G_\rho(x, u).$$

This can be shown by contradiction. Assume the opposite, and denote (x_ρ, u_ρ) a global minimizer of G_ρ. We then have

$$G_\rho(\hat{x}, \hat{u}) > \min G_\rho(x, u) = G_\rho(x_\rho, u_\rho). \tag{3.13}$$

Lemma 3.3 shows that $\|x_\rho\|_1 =< x_\rho, u_\rho >$, so $G_\rho(x_\rho, u_\rho) = G(x_\rho, u_\rho)$ and we have

$$G(\hat{x}, \hat{u}) = G_\rho(\hat{x}, \hat{u}) > \min G_\rho(x, u) = G_\rho(x_\rho, u_\rho) = G(x_\rho, u_\rho),$$

and more precisely, $G(\hat{x}, \hat{u}) > G(x_\rho, u_\rho)$ which is not possible, since $(\hat{x}, \hat{u})$ is a global minimizer of G.

We therefore have shown that $G_\rho(\hat{x}, \hat{u}) \leqslant \min G_\rho(x, u)$, and we have

$$G_\rho(\hat{x}, \hat{u}) \leqslant \min G_\rho(x, u) \leqslant G_\rho(x, u) \quad \forall (x, u).$$

$(\hat{x}, \hat{u})$ is thus a global minimizer of G_ρ.

□

Theorem 3.4 (Penalized form). *Assume that* $\rho > \sigma(A)\|d\|_2$, *and* A *is full rank. Let* G_ρ *and* G *be defined respectively in (3.9) and (3.8) with* $I(u)$ *defined in (3.7). We have:*

1. *If* (x_ρ, u_ρ) *is a local (respectively global) minimizer of* G_ρ, *then we can construct* $(x_\rho, \tilde{u}_\rho)$ *which is a local (respectively global) minimizer of* G.

2. *If* $(\hat{x}, \hat{u})$ *is a global minimizer of* G, *then* $(\hat{x}, \hat{u})$ *is a global minimizer of* G_ρ.

For the proof, we need two lemmas, Lemma 3.2 which is already presented and the following lemma.

Lemma 3.5. *Let* (x_ρ, u_ρ) *be a local or a global minimizer of* G_ρ *for the penalized form* ($I(u)$ *defined by (3.7)). If* $\rho > \sigma(A)\|d\|_2$ *then* $\forall i$ *such that* $(u_\rho)_i = 0$ *we have* $(x_\rho)_i = 0$

Proof. From Lemma B.7 (see Appendix B), we have that $(u_\rho)_i = 0$ iff $(x_\rho)_i \in (-\frac{\lambda}{\rho}, \frac{\lambda}{\rho})$. We denote ω the set of indices where $u_\rho = 0$, and we can apply Lemma 3.2, and conclude that $(x_\rho)_\omega = 0$.

□

Remark 3.1. *If $\rho > \sigma(A)\|d\|_2$, note that the cost function G_ρ with minimizers (x_ρ, u_ρ) is constant on $|(x_\rho)_i| = \frac{\lambda}{\rho}$ and $|(u_\rho)_i| \in [0, 1]$.*

Remark 3.2. *In the case of the penalized form, the minimizers (x_ρ, u_ρ) of G_ρ with $\rho > \sigma(A)\|d\|_2$ may be such that $< x_\rho, u_\rho > \neq \|x_\rho\|_1$. This may only happen if $|(x_\rho)_i| = \frac{\lambda}{\rho}$.*

Remark 3.3. *If $\rho > \sigma(A)\|d\|_2$. From Remark 3.1, from a minimizer (x_ρ, u_ρ) of G_ρ, we can construct a minimiser $(x_\rho, \tilde{u}_\rho)$ of G_ρ such that $\|x_\rho\|_1 = < x_\rho, \tilde{u}_\rho >$. This can be done by denoting Z, the set of indices such that $0 < |(u_\rho)_i| < 1$. If Z is non-empty, we have $< x_\rho, u_\rho > \neq \|x_\rho\|_1$. From Remark 3.2, $|(x_\rho)_i| = \frac{\lambda}{\rho} \forall i \in Z$. Take $\tilde{u}_{\rho i} = \text{sign}(x_i)\ \forall i \in Z$ and $\tilde{u}_{\rho i} = (u_\rho)_i \forall i \notin Z$, then $< x_\rho, \tilde{u}_\rho > = \|x_\rho\|_1$. Furthermore, $(x_\rho, \tilde{u}_\rho)$ is a minimizer of G_ρ according to Remark 3.1 and the fact that $G_\rho(x_\rho, u)$ is convex with respect to u.*

With Lemma 3.5 and the above remarks, we can prove Theorem 3.4.

Proof. We start by proving the first part of the theorem. Given (x_ρ, u_ρ) a local or global minimizer of G_ρ, with $I(u)$ on the penalized form, that is, defined as in (3.6). Let $\mathcal{S}$ denote the space where $\|x\|_1 = < x, u >$. If $\rho > \sigma(A)\|d\|_2$ then, from Remark 3.3, we can construct $(x_\rho, \tilde{u}_\rho)$ such that

$$(x_\rho, \tilde{u}_\rho) \text{ verifies } \|x_\rho\|_1 = < x_\rho, \tilde{u}_\rho > .$$

Furthermore, from the definition of a minimizer, we have

$$G_\rho(x_\rho, \tilde{u}_\rho) \leqslant G_\rho(x, u)\ \forall (x, u) \in \mathcal{N}((x_\rho, \tilde{u}_\rho), \gamma)$$

and so we get

$$G_\rho(x_\rho, \tilde{u}_\rho) \leqslant G_\rho(x, u)\ \forall (x, u) \in \mathcal{N}((x_\rho, \tilde{u}_\rho), \gamma) \cap \mathcal{S}$$

Since $\forall (x, u) \in \mathcal{S}, G_\rho(x, u) = G(x_\rho, u_\rho)$, we obtain

$$G(x_\rho, \tilde{u}_\rho) \leqslant G(x, u)\ \forall (x, u) \in \mathcal{N}((x_\rho, \tilde{u}_\rho), \gamma) \cap \mathcal{S} \tag{3.14}$$

Then, $(x_\rho, \tilde{u}_\rho)$ is also a local minimizer of G.

The second part of Theorem 3.4 can be proved as in the proof of Theorem 3.1. □

3.3 MINIMIZATION OF THE PROPOSED METHOD

The minimization algorithm for G_ρ is presented in this section. We refer to the *constrained* biconvex algorithm by CoBic and to the *penalized* biconvex algorithm by PeBic.

The main body of the algorithm depends on two particularities of G_ρ; G_ρ is convex when $\rho = 0$, and the non-convexity of G_ρ is due to the coupling term $< x, u >$. These two properties inspire the idea of an homotopy continuation algorithm for minimizing $G_\rho(x, u)$. The minimization is initialized with a small $\rho^{(0)}$ and $(x^{(0)}, u^{(0)})$ as zero vector. We minimizes $G_{\rho^{(0)}}(x^{(0)}, u^{(0)})$ and the result is denoted $(x^{(1)}, u^{(1)})$. The penalty parameter ρ increases at each iteration. For a given iteration, p, we minimize $G_{\rho^{(p)}}(x^{(p)}, u^{(p)})$, with $(x^{(p)}, u^{(p)})$ the solution of $\arg\min G_{\rho^{(p-1)}}(x^{(p-1)}, u^{(p-1)})$. This method will hopefully give a proper initialization for the final minimization, which is when $\rho > \sigma(A)\|d\|$. The second attractive property of functional G_ρ is the bi-convexity. Alternating minimization is therefore suitable. With this in mind, and following [YG16], we propose the following algorithm.

3.3.1 *Algorithm*

Algorithm 3 : Biconvex minimization

Input :
$\rho^{(0)}$ small ;

Initialization :
$x^{(0)} = 0 \in \mathbb{R}^N$; $u^{(0)} = 0 \in \mathbb{R}^N$; $p = 0$;

Repeat :
Solve problem $G_{\rho^{(p)}}$

$$\{x^{(p+1)}, u^{(p+1)}\} \in \arg\min G_{\rho^{(p)}}(x^{(p)}, u^{(p)}) \tag{3.15}$$

Update the penalty parameter:

$$\rho^{(p+1)} = \min(\sigma(A)\|d\|_2, 2\rho^{(p)}) \tag{3.16}$$

Until : $\rho^{(p+1)} = \sigma(A)\|d\|_2$

Output : $x^{(p+1)}$

The Proximal Alternating Minimization (PAM) algorithm [Att+10] minimizes (3.15). The algorithm ensures convergence to a critical point, and thus Algorithm 3 converges to a critical point. [1]

1 Note that minimization by block (also known as Gauss-Seidel minimization) is not suited as for each step, the minimum should be unique. Furthermore, Proximal alternating *linearized* minimization [BST14] is not suited, as the functional $G_\rho(x, u)$

The PAM minimizes functions of the form

$$L(x,u) = f(x) + g(u) + Y(x,u) \tag{3.17}$$

In our case, we have, $f(x) = \frac{1}{2}\|Ax-d\|^2 + \rho\|x\|_1 + \chi_{\cdot\geqslant 0}(x)$, $g(u) = I(u)$ and $Y(x,u) = -\rho < x,u >$. PAM has the following outline

$$\begin{cases} \text{Repeat} \\ x^{(s+1)} \in \arg\min_x \left\{ G_\rho(x,u^{(s)}) + \frac{1}{2c^{(s)}}\|x-x^{(s)}\|_2^2 \right\} \\ u^{(s+1)} \in \arg\min_u \left\{ G_\rho(x^{(s+1)},u) + \frac{1}{2b^{(s)}}\|u-u^{(s)}\|_2^2 \right\} \\ \text{Until convergence} \end{cases} \tag{3.18}$$

$c^{(s)}$ and $b^{(s)}$ add strict convexity to each block, and $c^{(s)}, b^{(s)}$ are bounded from below and above. In this work, we fix $c^{(s)} = b^{(s)} = 10^4$. In the following section we develop minimization schemes for (3.18) in the case of the constrained ($I(u)$ defined by (3.6)) and respectively the penalized ($I(u)$ defined by (3.7)) problem. Recall that $\arg\min G_\rho$ is defined as

$$\arg\min_{x,u} \frac{1}{2}\|Ax-d\|^2 + I(u) + \rho(\|x\|_1 - < x,u >) + \chi_{\cdot\geqslant 0}(x) \tag{3.19}$$

where $I(u)$ is defined by (3.6) or (3.7).

3.3.2 *Minimization with respect to* x

The minimization with respect to x using PAM is

$$x^{(s+1)} \in \arg\min_{x\in\mathbb{R}^N} \frac{1}{2}\|Ax-d\|^2 + \rho(\|x\|_1 - < x,u^{(s)} >) + \frac{1}{2c^{(s)}}\|x-x^{(s)}\|_2^2 + \chi_{\cdot\geqslant 0}(x)$$

which can be rewritten as

$$x^{(s+1)} \in \arg\min_{x\in\mathbb{R}^N} \frac{1}{2}\|Ax-d\|^2 + \frac{1}{2c^{(s)}}\|x-(x^{(s)} + \rho c^{(s)}u^{(s)})\|^2 + \rho\|x\|_1 + \chi_{\cdot\geqslant 0}(x)$$

We apply the FISTA algorithm [BT09] to solve the above problem. This algorithm is designed to work with functionals on the form of $F(x) = f(x) + g(x)$ where f is a smooth convex function with a Lipschitz continuous gradient $L(f)$. g is a continuous convex function and possibly non-smooth. In our case we have

does not meet all the hypotheses. Further note that ADMM [Boy+11] is not suitable as the algorithm supposes a linear relation between the variables x and u.

$$f(x) = \frac{1}{2}\|Ax - d\|^2 + \frac{1}{2c^{(s)}}\|x - (x^{(s)} + \rho c^{(s)} u^{(s)})\|^2 \tag{3.20}$$

$$g(x) = \rho\|x\|_1 + \chi_{\cdot \geqslant 0}(x) \tag{3.21}$$

which verify the conditions to apply FISTA. The proximal operator of $g(x)$ is the soft thresholding with positivity constraint

$$\text{prox}_{\frac{g}{L(f)}}(x) = \begin{cases} x_i - \frac{\rho}{L(f)} \text{ if } x_i > \frac{\rho}{L(f)} \\ 0 \text{ if } x_i \leqslant \frac{\rho}{L(f)} \end{cases}$$

3.3.3 *Minimization with respect to* u

We study how to find a solution to the convex minimization problem

$$u^{(s+1)} = \arg\min_{u \in \mathbb{R}^N} \frac{1}{2b^{(s)}}\|u - u^{(s)}\|_2^2 - \rho < x^{(s+1)}, u > + I(u)$$

The above problem can be written as

$$u^{(s+1)} = \arg\min_{u \in \mathbb{R}^N} \frac{1}{2b^{(s)}}\|u - (u^{(s)} + \rho b^{(s)} x^{(s+1)})\|^2 + I(u) \tag{3.22}$$

and to simplify, we denote $z = u^{(s)} + \rho b^{(s)} x^{(s+1)}$. In the next two paragraphs, we study the above problem for $I(u)$ defined by (3.6) or (3.7).

Firstly, we work with the constrained biconvex formulation of G_ρ, CoBic. $I(u)$ is thus defined by (3.6). The minimization problem (3.22) can be written as

$$u^{(s+1)} = \arg\min_{u \in \mathbb{R}^N} \frac{1}{2}\|u - z\|^2 \text{ s.t. } \|u\|_1 \leqslant k \text{ and } \forall\, i, -1 \leqslant u_i \leqslant 1$$

The minimizer of $\arg\min_u \frac{1}{2}\|u - z\|^2$ is reached for $u = z$, and we can write

$$u^{(s+1)} = \text{sign}(z) \arg\min_u \frac{1}{2}\||u| - z\|^2.$$

Furthermore, since the $\|\cdot\|_1$ is invariant with respect to the sign, we can write the minimization problem as

$$|u^{(s+1)}| = \arg\min_{u \in \mathbb{R}^N} \frac{1}{2}\||u| - z\|^2 \text{ s.t. } \|u\|_1 \leqslant k \text{ and } \forall\, i, 0 \leqslant u_i \leqslant 1$$

and then $u^{(s+1)} = \text{sign}(z)|u^{(s+1)}|$. The above minimization problem is a variant of the well-known knapsack problem and can be solved using a classical minimization scheme such as [Ste04] :

$$|u^{(s+1)}| = \arg\min_{u \in \mathbb{R}^N} \frac{1}{2} < u, u > - < u, |z| > \quad \text{s.t. } \left(\sum_i u_i\right) \leqslant k \quad \text{and } \forall\, i, 0 \leqslant u_i \leqslant 1$$

Secondly, we work with the penalized formulation of G_ρ, PeBic, with $I(u)$ on the penalized form (3.7). We write the problem as

$$u^{(s+1)} = \arg\min_{u\in\mathbb{R}^N} \lambda\|u\|_1 + \frac{1}{2b^{(s)}}\|u-z\|^2 + \chi_{-1\leqslant\cdot\leqslant 1}(u).$$

The solution is reached for

$$(u^{(s+1)})_i = \begin{cases} 1 \text{ if } z_i \in [1+\lambda b^{(s)}, \infty) \\ z_i - \lambda b^{(s)} \text{ if } z_i \in (\lambda b^{(s)}, 1+\lambda b^{(s)}) \\ 0 \text{ if } z_i \in \lambda b^{(s)}[-1,1] \\ z_i + \lambda b^{(s)} \text{ if } z_i \in (-1-\lambda b^{(s)}, -\lambda b^{(s)}) \\ -1 \text{ if } z_i \in (-\infty, -1-\lambda b^{(s)}] \end{cases}.$$

The proof is given in Appendix B.

3.3.4 *Small numerical examples*

In this section, we test the proposed algorithms in a small dimension to better understand them. Let $A \in \mathbb{R}^{2\times 2}$ and $d \in \mathbb{R}^2$

$$A = \begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix} \Lambda_{1/\|a_i\|} \quad \text{and } d = \begin{pmatrix} 1 \\ 1.5 \end{pmatrix} \tag{3.23}$$

where $\Lambda_{1/\|a_i\|}$ is a diagonal matrix with $\frac{1}{\|a_i\|}$ on its diagonal, and $\|a_i\|$ is the norm of the ith column of A. Note that each column does not need to be normalized, such as in the previous chapter. $\rho^{(0)} = 0.02$, and x and u are initialized in zero.

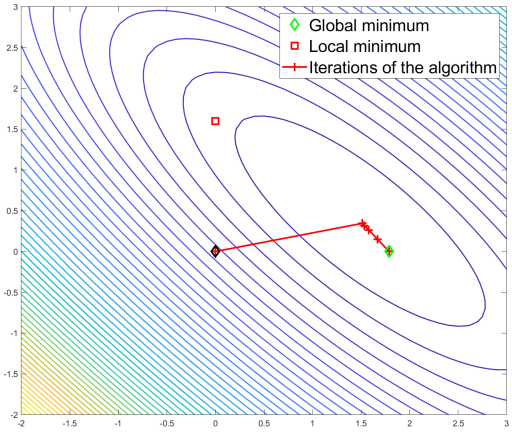


Figure 3.1: Minimization using CoBic

In Fig. 3.1, each step when ρ is updated is plotted over the level lines. The black diamond spots are the initial and finish points. CoBic

converges to the global minimum. Note that at the first iteration, CoBic converges towards the global minimum of the data fidelity term, and only when ρ grows, the algorithm converges to a sparse solution. Figure 3.2 shows the minimization using PeBic for four different values of λ. For all λ, except when λ = 1, the algorithm converges to the global minimum. The PeBic has the same behaviour as CoBic. The algorithm converges to the minimum of the data fidelity term when ρ is small. When ρ increases, we observe that the algorithm converges towards something more sparse, and this despite it being already in a local minimum.

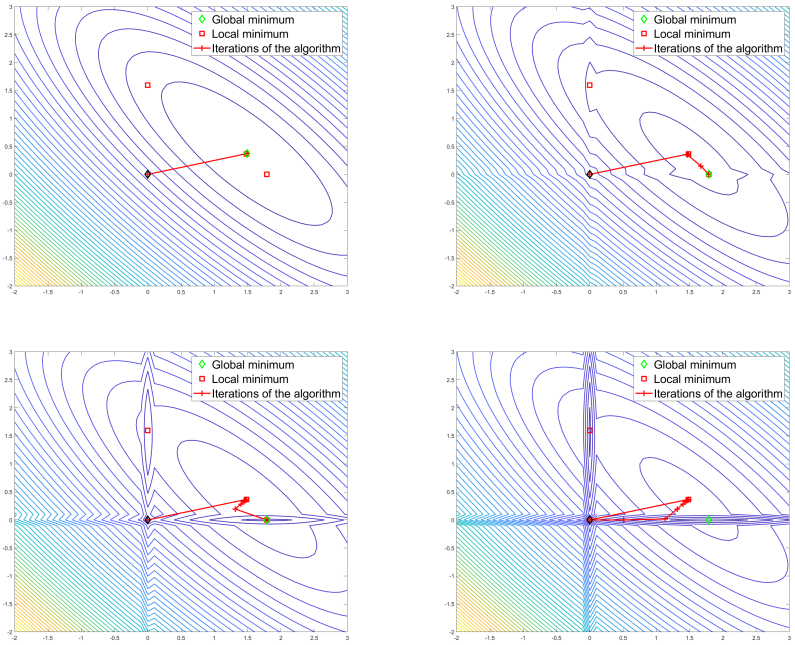


Figure 3.2: Minimization using PeBic. From left to right, top to bottom $\lambda = 0.001$, $\lambda = 0.1$, $\lambda = 0.5$ and $\lambda = 1$

3.4 CONCLUSION

In this chapter, we have presented a reformulation of the $\ell_2 - \ell_0$ constrained and penalized problems. We have proved in Theorem 3.1 and Theorem 3.4 the exactness of the reformulations, that is, we can from a minimizer of the reformulation obtain a minimizer of the initial problem. Furthermore, both reformulations are biconvex. Using two central properties of the reformulation, we derive a general algorithm in order to minimize the constrained or the penalized reformulation. This algorithm is easy to implement as each step can be decomposed to well-studied problems. In Chapter 5, we apply the algorithms to Single-Molecule localization microscopy.

4

A MULTIPLICATIVE CRITERION

Contents

This chapter includes an ongoing research project. It does not focus on minimizing the constrained formulation, nor the penalized one, but a new formulation. This method is somewhat less intuitive than the standard approaches but has compelling reasons for its use. Most importantly, the method is, in theory, parameter-free! As the thesis's goal is to develop sparse minimization schemes with easy-to-choose-parameters, this method should be the holy grail. However, there is no such thing as a free lunch, and finally, no such thing as parameter-free sparse optimization.

To start the chapter, we introduce the formulation and gives the arguments to convince the reader of its favorable properties. We propose a minimization algorithm to deal with the new cost function. We finish this chapter by giving some plots of the cost function and discuss the function.

4.1 INTRODUCTION

Let us at first forget sparse optimization for this section, and we go back to the standard formulation of inverse problems. As presented in the introduction of this thesis, Section 1.1.2, inverse problems are often on the form:

$$G(x) = f(x) + \lambda R(x). \tag{4.1}$$

In the introduction of this thesis, $f(x)$ is defined as the ℓ_2 data fidelity term but here it can be any loss function. $R(x)$ is a regularization term. The idea of adding a regularization term is an old one. In the article [Abu+04; ADS17], the authors propose to minimize:

$$J(x) := f(x)R(x). \tag{4.2}$$

The idea is not as intuitive as the standard formulation. Inevitably, minimizing $G(x)$ (4.1) yields an argument that minimizes the sum of

the data fidelity term and the regularization. The multiplicative is not so much different. The solution will be an argument that minimizes both functions, or *either*. It is clear that an argument $\hat{x}$ such that $f(\hat{x}) = 0$ or $R(\hat{x}) = 0$ is a global minimum of J (4.2). So, quite unusual, we are mostly interested in the local minima of the $J(x)$. Otherwise, it would suffice to minimize either $f(x)$ or $R(x)$ alone.

Furthermore, the goal is to study the multiplicative formulation, and from this, be inspired to develop an adaptive minimization algorithm, such as in the [ADS17].

However, to really understand why this formulation is of interest, we develop the expression in the next section.

4.2 THE OPTIMAL CONDITION

In one dimension, the initial function we want to minimize is

$$J(s) = f(s)R(s). \tag{4.3}$$

Assume that both f and R are differential. The optimality condition is $J'(\hat{s}) = 0$, which yields, assuming $R(\hat{s}) \neq 0$,

$$J'(\hat{s}) = f'(\hat{s})R(\hat{s}) + f(\hat{s})R'(\hat{s}) = R(\hat{s})[f'(\hat{s}) + \frac{f(\hat{s})}{R(\hat{s})}R'(\hat{s})] = 0. \tag{4.4}$$

Let

$$\alpha(\hat{s}) = \frac{f(\hat{s})}{R(\hat{s})}, \tag{4.5}$$

then, if $R(\hat{s}) \neq 0$,

$$J'(\hat{s}) = 0 \Leftrightarrow f'(\hat{s}) + \alpha(\hat{s})R'(\hat{s}) = 0. \tag{4.6}$$

With this formulation, and supposing $\lambda = \alpha(\hat{s})$, the optimality condition is identical to the one of (4.1). Thus in one dimension, we propose the following minimization scheme:

$$\begin{cases} s^{(0)} \in \mathbb{R}, \alpha^{(0)} = \alpha(s^{(0)}) \\ \text{Repeat:} \\ s^{(p+1)} \in \arg\min_s f(s) + \alpha^{(p)}R(s) \\ \alpha^{(p+1)} = \alpha(s^{(p+1)}) \\ \text{Until convergence} \end{cases} \tag{4.7}$$

The algorithm can be viewed as a minimization of (4.1), where λ is updated at each minimization. However, instead of having a fixed λ chosen a priori, we have $\alpha(s)$, an *adaptive* regularization parameter. This feature is the primary advantage because it eliminates the need to choose lambda.

4.3 ADAPTION TO SPARSE OPTIMIZATION

Let us go back to sparse optimization. More precisely, let f be the ℓ_2 data fidelity term and R a sparsity term. The ℓ_0-norm may not be appealing as it is not continuous.

A possible candidate is the CEL0 function, which has been presented earlier in Section 1.4.1.2. However, the function is dependent on λ, which could be set to 1 to avoid having a parameter. The tight results that CEL0 has concerning minimizers are only valid for the standard additive formulation. Furthermore, CEL0 is not differential in 0. This will be dealt with later.

Let $\phi_{\mathrm{CEL0}}(s)$ be as defined in (1.20):

$$\phi_{\mathrm{CEL0}}(s) = 1 - \frac{a^2}{2}\left(|s| - \frac{\sqrt{2\lambda}}{a}\right)^2 \mathbf{1}_{|s|\leqslant\frac{\sqrt{2}}{a}} \tag{4.8}$$

We want to study a function with a similar shape as the CEL0 function, but with a parameter to approach the ℓ_0-norm, thus we investigate also ϕ_ϵ, defined in (4.9):

$$\phi_\epsilon(s) = 1 - \frac{1}{\epsilon^2}\left(|s| - \epsilon\right)^2 \mathbf{1}_{|s|\leqslant\epsilon}. \tag{4.9}$$

Let us first study:

$$J_{\mathrm{CEL0}}(s) = f(s)\phi_{\mathrm{CEL0}}(s). \tag{4.10}$$

Assume $\hat{s} \neq 0$, as ϕ_{CEL0} is not differential in 0, then we search

$$J'_{\mathrm{CEL0}}(\hat{s}) = 0.$$

We can expand the above expression such as in Section 4.2, and we have

$$f'(\hat{s}) + \alpha(\hat{s})\phi'_{\mathrm{CEL0}}(\hat{s}) = 0 \tag{4.11}$$

where

$$\alpha(\hat{s}) = \frac{f(\hat{s})}{\phi_{\mathrm{CEL0}}(\hat{s})}. \tag{4.12}$$

Furthermore,

$$\phi'_{\mathrm{CEL0}}(s) = \begin{cases} \mathrm{sign}(s)\sqrt{2}a - a^2 s & \text{if } 0 < |s| \leqslant \frac{\sqrt{2}}{a} \\ 0 & \text{if } |s| \geqslant \frac{\sqrt{2}}{a}. \end{cases} \tag{4.13}$$

Inserting (4.13) into (4.11), and we get

$$J'_{\mathrm{CEL0}}(\hat{s}) = \begin{cases} f'(\hat{s}) + \alpha(\frac{\sqrt{2}a}{|\hat{s}|} - a^2)\hat{s} & \text{if } 0 < |\hat{s}| \leqslant \frac{\sqrt{2}}{a} \\ f'(\hat{s}) + 0 & \text{if } |\hat{s}| \geqslant \frac{\sqrt{2}}{a}. \end{cases} \tag{4.14}$$

We introduce

$$\beta(s) = \begin{cases} \frac{f(s)}{\phi_{CEL0}(s)}\left(\frac{\sqrt{2}a}{|s|} - a^2\right) & \text{if } 0 < |s| \leqslant \frac{\sqrt{2}}{a} \\ 0 & \text{if } |s| \geqslant \frac{\sqrt{2}}{a}. \end{cases} \tag{4.15}$$

We have assumed $\hat{s} \neq 0$. We further assume that $f(s) \neq 0$. From (4.15), we observe that $\lim_{s\to 0} = +\infty$, and we define $\beta(0) = +\infty$. We can then define the optimality condition, assuming $\hat{s} \neq 0$, of J_{CEL0} as

$$f'(\hat{s}) + \beta(\hat{s})\hat{s} = 0 \tag{4.16}$$

which is equivalent of minimizing $f(s) + \frac{1}{2}\beta(\hat{s})s^2$. As $\beta(\hat{s})$ depends on the solution, a minimization scheme such as (4.7) could be proposed. Once again, there is no need to choose the regularization parameters, as this is updated for each iteration.

Similarly, with ϕ_ϵ (4.9), we can define our cost function J_ϵ:

$$J_\epsilon(s) = f(s)\phi_\epsilon(s). \tag{4.17}$$

We can use the same procedure as for J_{CEL0}, and we define

$$\beta(s) = \begin{cases} \frac{f(s)}{\phi_\epsilon(s)}\frac{2}{\epsilon}\left(\frac{1}{|s|} - \frac{1}{\epsilon}\right) & \text{if } 0 < |s| \leqslant \epsilon \\ 0 & \text{if } |s| \geqslant \epsilon \\ +\infty & \text{else.} \end{cases} \tag{4.18}$$

We propose to minimize J_ϵ by trying to solve

$$f'(\hat{s}) + \beta(\hat{s})\hat{s} = 0$$

using a minimization scheme such as (4.7).

4.4 N-DIMENSIONS AND ALGORITHM

In N-dimensions, the formulation does not change much. Both proposed relaxations are separable relaxations, and thus the adaptation to N-dimension is evident. Furthermore, we fix $f(x) = \frac{1}{2}\|Ax - d\|^2$.

We search to minimize

$$J(x) = \frac{1}{2}\|Ax - d\|^2 \left(\sum_i^N \phi(x_i)\right) \tag{4.19}$$

where ϕ is either ϕ_ϵ or ϕ_{CEL0}. Using the optimality conditions, such as in (4.16), we search the following optimality conditions:

$$\nabla\left(\frac{1}{2}\|A\hat{x} - d\|^2\right) + \Lambda_{\beta(\hat{x})}\hat{x} = 0, \tag{4.20}$$

where Λ is a diagonal matrix with $\beta(x)$ on its diagonal. Each component of β is defined as

$$\beta_i(x_i) = \begin{cases} \frac{\frac{1}{2}\|Ax-d\|^2}{\sum_i^N \phi(x_i)}\left(\frac{\sqrt{2}a}{|x_i|} - a^2\right) & \text{if } 0 < |x_i| \leqslant \frac{\sqrt{2}}{a} \\ 0 & \text{if } |x_i| \geqslant \frac{\sqrt{2}}{a} \\ +\infty & \text{else,} \end{cases} \tag{4.21}$$

or

$$\beta_i(x_i) = \begin{cases} \frac{\frac{1}{2}\|Ax-d\|^2}{\sum_i^N \phi(x_i)}\frac{2}{\epsilon}\left(\frac{1}{|x_i|} - \frac{1}{\epsilon}\right) & \text{if } 0 < |x_i| \leqslant \epsilon \\ 0 & \text{if } |x_i| \geqslant \epsilon \\ +\infty & \text{else,} \end{cases} \tag{4.22}$$

depending on the function ϕ.

Given (4.20), we search to minimize, for a fixed β,

$$\min_x J_m(x, \beta) := \frac{1}{2}\|Ax - d\|^2 + \frac{1}{2}\|\Lambda_{\sqrt{\beta_i}} x\|^2 \tag{4.23}$$

where $\Lambda_{\sqrt{\beta_i}}$ is a diagonal matrix with $\sqrt{\beta_i}$ is on the diagonal. We propose the following algorithm.

Algorithm 4 : Multiplicative functional

Input :

$(x^{(0)}, \beta^{(0)}) \in \mathbb{R}^N \times \mathbb{R}^N$;

Repeat :

Solve the following problem:

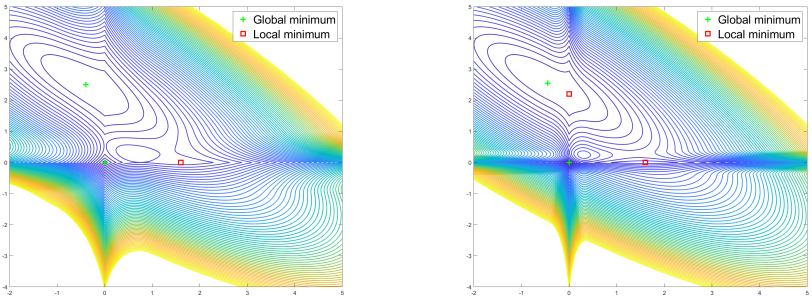
$$x^{(p+1)} \in \arg\min_x J_m(x, \beta^{(p)}) \tag{4.24}$$

Update $\beta^{(p+1)}$ in either (4.21) or (4.22).

Until : Convergence

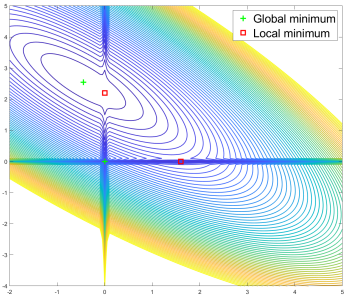
Output : $x^{(p+1)}$

There are many methods to minimize (4.24). For example, we can use the gradient step algorithm, with step size smaller than $\frac{1}{L}$, where L is the gradient Lipschitz constant of $J_m(x, \beta^{(p)})$ (4.23). In this case $L \geqslant \|A\|^2 + \|\Lambda_{\sqrt{\beta_i^{(p)}}}\|^2$, where $\|\Lambda_{\sqrt{\beta_i^{(p)}}}\|^2$ is the squared of the largest value on the diagonal. A problem arises when some $x_i = 0$, as then, $\beta(x_i) = +\infty$, and the Lipschitz gradient constant is $+\infty$. To avoid this problem, we define $\tilde{\beta}(x) = \beta(|x| + \delta)$, with β defined as in (4.21) or (4.22), where $\delta \approx 10^{-10}$.

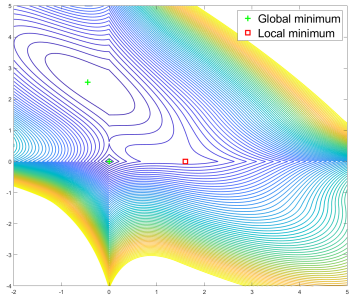


(a) $J_{\epsilon=1}$

(b) $J_{\epsilon=0.4}$



(c) $J_{\epsilon=0.1}$



(d) J_{CEL0}

Figure 4.1: Top: Level lines of the cost function J_ϵ, with to the left, $\epsilon = 1$ and to the right, $\epsilon = 0.4$. Bottom: Left: J_ϵ, with $\epsilon = 0.1$. Right: J_{CEL0}

4.5 NUMERICAL EXAMPLES

In this section, we plot the level lines of the cost function, J (4.19), using either the ϕ_ϵ and the ϕ_{CEL0}. These numerical examples show why ϕ_ϵ may be better to use.

$$A = \begin{pmatrix} 3 & 2 \\ 1 & 3 \end{pmatrix} \Lambda_{1/\|a_i\|} \quad \text{and } d = \begin{pmatrix} 1 \\ 2 \end{pmatrix} \tag{4.25}$$

The level lines of each of the cost functions (see Fig. 4.1) show the global and local minima. As mentioned, the global minima are situated in the point that minimizes the ℓ_2 data fidelity term and in $(0,0)^T$ as $R((0,0)^T) = 0$. In the case of CEL0, there is only one local minimum. This happens for ϕ_ϵ, with ϵ equal to 1, and this can be observed in Fig. 4.1d and 4.1a. If ϵ is smaller than 0.5, then there are two local minima. It is only when we are close to the axis that the regularization function, with a small ϵ, impacts the cost function, see Fig. 4.1b and 4.1c. Thus, a natural question and a problem is the issue of how to chose ϵ.

Furthermore, let us look at how a minimization algorithm act when minimizing the function J_ϵ. If the initial point is in a point such that the regularization function is not active, then we will obtain the minimum of the cost function. Why? When the regularization function is

not active, it is constant to 1, and $b = 0$ in that case. Thus β, defined in (4.21) or (4.22) is equal to 0. The first iteration of the algorithm (4.24) minimizes the sum of the ℓ_2 data fitting term and the square norm of the Hadamard product of x and $alpha$, see (4.23). Since $\beta = 0$, we minimize only the data fitting term in the first iteration, and the minimizer is a global minimizer of the multiplicative term.

As we can observe in Fig. 4.2, the choice of starting point is crucial. If we are at a point with a value larger than ϵ, the corresponding β will be zero. To avoid this, we must make sure that the starting point has values smaller than ϵ. However, we cannot be too close to zero, as this is a global minimum. This is clear in 2-dimensions, but it remains to study closer how the algorithm behaves in N-dimensions.

4.6 CONCLUSION

There is no such thing as free lunch and no such thing as a parameter-free sparse optimization algorithm. We have studied the minimization of the product of the data fidelity term and a sparsity term. From this, we propose an algorithm where we have an adaptive regularization parameter. This is interesting, seeing that choosing a regularization term is often very time-consuming.

We have seen that the algorithm, in 2-dimensions, depends on the choice of regularization term as well as the initialization. The multiplication minimization is an ongoing research area, and the results are too preliminary to include in the application part of the thesis.

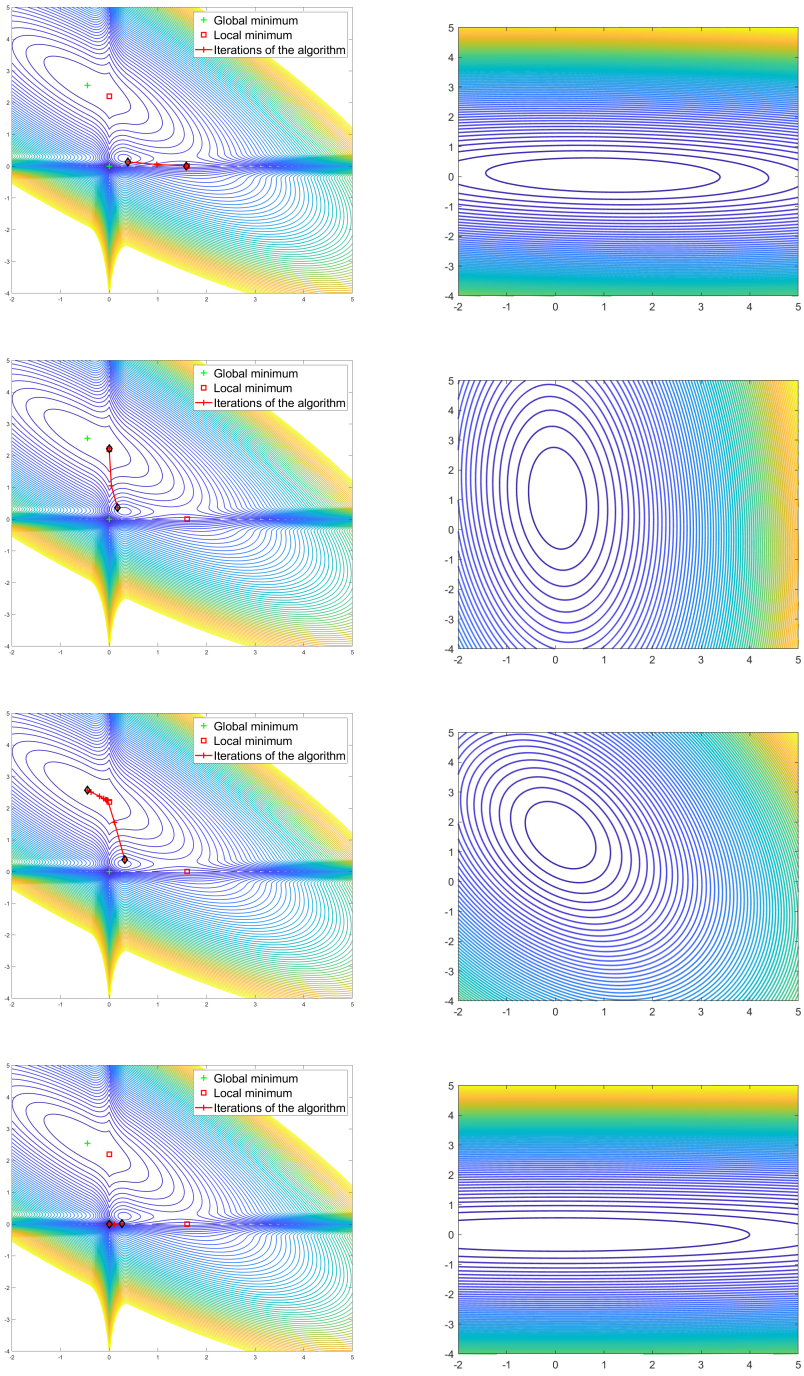


Figure 4.2: Left column: The cost function $J_{\epsilon=0.4}$, and the convergence of the algorithm depending on the initial position. Right column: The cost function of (4.23), with the β defined from the initial point.

Part II

APPLICATION

"Mathematics is applied by everyone except applied mathematicians"

—[Wil09]

5

SINGLE-MOLECULE LOCALIZATION MICROSCOPY

Contents

In this chapter, we give an introduction to SMLM. We start with an overview of classical microscopy and the advances leading to SMLM. We introduce SMLM and the challenges appearing in this microscopy method. We present a method to model the acquisition system, and we show that this is a sparse optimization problem. We introduce the evaluation tool, and perform numerical experiments of the methods proposed in Chapter 2 and Chapter 3, and compare them to other $\ell_2 - \ell_0$ based algorithms. We first compare the methods on a small numerical example. Further, we compare the algorithms with the state of the art of 2D SMLM grid-based algorithms on two datasets from the ISBI 2013 SMLM challenge [Sag+15]. A more recent challenge was launched in 2016 [Sag+19]. We decided to use the 2013 challenge as the data are denser in the 2013 challenge (220 fluorophores per acquisition in 2013 compared to 12 in the 2016 challenge). Furthermore, the 2D data in the 2016 challenge contains observations where some elements are far from the focal plane[1]. Thus the image formation model presented in Section 5.3 is not optimized for this image acquisition method. The algorithms are coded on MATLAB2019 with a computer

1 The data in 2016 has a sample depth of 1.5 micrometers compared to the data from 2013, with a sample depth of 300 nm.

running on Linux, with CPU INTEL core i7-3920XM, except DeepSTORM that was launched on a computer running on Linux, with CPU Intel Xeon E5-2687WV3.

The codes for PeBic, CoBic and the minimization of G_Q, as well as an example, can be found on
https://github.com/abechens/CoBic-and-PeBic-SMLM/ and
https://github.com/abechens/SMLM-Constraint-Relaxation.

5.1 INTRODUCTION

Microscopes generate magnified images. They have been used since the late 17th century and have been crucial in understanding the world around us. The constant wish to better explain the mysteries of life demands for more and more efficient microscopes. In microscopy, efficiency may have multiple meanings. The most prominent one is the resolution. Lord Rayleigh defined in 1896 resolution as the shortest distance between two specimens such that an observer can distinguish them as separate entities. The definition is built on the work of George Biddell Airy and Ernst Abbe. Airy is, among other things, known for the Airy disk. He first wrote about the phenomenon in a journal for astronomy, but the same phenomenon arrives in microscopy as well. The Airy disk is a diffraction problem where instead of observing one small point light source, we observe a pattern, or an Airy disk, see Fig. 5.1.

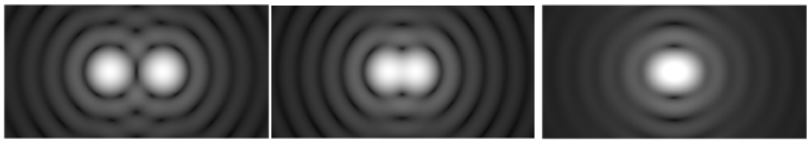


Figure 5.1: The Airy disk for two points emitting light, with different distances between them. Source [Com14].

Ernst Abbe developed the equation of the diffraction in 1873, which gives the diffraction limit of a microscope in the lateral plane. The limit is given as

$$d_{min} \approx \frac{\lambda}{2\,NA} \tag{5.1}$$

where λ is the wavelength of the light and NA is the numerical aperture of the optical system [LLL10]. Using the diffraction limit, Rayleigh defined more precisely the resolution as

$$r_{min} \approx 0.61 \frac{\lambda}{NA}. \tag{5.2}$$

Figure 5.2 illustrates this resolution limit.

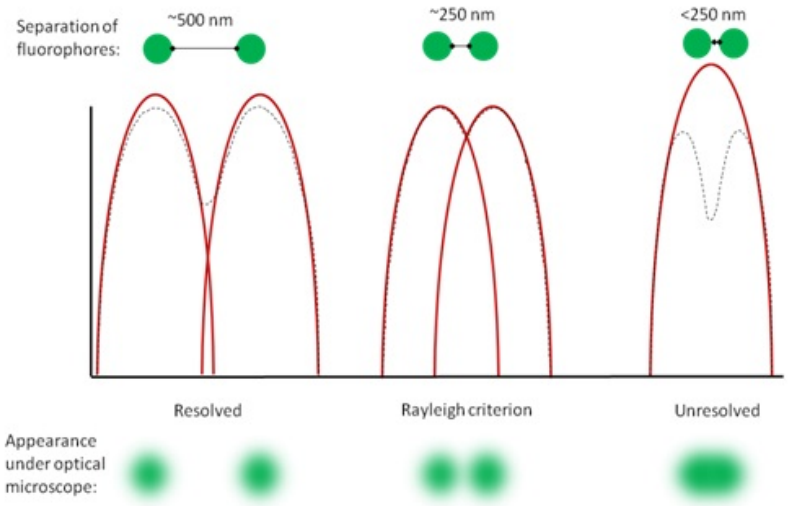


Figure 5.2: Resolution and the Rayleigh criterion. Point sources can be resolved if the are separated by more than the Rayleigh criterion (5.2). [2]

If we use the visible blue light at around 500 nm, with a numerical aperture of 1.3, the Rayleigh resolution is 235nm.

This limit poses a problem for scientists as this is not sufficient when observing small biological structures, such as proteins and viruses, see Fig. 5.3. An electron microscope uses electrons instead of the visible light and reaches a resolution up to 0.2nm. However, the samples must be pretreated and fixed, and in-vivo imaging (images of living structures) is then impossible.

Several methods bypass the resolution limit, such as Stimulated Emission Depletion (STED) [KH99], Structured Illumination Microscopy (SIM) [Gus00], Super-Resolution Radial Fluctuations (SRRF) [Gus+16], and Single-Molecule Localization Microscopy (SMLM) [HGM06; Bet+06; RBZ06]. See [Sch+19; ML19] for a recent overview of these and other methods. They have in common that they are based on fluorescence microscopy. In contrast to traditional microscopy methods, which use light to illuminate the sample, fluorescence microscopy collects the light in another wavelength range than the excitation. Fluorophores are, by definition, molecules that emit light after excitation by light (contrary to, for example, bioluminescence and chemiluminescence). The fluorophores absorb some of the energy and reach an unstable level. Thus the fluorophore will release some of this energy as light. It is important to note that not all the energy absorbed is emitted by the fluorophore. Thus it is possible to distinguish between the light

2 Reprinted from Fluorescence Microscopy , 1, Jennifer A. Thorley, Jeremy Pike,Joshua Z. Rappoport (Editors: Anda Cornea P. Michael Conn), Chapter 14 Super-resolution Microscopy A Comparison of Commercially Available Options , 199-212, 2020, with permission from Elsevier

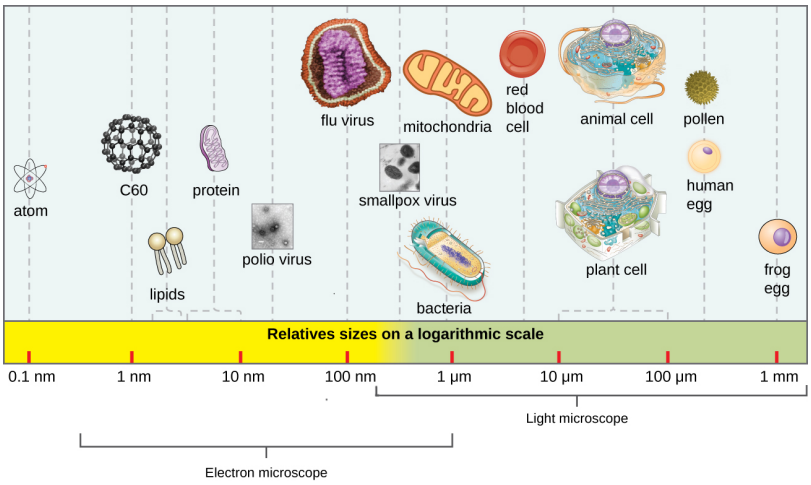


Figure 5.3: The limits of classical light microscopy. Source [Mic16]

emitted by the fluorophore and the illumination light source. A sensor captures the emitted light from the fluorophores.

Furthermore, SMLM uses photoswitchable fluorophores. These are fluorophores that can be controlled to be in a bright and dark state using a laser.

Many subjects of interest for biologists and doctors are not by nature fluorescent. Thus they "dye" the subject with fluorophores, i.e., insert the fluorophores artificially. This can be done by, for example, using immunofluorescence, which uses fluorescent antibodies that binds itself to the protein.

In this thesis, we focus on SMLM.

5.2 SINGLE-MOLECULE LOCALIZATION MICROSCOPY

Single-Molecule localization microscopy is an acquisition method that allows to obtain images with a higher resolution than the diffraction limit. 20 nm resolution is reported on SMLM when it was first introduced in [HGM06; Bet+06; RBZ06] under the names Fluorescence photoactivation localization microscopy (FPALM), Photoactivated Localization Microscopy (PALM), Stochastic Optical Reconstruction Microscopy (STORM). The methods can be used to observe fine structures. The work was rewarded the Nobel Prize in Chemistry in 2014.

The idea behind SMLM is quite simple. Let say we only observe one fluorophore in the microscope, and the observation is without noise. We can then precisely localize the fluorophore despite the airy disk by assuming it to be in the disk's center. Now, imagine we observe only a few fluorophores in the microscopes, and that they distributed far from each other. As long as the distance is greater than the Rayleigh criterium, the localization is not too tricky (assuming no noise). To take advantage of the "easy" localization when observing only a few molecules, SMLM uses photoactivatable fluorophores. This allows one

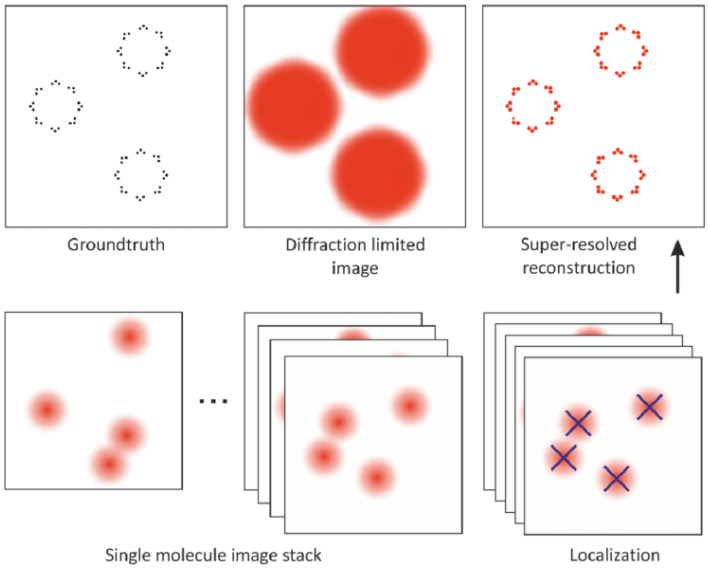


Figure 5.4: The principles of SMLM. Instead of all the fluorophores emit light at the same time (Diffraction limited image), only sparse subsets emit light (Single molecule image stack), and the fluorophores are precisely localized (Localization). The sum of all the localizations creates one super-resolved image (Super-resolved reconstruction). Source: [Hei20]

.

to activate a sparse subsample of the fluorophores using a weak activation light, acquire an observation, and then "turn the previous active fluorophores off" afterward. In practice, the fluorophores emit light until photobleaching. After that, the fluorophore is permanently unable to emit light. Then SMLM repeats this procedure until a sufficient number of fluorophores have been activated. Thus, each observation is a result of only a few fluorophores, and precise localization is possible. An SMLM image is the sum of all the precisely located fluorophores and obtains super-resolution. Note that classical fluorescence microscopy illuminates all the fluorophores simultaneously. See Figure 5.4 for a graphical example. However, acquiring all the low-density images takes time, and the sample may move during this time. Thus the temporal resolution of the image may be inadequate. High-density acquisitions reduce the total acquisition time and increase temporal resolution. However, now, the fluorophores may be too close to each other, and thus efficient localization algorithms are needed.

5.2.1 *State of the Art*

The ISBI 2013 [Sag+15] and 2016 [Sag+19] SMLM challenges address the localization problem in SMLM. The 2013 challenge focused on 2D reconstruction and images, while the 2016 SMLM challenge focus on

both 3D and 2D reconstruction. Both challenges receive a large number of algorithms, and the results are compared to different datasets. The state of the art of 2D localization algorithms presented here is a summary of what can be found in [Sag+19].

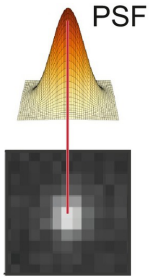


Figure 5.5: Center of gravity localization method. Adapted from [Wu+20]

There are many possibilities to solve the localization problem, and some algorithms prioritize fast computational time with a loss of precision. These methods localize spots in the observed image and fit specific shapes (templates, PSF) on these spots [Tak+18], or find the center of mass in the spot [Hen+10], see Fig. 5.5.

More sophisticated algorithms, but also more computational costly, test the likeness between the spot and a sum of different Gaussians on each spot [BSZ12].

Another approach consists of modeling the localization problem as an inverse problem with a sparsity term. These methods can be divided into two groups: Grid methods and grid-less methods. The grid methods require a fine grid to obtain a sufficient precision, but at the cost of computational time. Grid-less algorithms do not have this problem, but the observation model is non-linear. See [BSR17; Min+14]. With a grid, this observation model is linear. Some algorithms of the inverse problem approach are SMLM-CEL0 and L1H [GSBF17; Bab+13].

Finally, deep-learning methods are emerging as well, with promising results [STM19; Boy+18; Neh+18].

In this thesis, we focus on grid-based methods. In the following section, we describe how the localization problem can be formulated as a sparse optimization problem.

5.3 MATHEMATICAL MODEL

5.3.1 *The image formation model*

Let $d \in \mathbb{R}^{M \times M}$ be an image acquisition. We suppose that only a small number of fluorophores have contributed to this image. We want to localize the fluorophores on a finer grid $x \in \mathbb{R}^{ML \times ML}$, where L is the refinement factor. The goal is to reconstruct x from d. To do so, we need to model the acquisition process.

The fluorophores are observed through an optical system, and thus we observe diffraction discs (Airy disks) instead of the fine position

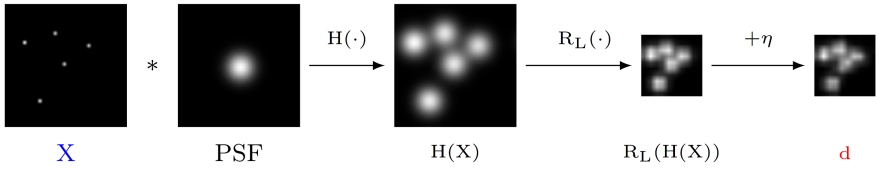


Figure 5.6: The image formation model. The fluorophores are on the fine grid X, convolved with the PSF, and downsampled. We assume additive noise.

of the fluorophores. This is modeled by a convolution with the Point Spread Function (PSF) of the microscope. We suppose the PSF to be a Gaussian kernel:

$$\mathrm{PSF}(z, y) = I \exp\left(-\frac{z^2 + y^2}{2\sigma^2}\right)$$

where I is a normalization factor. Furthermore, a sensor captures the observation with a resolution inferior to the fine grid. We model this as an operator that sums pixel groups of $L \times L$. The result is an observation of size $M \times M$. Finally, this observation is affected by noise η, which is assumed to be a mix of Poisson noise and Gaussian noise. We simplify the noise assumptions and consider only additive white Gaussian noise.

The convolution operation is either noted $h \star x$, h being the convolution kernel, or noted $H : \mathbb{R}^{ML \times ML} \to \mathbb{R}^{ML \times ML}$. The downsampling operator is noted noted $R_L : \mathbb{R}^{ML \times ML} \to \mathbb{R}^{M \times M}$. Thus, the model can be written, in terms of linear algebra, as

$$d = Ax + \eta$$

where $A \in \mathbb{R}^{M^2 \times (ML)^2}$ is the matrix that performs a convolution and downsampling. The image formation model is graphically explained in Fig. 5.6.

Note that the sensor that captures the signal follows the Shannon-Nyquist theorem. Most sensors have pixels of size 100 nm times 100 nm. The Rayleigh criterium gives the lowest possible resolution of light microscopy around 250 to 200 nm. Following Shannon-Nyquist, to correctly reconstruct the signal, it must be sampled with half the wavelength, thus around 100 nm.

5.3.2 *Formulation of the inverse problem*

With the assumption of Gaussian noise we can write the recovering of x as

$$\arg\min_x \frac{1}{2}\|Ax - d\|_2^2.$$

However, A is a matrix with more columns than lines, and thus the problem is underdetermined and ill-posed. Some *a priori* knowledge

of x is needed to correctly localize the fluorophores. The first hypothesis is that only a few fluorophores are excited and emit light. Thus the solution should be *sparse*. We can use the ℓ_0-norm to enforce sparsity. The second hypothesis is that the fluorophores emit light, and we wish to reconstruct the intensity, which is positive. We add, therefore, that the solution should be non-negative. This yields that we can search a solution $\hat{x}$ as

$$\hat{x} \in \arg\min_x G_{k+}(x) := \frac{1}{2}\|Ax - d\|_2^2 + \chi_{\cdot \geqslant 0}(x) \text{ s.t. } \|x\|_0 \leqslant k \quad (5.3)$$

$$\hat{x} \in \arg\min_x G_{\ell_0+}(x) := \frac{1}{2}\|Ax - d\|_2^2 + \lambda\|x\|_0 + \chi_{\cdot \geqslant 0}(x) \quad (5.4)$$

where χ_X is the indicator function, and $\chi_{\cdot \geqslant 0}(x)$ enforces the positivity constraint. We use also the following function

$$\mathrm{dist}^2_{\mathbb{R}_{\geqslant 0}}(x) := \inf_{y \geqslant 0} \frac{1}{2}\|x - y\|^2$$

to promote positivity when the hard constraint is not possible to implement. We do however, continue the same notation (G_{k+}):

$$\hat{x} \in \arg\min_x G_{k+}(x) := \frac{1}{2}\|Ax - d\|_2^2 + \mathrm{dist}^2_{\mathbb{R}_{\geqslant 0}}(x) \text{ s.t. } \|x\|_0 \leqslant k. \quad (5.5)$$

As seen in the introduction of the thesis, Section 1.2, the problems differ in the regularization. The constrained problem reconstruct at maximum k non-zero components. The regularized problem reconstructs a sparse solution, and the number of non-zero elements depends on the regularization parameter λ and the acquisition d. In an ideal case, where the fluorophores are sufficiently separated, one fluorophore would represent a non-zero component in x. Thus the parameter k represents the maximum number of fluorophores to reconstruct for each acquisition. Indeed, in practice, this is not the case since we work mainly with high-density acquisitions, and multiple fluorophores can be situated in one pixel, even on the fine grid. Thus the parameter k will represent the number of *distinguishable* fluorophores in the reconstruction.

5.4 QUANTITATIVE PERFORMANCE ASSESSMENT

Most signal-processing researchers are familiar with classical evaluation methods such as Peak Signal-to-Noise ratio (PSNR), Mean squared error (MSE), and structural similarity index measure (SSIM). However, we use the Jaccard index to perform the numerical evaluation of the reconstructions. The Jaccard index evaluates the localization of the reconstructed fluorophores (see [Sag+15]). It is defined using correctly reconstructed (CR)-, false negatives (FN)- and false positives (FP) fluorophores. An FP is when we reconstruct a fluorophore that should not be there, and an FN is when we do not reconstruct a fluorophore where it should be. The Jaccard index is the

ratio between the CR fluorophores and the sum of CR-, FN- and FP fluorophores. A perfect reconstruction yields an index of 100, and the lower the index, the poorer the reconstruction.

$$\mathrm{Jac} = \frac{\mathrm{CR}}{\mathrm{CR} + \mathrm{FP} + \mathrm{FN}} \times 100. \tag{5.6}$$

Furthermore, the Jaccard index includes an error tolerance,Δ , such that a reconstructed fluorophore does not have to be at the exact correct position, but in a small neighborhood, see Fig. 5.7. Methods such as MSE and PSNR compare pixel with pixel. Thus, a fluorophore situated in a pixel close to the ground truth will have the same error as one fluorophore far from the ground truth. Furthermore, when using

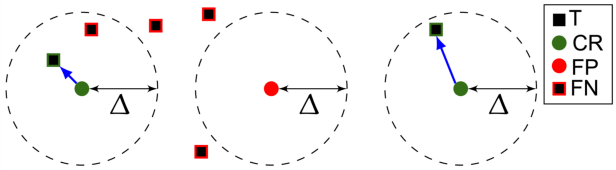


Figure 5.7: A graphic example of evaluation. The blue arrow shows how a reconstructed fluorophore is linked to the ground truth (T), represented as a black square. The figure also shows the use of the error parameter, Δ.

the Jaccard index, the number of reconstructed fluorophores should be equal.

5.5 SINGLE IMAGE PERFORMANCE

In this section, we are interested in how well the proposed algorithms can successfully reconstruct one image. We construct an image artificially with 213 of fluorophores randomly scattered on a 256×256 grid, where each square measures 25×25 nm. The observed image is 64×64-pixel image, where each pixel measures 100×100 nm, with a simulated Gaussian PSF with an FWHM of 258.21nm. Note that we use these parameters as this is representative of the simulated 2D-ISBI data presented in the next section. The 100 observations are then created by applying different realizations of Poisson noise to the same image.

The methods are compared to four methods; the CEL0 [GSBF17], Constrained Iterative Hard Thresholding (C-IHT), Penalized Iterative Hard Thresholding (P-IHT) [CW05] and the ℓ_1-relaxation. The algorithms used, as well as the best initialization obtained is listed in Table 1 for the constrained-based methods, and 2 for the penalized-based methods.

3 Note that we use this algorithm for the small numerical example, in order to compare G_k and G_Q fairly. For the ISBI data, the FBS algorithm is applied.

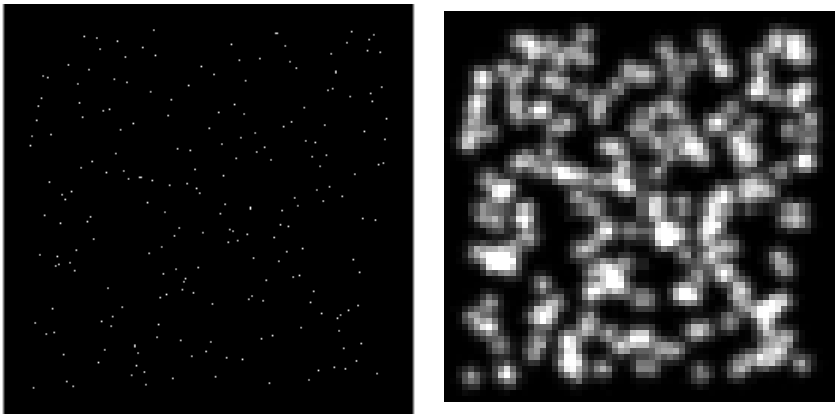


Figure 5.8: Example of the simulated dataset. The number of fluorophores is 213. To the left: Ground truth. To the right: One of the 100 observations.

Table 1: The different constrained based methods with the algorithms and initializations used.

Method	G_Q	CoBic	C-IHT
Algorithm	nmAPG	Algorithm 3	nmAPG[3]
Initialization	$0 \in \mathbb{R}^{N\times N}$	$\rho^{(0)} = 0.1$ $x = u = 0 \in \mathbb{R}^{N\times N}$	$0 \in \mathbb{R}^{N\times N}$

Table 2: The different penalized-based methods with the algorithms and initializations used.

Method	PeBic	P-IHT	CEL0	ℓ_1
Algorithm	Algorithm 3	FBS	IRL1 [Och+15]	FISTA
initialization	$\rho^{(0)} = 0.1$ $x = u = 0 \in \mathbb{R}^{N\times N}$	$A^T d$	$A^T d$	$A^T d$

We use the IRL1 algorithm [Och+15] to minimize CEL0, the exact relaxation [SBFA15] of the penalized formulation (5.4). The ℓ_1-relaxation is minimized using FISTA.

The algorithms are compared with their initial function to demonstrate the advantages of each proposed method. Thus G_Q and CoBic are compared to C-IHT, and PeBic is compared mainly with P-IHT, but also CEL0 and ℓ_1-relaxation.

The P-IHT, ℓ_1-relaxation, and PeBic have the trade-off parameter λ to choose. G_Q, CoBic, and the C-IHT algorithm use the sparsity parameter k. This parameter ensures that the algorithm does not reconstruct more than k non-zero elements in the solution.

We compare the methods using both the ℓ_2 data fidelity term and the Jaccard index.

We want the algorithms to reconstruct the image as good as possible, knowing there are 213 fluorophores present.

Thus, to properly compare the minimization methods, we chose λ such that the average number of reconstructed pixels corresponds to the number of fluorophores in the image.

The comparison using only the ℓ_2 data fidelity may be unfavorable to the penalized methods, as they minimize the sum of the ℓ_2 data fidelity term and a regularization term. Furthermore, they are designed to be used when no prior knowledge of the sparsity is known.

As previously stated, choosing a λ is through a test- and fail- strategy, so we test if for a given λ we will obtain a sparse solution. To give an example: here we tested λ first equal to 1, 0.1, 0.01, 0.02, and 0.015 for the first observation, then we test the λ on the ten first observations. From these tests λ was adjusted to 0.025. In all, it took around 1 hour to find a proper λ. For CoBic, G_Q and C-IHT, we set simply the parameter k equal to 213.

5.5.1 *Results*

In this part, we show the obtained results. The results of the 100 image reconstructions are presented with boxplots. The red mark in the box is the median of the reconstruction result of the 100 noisy, blurred, and downsampled images. The upper (respectively lower) part of the box indicates the 75th (25th) percentiles median. An outlier is represented as a red +.

In Fig. 5.9, we compare the results of G_Q, CoBic, and C-IHT using the data fidelity term. We can observe that both G_Q and CoBic have lower values than C-IHT. Thus they are more efficient in solving the initial problem. Furthermore, CoBic has a median value of 0.99, which is lower than that of G_Q, 1.55. In comparison, C-IHT has 2.74 as a median value. It may seem that CoBic is far superior in resolving the constrained problem. That said, the algorithm does not always saturate the constraint. Among the 100 reconstructions, three of the re-

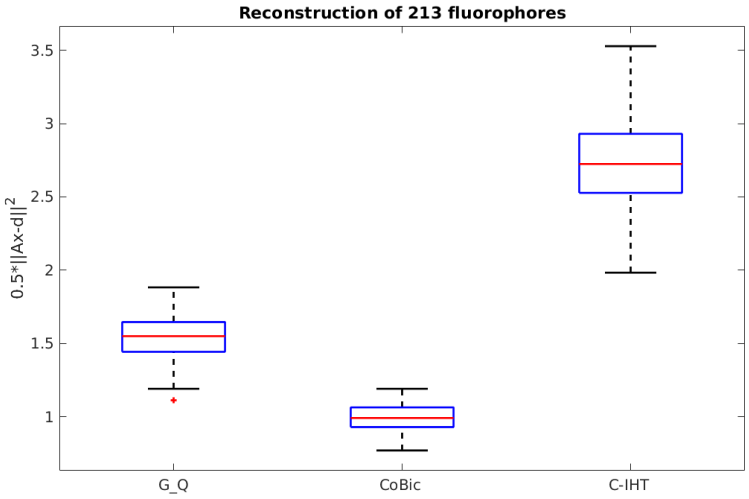


Figure 5.9: Comparison of the constrained-based algorithms: G_Q, CoBic, and C-IHT. The y-axis represent the value $\frac{1}{2}\|Ax - d\|^2$. The lower, the better.

construction contain only 212 fluorophores, and not 213. This may be counter-intuitive at first, but there is a possible reason. The algorithm has either converged to a possible saddle point or a critical point, in which it is not possible to add a positive pixel. This happens as well in the reconstruction of the ISBI 2013 simulated data on a larger scale, see section 5.6.3. Note that the data fidelity value of the three reconstructions that reconstructed 212 non-zero pixels instead of 213 non-zero pixels are less than the median value of CoBic presented in the Fig. 5.9.

Furthermore, we compare the penalized-based methods using only the data fidelity term. As explained in the previous section, we choose a λ such that the methods reconstruct on average 213 fluorophores. Figure 5.10 shows clearly the difficulties P-IHT has, as the values are far superior to those of PeBic and CEL0. The figure also shows that the ℓ_1-relaxation may not be an efficient relaxation of the ℓ_0-norm in this case. We are not assured of obtaining the global minima of the initial problem since the observation matrix does not satisfy the RIP conditions. Due to the poor reconstruction of the P-IHT and ℓ_1-relaxation, we add a plot of only PeBic and CEL0 in Fig. 5.11. Both methods perform as good as the constrained based methods G_Q and CoBic, with a median value of 1.04 for PeBic and 1.10 for CEL0. Furthermore, as expected, the variance of PeBic is slightly more significant than CoBic. CEL0 has a variance equal to the one of CoBic.

In Single-Molecule localization microscopy, the most important aspect is the precise positioning of the fluorophores. Thus the Jaccard index is a convenient performance measurement. The boxplot for the

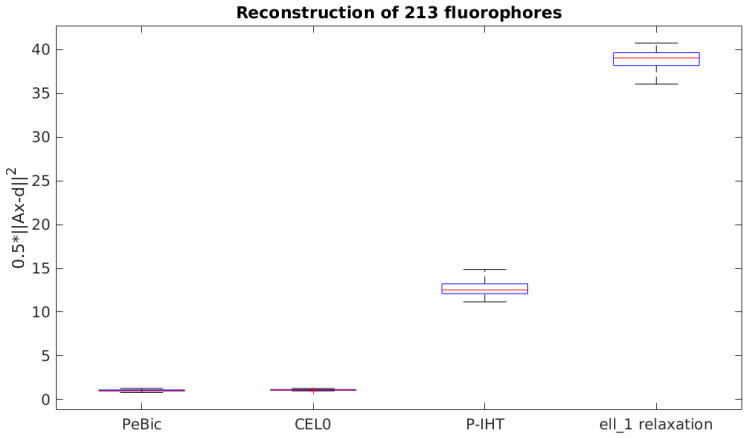


Figure 5.10: Comparison of the penalized-based algorithms: PeBic, CEL0, P-IHT and ℓ_1-relaxation. The y-axis represent the value $\frac{1}{2}\|Ax - d\|^2$. The lower, the better.

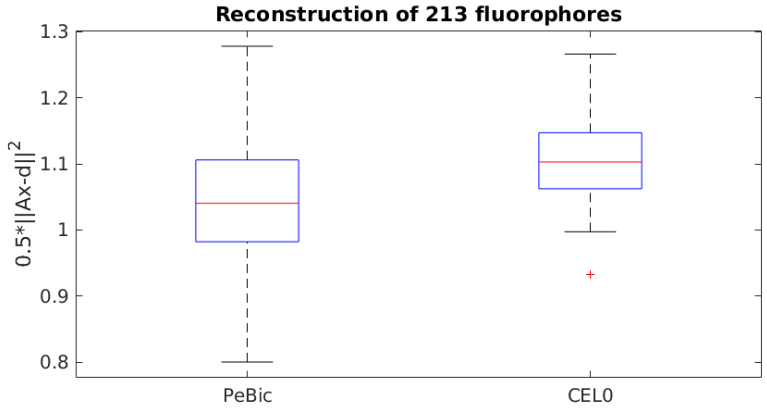


Figure 5.11: Comparison of the penalized-based algorithms: PeBic, and CEL0. The y-axis represent the value $\frac{1}{2}\|Ax - d\|^2$. The lower, the better.

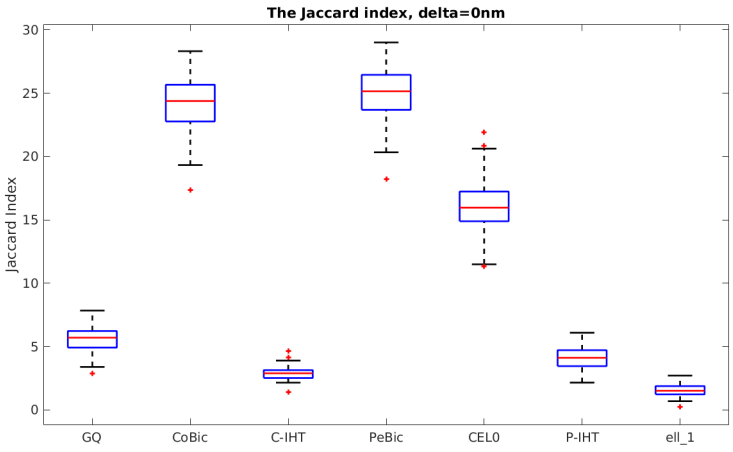


Figure 5.12: Comparison of the algorithms. The y-axis represent Jaccard index with a margin of error of 0nm. The higher, the better.

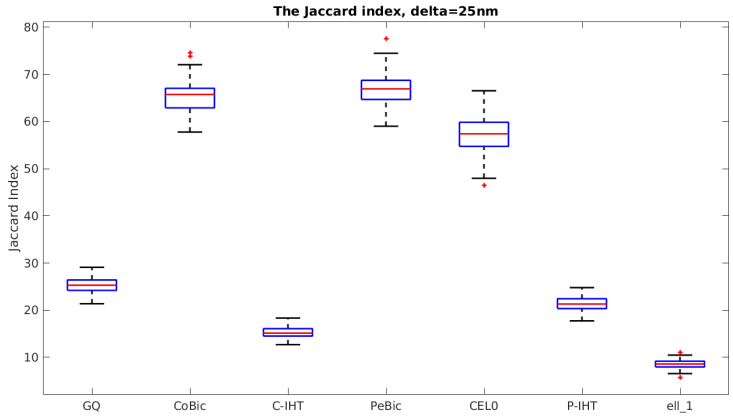


Figure 5.13: Comparison of the algorithms. The y-axis represent Jaccard index with a margin of error of 25nm. The higher, the better.

Jaccard index is plotted in Fig. 5.12 and Fig. 5.13. A perfect reconstruction yields a Jaccard index of 100.

In Fig. 5.12, we set the margin of error, δ, equal to 0. This means that we see how many fluorophores we reconstruct perfectly. We observe that CoBic, PeBic, and CEL0 stand out compared to the other methods. The intuition would be that a low data fidelity value would yield a good reconstruction. However, P-IHT yields a better reconstruction than C-IHT, even though C-IHT reconstructs better with respect to the data fidelity term. We discuss this discrepancy later in this section.

With a margin of error of 25nm, we can see the same tendencies as in the former, see Fig. 5.13.

Even though the proposed methods reconstruct quite precisely the 213 fluorophores, we are also interested in how the methods behave when demanding to reconstruct *less* than 213 fluorophores. The rea-

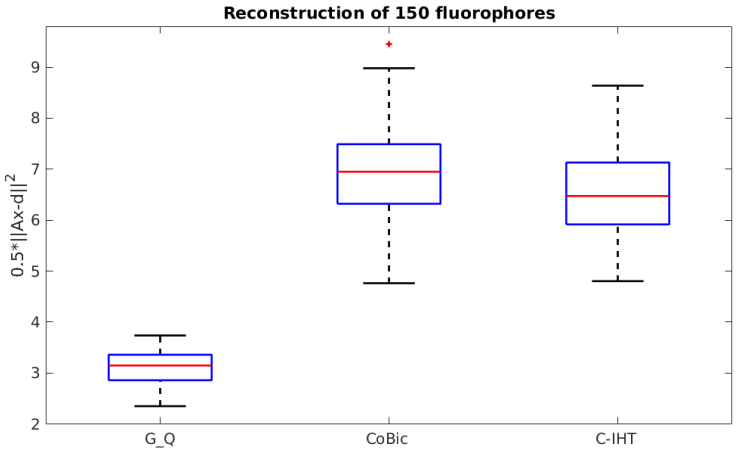


Figure 5.14: Comparison of the constrained-based algorithms: G_Q, CoBic, and C-IHT. The y-axis represent the value $\frac{1}{2}\|Ax-d\|^2$. The lower, the better.

son for this is more apparent in the section of the ISBI 2013 data and is discussed there, see Section 5.6.2. In any case, to know precisely the sparsity of a solution is rare, and the natural question is to ask what happens if we underestimate the total number of fluorophores.

The algorithms reconstruct now 150 non-zero element in each image. The results are significantly different for one algorithm. See Fig. 5.14. CoBic, which obtained the best reconstruction concerning the data fidelity term, perform significantly worse. It obtains higher values than both G_Q and C-IHT. PeBic has a similar change, as observed in Fig. 5.15, but in a less degree. G_Q reconstructs significantly better than CoBic and C-IHT.

Surprisingly, the algorithms CoBic and PeBic have both a Jaccard index higher than the other methods, see Fig. 5.16 and Fig. 5.17. A possible explanation for this discrepancy is that the methods converge to local minima, which correspond to good placements of the fluorophores. However, methods, such as G_Q, CEL0, may find solutions that are better concerning the data fidelity term, but not to the placement.

The following example demonstrates the problem well. Two fluorophores are placed on a distance of 80nm from each other. The captured signal can be observed in Fig. 5.18. *We want to locate only one fluorophore precisely.* If we reconstructed an image, should we place the fluorophore in the middle of both fluorophores? Or should we place it correctly on one of the two fluorophores positions? In terms of the cost of the data fidelity term, the choice is in the middle (cost for placing a non-zero pixel in the middle: 0.26 vs. cost for placing a non-zero pixel in one of the two perfect places: 0.94). However, in terms of the Jaccard index, choosing one correct place is better. Finally, this is a discussion without a final answer. As a mathematician, we

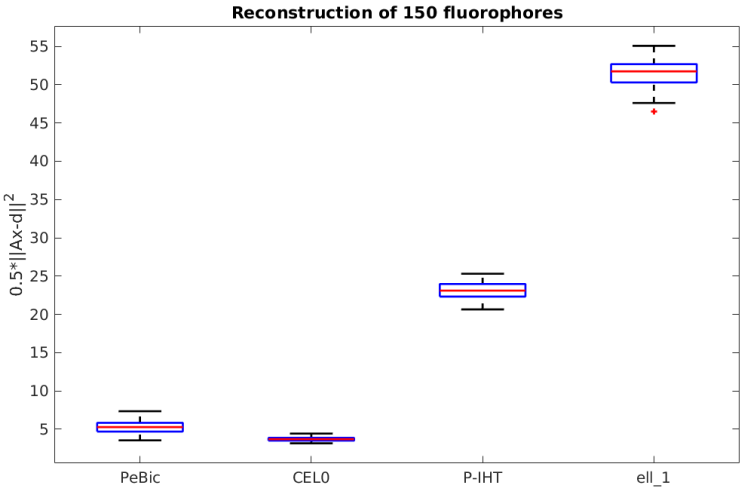


Figure 5.15: Comparison of the penalized-based algorithms: PeBic, CEL0, P-IHT and ℓ_1-relaxation. The y-axis represent the value $\frac{1}{2}\|Ax - d\|^2$. The lower, the better.

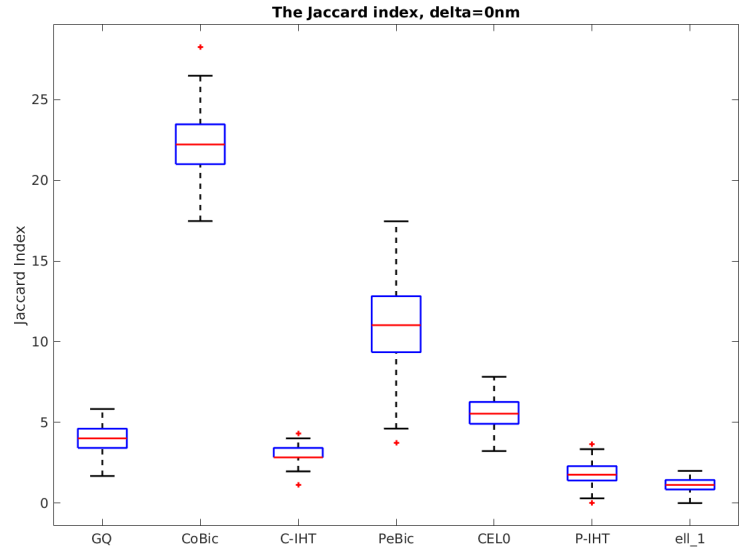


Figure 5.16: Comparison of the algorithms. The y-axis represent Jaccard index with a margin of error of onm. The higher, the better.

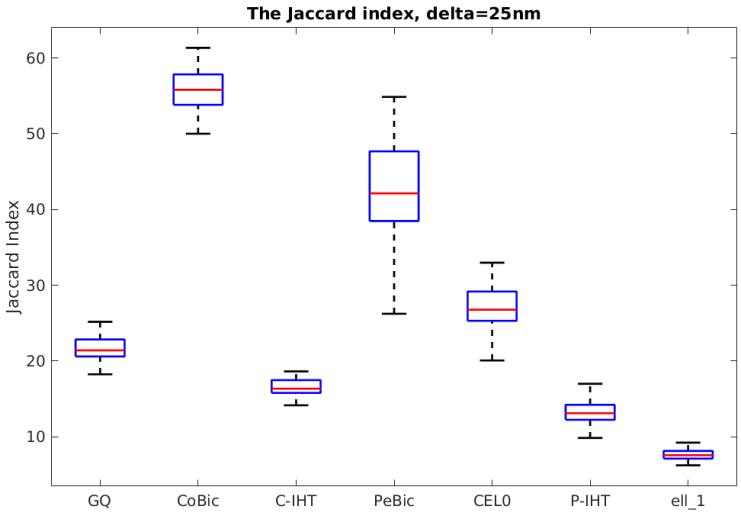


Figure 5.17: Comparison of the algorithms. The y-axis represent Jaccard index with a margin of error of 25nm. The higher, the better.

would prefer to minimize the cost function. As for reconstructing fluorophores, having one correct position is better than reconstructing a fluorophore in the wrong position.

5.6 ISBI SIMULATED DATA

We compare CoBic, PeBic, and G_Q with the same four algorithms in the previous section; the CEL0, ℓ_1-relaxation, and C-IHT and P-IHT. We also compare them to a deep-learning algorithm, Deep-STORM [Neh+18], and we use the public codes of Deep-STORM [She18]. The Deep-STORM is a deep learning algorithm. To teach the algorithm, the user needs to create proper simulated images that represent the dataset. This requires the knowledge of the density, the PSF, as well as an estimation of the noise level.

5.6.1 *The observations*

The first dataset contains simulated acquisitions, making it possible to evaluate the reconstruction quantitatively. The ISBI simulated dataset represents eight tubes of 30 nm diameter. The acquisitions are captured with a 64×64- pixels sensor where each pixel is of size 100×100 nm^2. The Point Spread Function (PSF) is modeled by a Gaussian function where the Full Width at Half Maximum (FWHM) is 258.21 nm. In total, there are 81 049 fluorophores on a total of 361 images. Figure 5.19 (a), (b) and (c) show three of the 361 acquisitions of the simulated dataset, and we apply the localization algorithms

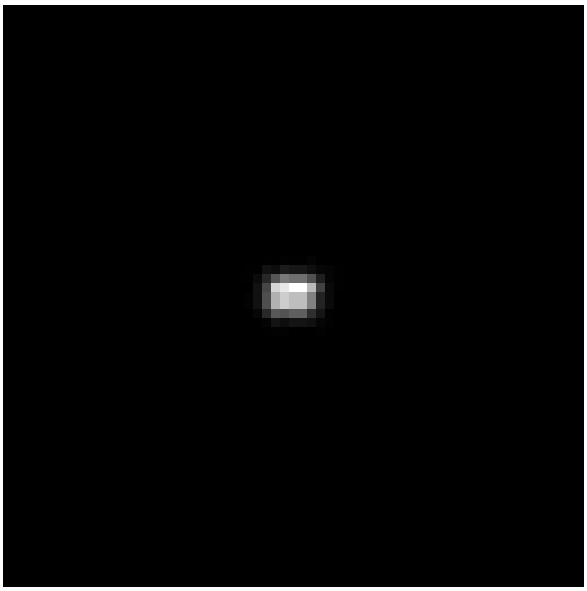


Figure 5.18: A simulated observation where two fluorophores are placed on a distance of 80nm from each other.

to each acquisition. We add the 361 localization results together to obtain one super-resolved image.

Figure 5.19 (d) shows the ground truth on an image of 265 times 265-pixels. Figure 5.19 (e) represents the sum of all the acquisitions, which gives an idea of the image resolution if conventional fluorescence microscopy was used. Highlighted in red and green are cases that show the limit of traditional microscopy. Since the fluorophores are too close in these parts, it is impossible to distinguish the tubes from each other.

5.6.2 *The choice of parameters*

We localize the fluorophores on a fine grid of 256×256 pixel image, where the size of each pixel is $25nm \times 25nm$. Mathematically, this is equivalent to reconstruct $x \in \mathbb{R}^{ML \times ML}$ from an acquisition $d \in \mathbb{R}^{M \times M}$, where $M = 64$ and $L = 4$. Note that L could be larger, but this introduces a greater number of local minima, and the results might be worse. See Table 6 in Appendix C for a small comparison using CoBic. The center of the pixel is used to estimate the precise position of the fluorophore.

Before showing the results of the reconstruction, a discussion is needed. As shown in the previous section, the algorithms do not manage to reconstruct perfectly the exact number of fluorophores. There are some false positives. In one image, this may not be too apparent. In the simulated dataset, we have more than 350 reconstructions to perform. The reconstructed images presented are the sum of all the

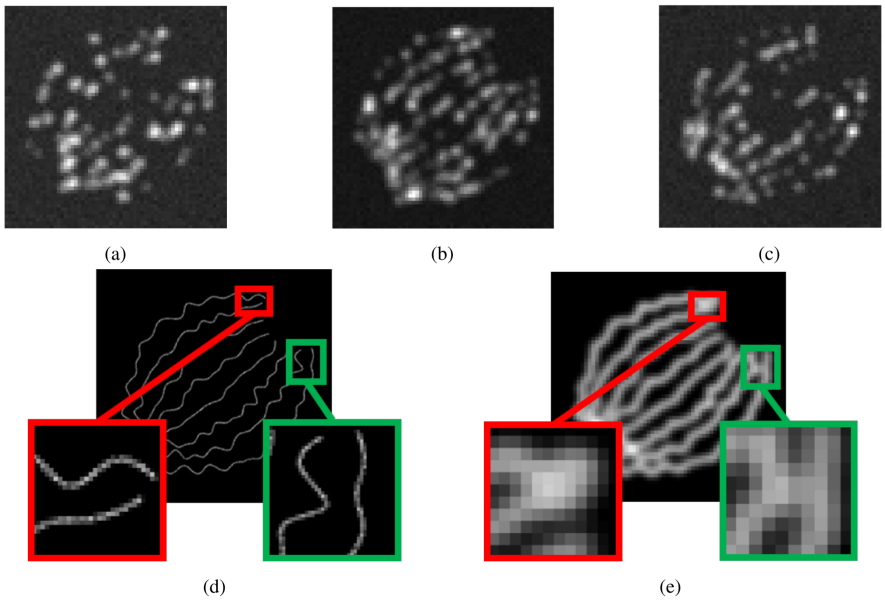


Figure 5.19: Top: (a) 1st, (b) 200th and (c) 361th frame of the simulated high density data. Bottom: (d) Ground truth and (e) the sum of all acquisitions.

reconstructions. Thus the number of False Positives is multiplied by the number of images in the dataset. In the final image, the False Positives looks like noise.

To avoid too many false positives, we could underestimate the number of fluorophores in each image. This increases the number of false negatives in each reconstruction, and hopefully decreases the number of False Positives. Thus the final image will appear with less "noise," as False Negatives are fluorophores that are not reconstructed.

In order to observe the reconstruction, we normalize the image after summing all the reconstruction. Thus the brightest points indicate strong intensity, and dark spots indicate a low intensity.

First, we set k equal to the average number of fluorophores for each acquisition, which is around 220, known from the ground truth. As discussed, we test cases lower than this. After testing each method for different cases, we have decided to compare the reconstruction for further three cases. The algorithms reconstruct 90, 99, and 140 fluorophores on average. 99 corresponds to the best reconstruction for CoBic and PeBic, and 140 corresponds to the best reconstruction for CEL0 and G_Q. Once again, we test 90 to compare the methods when we have a number lower than the best of the other methods.

In order to obtain the best results for CoBic and PeBic, we set $\rho^{(0)} = 1$ for a reconstruction of 89 and 99 fluorophores. For the reconstruction of 140, we set $\rho^{(0)} = 0.1$. We change the initialization as CoBic does not reconstruct more than 99 fluorophores on average, even when choosing k much larger.

For the other methods, we use the best initialization presented in Table 1 and Table 2.

5.6.3 *Results*

The ground truth and the sum of the 361 acquisitions can be observed Fig. 5.19(d) and Fig. 5.19(e).

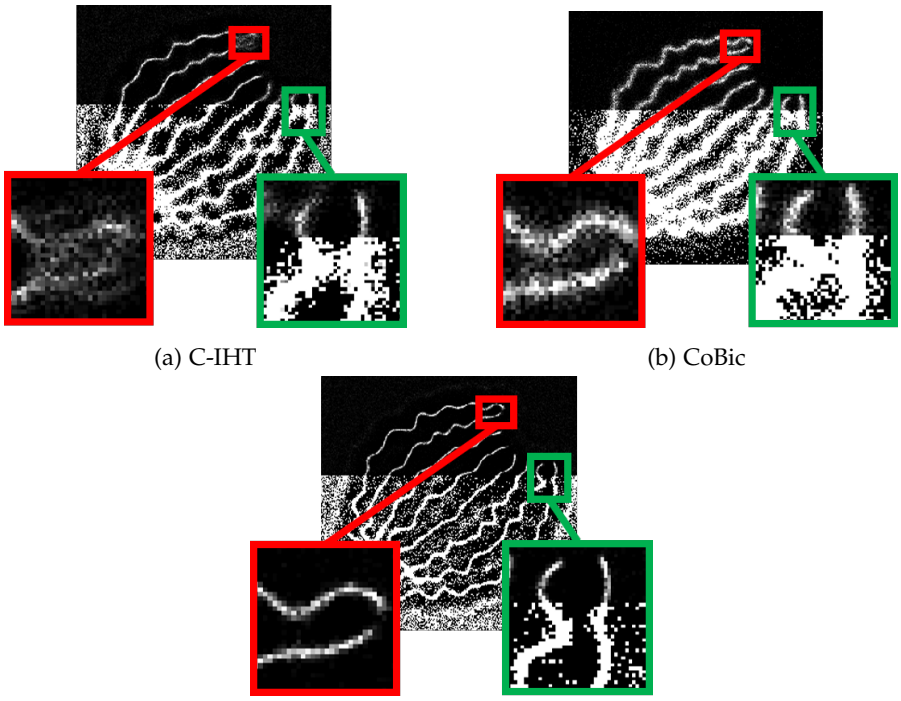
We present first the results when we reconstruct 220 fluorophores on average. Note that we present the results of Deep-STORM with the results for 99 fluorophores, later in the section. Further note that it is not possible to chose the sparsity of the solution of Deep-STORM.

The Jaccard index is presented in Table 3. The reconstructed images can be observed in Fig. 5.20 and Fig. 5.21. In each image, the lower half shows each non-zero pixel as a bright spot.

Table 3: The Jaccard index obtained for an reconstruction of 220 non zero pixels on average.

Jaccard index (%) for 220 non-zero pixels on average			
Method/Tolerance	50nm	100nm	150nm
C-IHT	19.3	37.1	47.0
P-IHT	23.2	49.6	55.9
CEL0	22.4	41.1	49.0
CoBic	19.8	47.1	60.6
PeBic	18.8	41.2	52.2
G_Q	30.7	38.4	39.5
ℓ_1-relaxation	**33.3**	**56.7**	**67.0**

The ℓ_1-relaxation performs best in terms of the Jaccard index for all error tolerances. Apart from G_Q, the other algorithms have a significantly lower index. Once the tolerance increases, the difference between G_Q and ℓ_1-relaxation increases as well, while the difference between P-IHT and ℓ_1-relaxation decreases. The reason can be observed in Fig. 5.20 and Fig. 5.21. ℓ_1-relaxation and P-IHT reconstruct mostly on the tubulins, the CEL0 and G_Q localize fluorophores both on and far from the tubulins. C-IHT, PeBic, and CoBic localize the fluorophores close to the tubulins, making them look thicker than in reality. However, they localize fewer fluorophores far from the ground truth. We remark that while the other methods' Jaccard index grows rapidly as the tolerance increases, G_Q's index stagnates. This means that the fluorophores we observe on the tubulins are reconstructed with high precision, and the rest are reconstructed far from the tubulins. Thus, even when all non-zero pixels are visible, we can easily distinguish the tubulins. This is not the case for the ℓ_1-relaxation, as observed in the green part in Fig. 5.21b. Most of the methods localize fluorophores



(a) C-IHT (b) CoBic

(c) G_Q

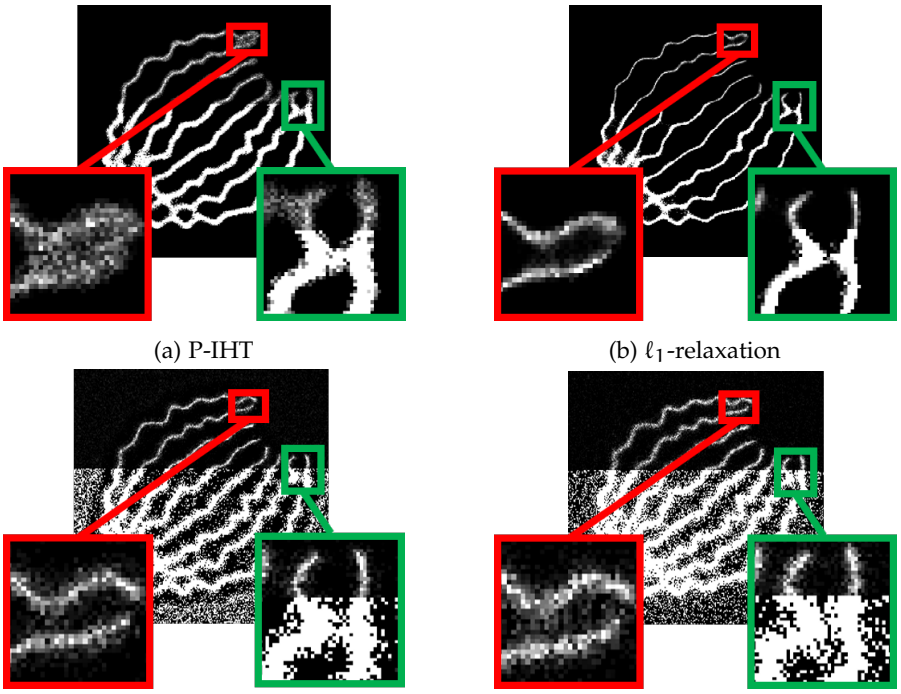
Figure 5.20: Reconstructed images from the simulated ISBI dataset, 220 non-zero pixels on average. Constrained-based algorithms. In the bottom part, each non-zero pixel is white.

far from the tubulins, which must be due to the observations' background noise.

We further remark that both C-IHT and P-IHT struggle to distinguish two tubulins when they are closed. This is highlighted in the red part.

The results when we reconstruct less fluorophores are presented in Fig. 5.22 and Fig. 5.23 with a reconstruction of 99 non-zero pixels on average. C-IHT and P-IHT do not manage to distinguish between two tubes when they are close (see the red case in Fig. 5.22a and Fig. 5.23a) compared to the other algorithms. G_Q and CoBic performs better than C-IHT, and G_Q sligtly better, which can be observed in the highlighted red part in Fig. 5.22. PeBic perform quite well, but not as well as the state-of-the-art CEL0 in penalized $\ell_2 - \ell_0$ minimization. The Deep-STORM algorithm seems to reconstruct the fluorophores best visually.

The Jaccard index is shown in Table 4 for a reconstruction of, on average, 90 non-zero pixels, 99 non-zero pixels, and 140 non-zero pixels. The case of 90 non-zero pixels demonstrates the algorithms' performance with a k chosen, which is not optimal for any algorithm. $k = 140$ is optimal for G_Q and CEL0, and $k = 99$ is optimal for CoBic and PeBic. We observe the low Jaccard index of the IHT constrained algorithm compared to CoBic. G_Q performs better or equivalent to CEL0.



(a) P-IHT (b) ℓ_1-relaxation

(c) CEL0 (d) PeBic

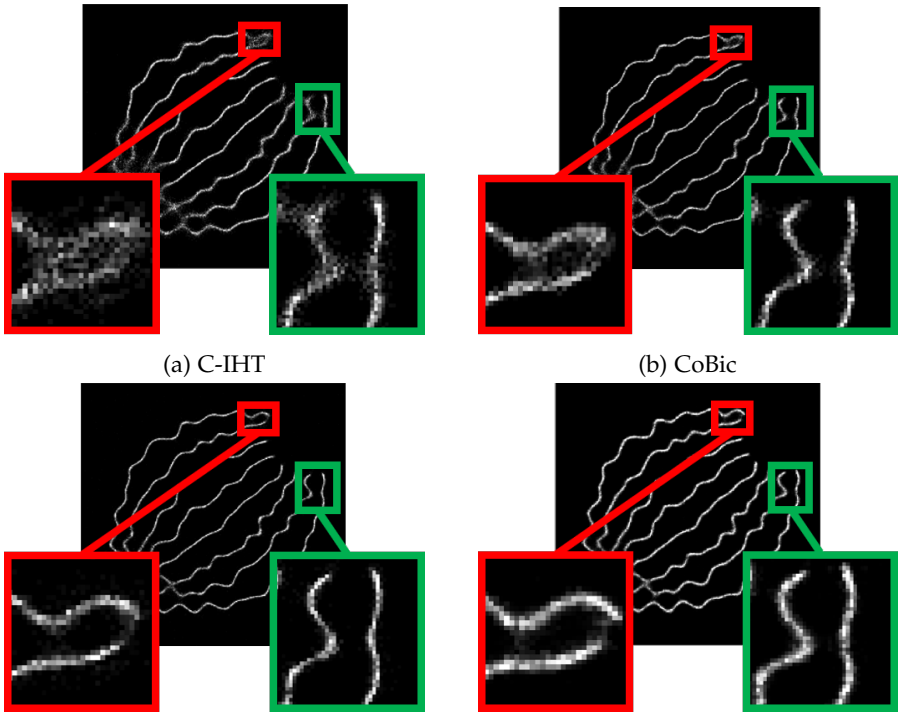
Figure 5.21: Reconstructed images from the simulated ISBI dataset, 220 non-zero pixels on average. Penalized-based algorithms. In the bottom part, each non-zero pixel is white.

Furthermore, G_Q performs better than any of the constrained formulation algorithms (CoBic and C-IHT). The Deep-STORM algorithm reconstruct images with an average of 44264 non-zero pixels. Thus, due to the high number of non-zero pixels, the calculation of the Jaccard index is too demanding, but the index would be close to 0. Most of the non-zero pixels have a low intensity, with higher intensity on the tubulins, which is why we observe in Fig. 5.22d a good reconstruction. We could fix a threshold and let all the pixels with an intensity less than the threshold be zero. However, this would not be fair to the other methods as the same operation could be performed on them.

5.6.3.1 *Time*

Table 5 shows the average computational time for one image acquisition from the simulated dataset. The Deep-STORM is fast and outperforms the other methods in speed. The other algorithms have not been optimized with respect to speed, and could possibly be accelerated by parallel computing and GPU computing.

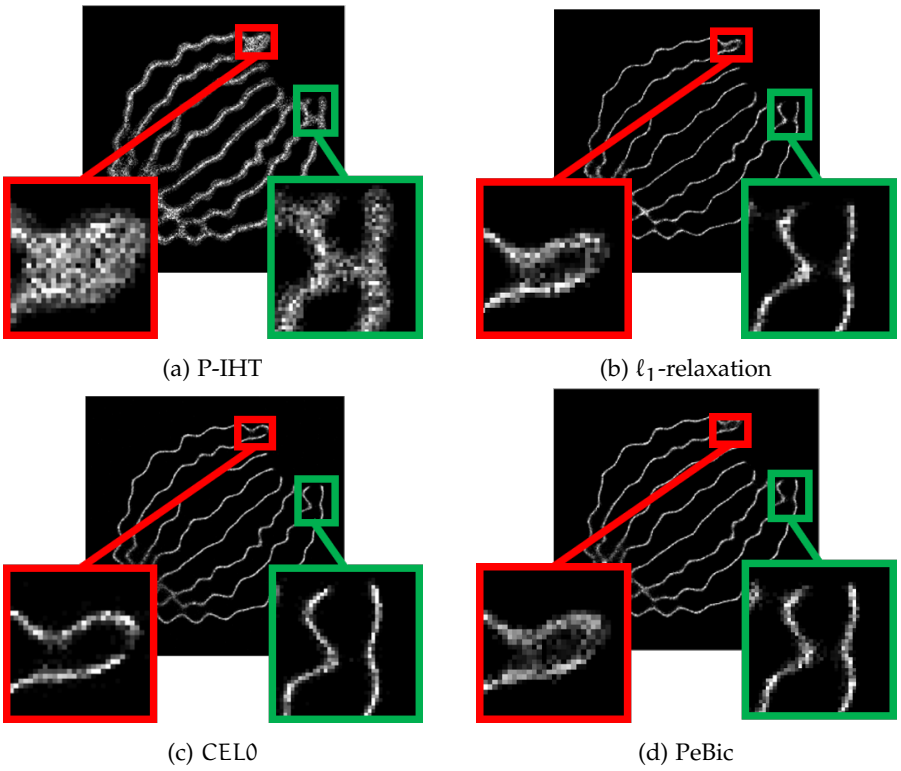
However, the *calibration time*, the time to find the best parameters, is something that cannot be measured by a computer. The advantage of the non-deep-learning methods is that we can fine-tune the parameters by testing them on a few images. This is not possible with Deep-STORM as each change of parameters needs a different training



(a) C-IHT (b) CoBic

(c) G_Q (d) Deep-STORM

Figure 5.22: Reconstructed images from the simulated ISBI dataset, 99 non-zero pixels on average. (a)-(c) are algorithms based on the constrained formulation. (d) Deep-STORM is an deep-learning algorithm.



(a) P-IHT (b) ℓ_1-relaxation

(c) CEL0 (d) PeBic

Figure 5.23: Reconstructed images from the simulated ISBI dataset, 99 non-zero pixels on average. The algorithms are based on the penalized formulation.

Table 4: The Jaccard index obtained for an reconstruction of 90, 99 and 140 non zero pixels on average.

Jaccard index (%) for 90 | 99 | 140 non-zero pixels on average			
Method/Tolerance	50nm	100nm	150nm
C-IHT	20.2 | 21.3 | 22.0	35.0 | 37.8 | 42.2	38.9 | 42.9 | 51.0
P-IHT	10.4 | 13.1 | 12.7	20.9 | 31.2 | 28.1	23.7 | 35.7 | 32.3
CEL0	26.7 | 29.3 | **32.7**	**37.7** | 41.3 | 46.9	38.8 | 42.4 | 49.2
CoBic	23.9 | 25.2 | 21.4	36.3 | 40.0 | 47.0	38.2 | 43.2 | **57.4**
PeBic	23.3 | 25.0 | 16.3	35.0 | 39.3 | 34.2	36.9 | 42.2 | 46.4
G_Q	**27.3** | **29.5** | 32.5	37.4 | **41.9** | 42.5	**39.5** | **43.5** | 44.0
ℓ_1-relaxation	20.1 | 22.4 | 27.5	33.5 | 37.7 | **47.3**	37.5 | 42.4 | 54.1
Deep-STORM	×	×	×

Table 5: Average reconstruction time for one image acquisition.

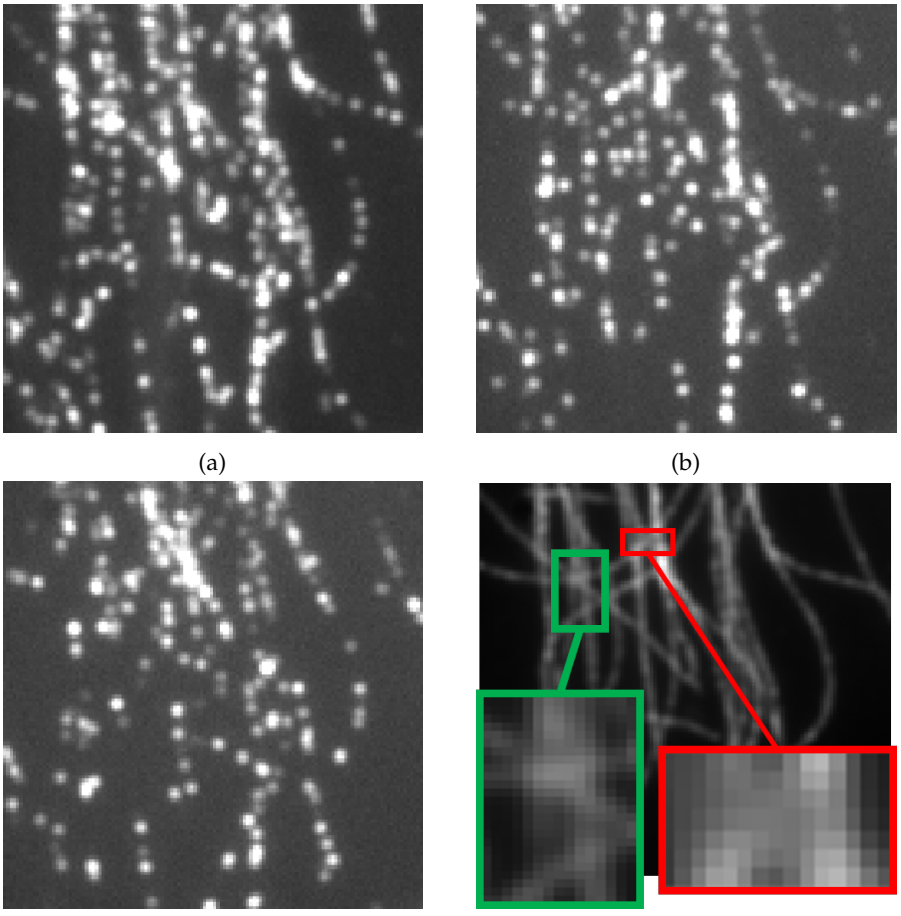
	Average reconstruction time							
Method	C-IHT	P-IHT	CEL0	CoBic	PeBic	G_Q	ℓ_1	D.STORM
Time (s)	67	88	105	87	83	84	49	< <1

set which must be simulated and then the deep neural network must be trained. The total training time is around 2 hours. In contrast to a maximum of 15 minutes if we test the parameters of another algorithm on 7-10 images. Furthermore, the C-IHT, CoBic and G_Q have k as the main parameter. The parameter is quite easy to choose and to adjust from testing. The λ for the penalized formulations is trickier to regulate as it is not possible to know how much to change it to obtain the wished-for result.

5.7 RESULTS OF THE REAL DATASET

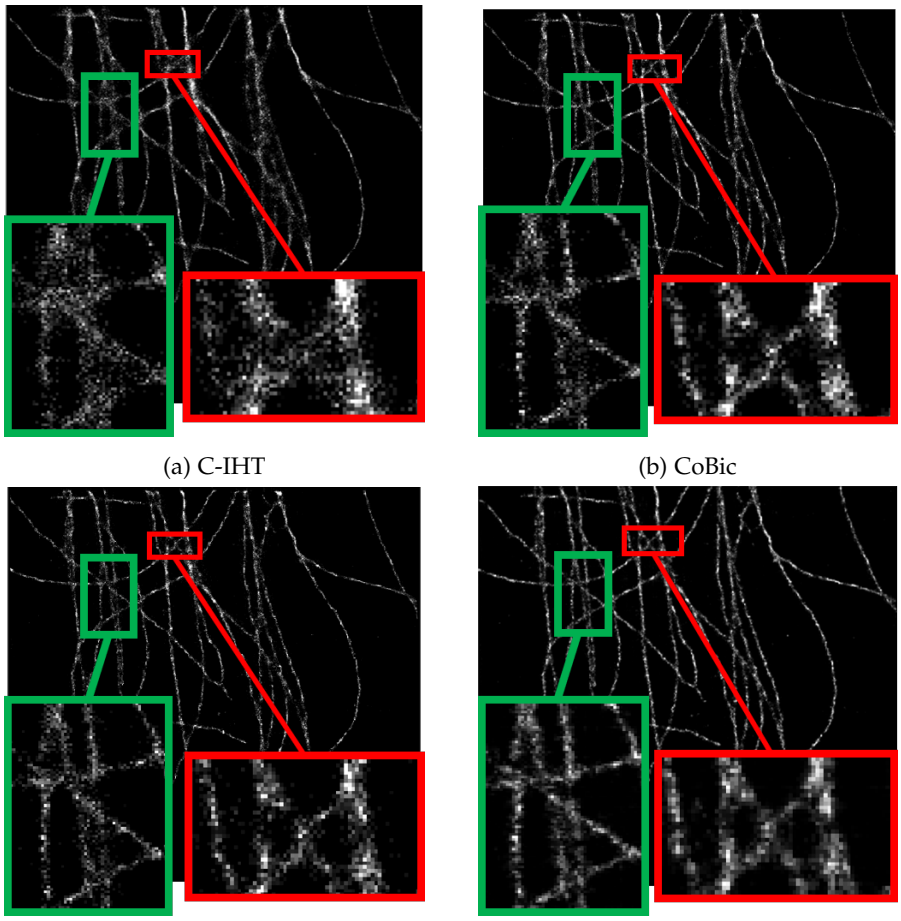
We compare the algorithms on a high-density dataset of tubulins provided from the 2013 ISBI SMLM challenge. The dataset contains 500 acquisitions of 128×128 pixels, and each pixel is of size 100×100 nm^2. The FWHM has been estimated to be 351.8 nm in [Cha14] by averaging many fitted PSF on observed single molecules in the given dataset. We localize the fluorophores on a 512×512 pixel grid. Each pixel is of size 25×25 nm^2. Figure 5.24 presents 3 acquisitions and the sum of all acquisitions.

We do not have any beforehand knowledge of the solution. The parameters, k or λ, are optimized for each method, based on two criteria. The solution should show evident structures such as the tubes on the right side of the image while distinguishing close tubes. P-IHT is an example where this trade-off was difficult to choose. For CoBic



(a) (b) (c) (d)

Figure 5.24: (a) 1st, (b) 250th and (c) 500th frame of the real high density data. (d) the sum of all acquisitions.



(a) C-IHT (b) CoBic

(c) G_Q (d) Deep-STORM

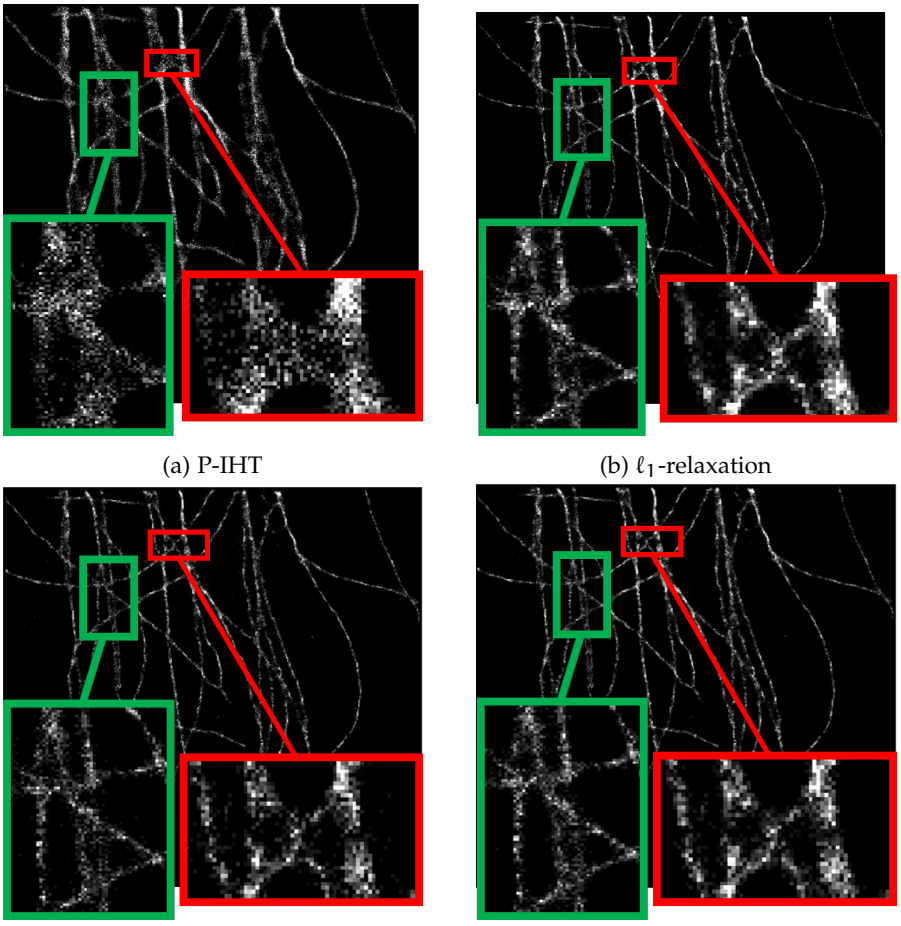
Figure 5.25: Reconstructed images from the real ISBI dataset. (a)-(c) are algorithms based on the constrained formulation. (d) Deep-STORM is an deep-learning algorithm.

and G_Q, we set $k = 140$. Figure 5.25 and Figure 5.26 present the reconstruction. The results are coherent with the obtained results of the simulated dataset. The IHT algorithms reconstruct not as good as the other algorithms, and the penalized version seems worse than the constrained version. The reconstructions obtained by the other algorithms (CoBic, PeBic, CEL0, G_Q and Deep-STORM) are equivalent, with the Deep-STORM algorithm slightly better.

5.8 CONCLUSION

In this chapter, we have applied the proposed methods to Single Molecule Localization Microscopy. The methods are compared to state of the art in 2D grid methods.

We have first compared each method on one simulated image with 100 different noise simulations, where the SNR is around 20 dB. CoBic obtains the lowest data fidelity value and highest Jaccard index when



(a) P-IHT (b) ℓ_1-relaxation

(c) CEL0 (d) PeBic

Figure 5.26: Reconstructed images from the real ISBI dataset. The algorithms are based on the penalized formulation.

reconstructing the exact number of fluorophores in the image. However, when reducing the number of fluorophores, the method is as good as the C-IHT in terms of the data fidelity term. However, in terms of the Jaccard index, CoBic performs better than C-IHT. G_Q is consistent regardless of the number of fluorophores and always outperforms the C-IHT, the function it relaxes. PeBic outperforms P-IHT and the ℓ_1-relaxation.

The proposed methods are applied to simulated ISBI data. Here, the ℓ_1-relaxation obtains a high Jaccard index when reconstructing a high number of fluorophores. However, the resolution is not as good, and we reduce the number. Then, G_Q obtains a higher Jaccard index. CoBic and PeBic obtain good results compared to their initial functions, C-IHT and P-IHT. The reconstruction of the real dataset confirms the excellent localization precision of the proposed methods.

Part III

CONCLUSION

6

CONCLUSION

This thesis has investigated $\ell_2 - \ell_0$ minimization. The main focus has been on the constrained formulation. The constrained $\ell_2 - \ell_0$ may be more intuitive to use than the penalized, as the sparsity constraint may be directly linked to the model. We have seen this in Single-Molecule Localization microscopy, where a non-zero pixel represents a visible fluorophore. In other applications, such as feature selection, the sparsity constraint signify the maximum number of features to use in the model. We have presented two methods to solve the constrained $\ell_2 - \ell_0$ method and one method to solve the penalized $\ell_2 - \ell_0$ method. We have also proposed a multiplicative method, which is still a work-in-progress.

We have applied the methods to SMLM, where we observe the efficient localization abilities of the methods. Furthermore, we can see how much easier it is to use the constrained formulation compared to the penalized formulation.

THE CONTINUOUS RELAXATION OF THE SPARSITY CONSTRAINT : Chapter 2 present a non-convex continuous relaxation of the $\ell_2 - \ell_0$ constrained problem. An explicit form is, to our knowledge, the first of its kind. The relaxation term was obtained by calculating the convex envelope of the initial function G_k with an orthogonal observation matrix. We were inspired by CEL0 relaxation, which has robust results. However, the calculation was significantly more difficult as the sparsity constraint is not separable. Several properties of the proposed relaxation are proven in the chapter. Most importantly, all minimizers of the relaxed function that satisfies the sparsity constraint are minimizers of the initial function. However, we can not guarantee that all minimizers satisfy the sparsity constraint. Numerically, to avoid this, we introduce a fail-safe method. Nevertheless, in the application of Single-Molecule Localization Microscopy, the fail-safe is never activated. Thus the algorithm converges always towards a minimizer of the initial problem.

Furthermore, we present an algorithm to minimize the relaxed function, using the proposed regularizer's proximal operator. We observe that this is a relaxation of the hard threshold.

A further study of the relaxed function and its subgradient would be interesting. We want to give certain conditions on the observation matrix, A, and the observation, d, such that the relaxation is exact. However, the direct study of the subgradient in two dimensions proved to be cumbersome. Furthermore, if we converge numerically

towards a point that does not satisfy the constraint, what can we say about this point? Small numerical examples have given few answers, except that it is not sufficient to project directly to the nearest space that satisfies the constraint to obtain the initial function's global minimum.

In two dimensions, we have also observed that we obtain a line of global minimizers if A and d are chosen in a specific way. If we converge towards such a point in N-dimensions, what kind of tools can we put in place to "follow" the line and obtain a global minimizer that satisfies the constraint?

In the numerical part, we have seen that using the penalization of negative numbers sufficed to reconstruct only non-negative elements. This may not be the case in all applications. We want to calculate the proximal operator of the regularization and the positivity constraint.

COBIC AND PEBIC: Chapter 3 presented CoBic and PeBic, two methods to minimize the constrained and penalized $\ell_2 - \ell_0$ problem. Based on a pre-print, we were inspired to adapt their method using an ℓ_2 data fidelity term. The reformulation of the ℓ_0-norm yields a bi-convex cost function. This is interesting as we can apply alternating minimization schemes. Each step of the proposed algorithm can be written as a known minimization problem. Thus efficient and already existing algorithms can be applied.

In chapter 5, we compare directly CoBic and PeBic with their initial functions. Both methods obtain significantly better results in terms of data fidelity and in terms of the Jaccard index than C-IHT and P-IHT when we know the exact number to reconstruct, as seen in Section 5.5.1. However, CoBic is not better than the C-IHT if we reconstruct less. PeBic performs better than the P-IHT in terms of the ℓ_2 data fidelity term.

There are several ideas to explore concerning the reformulation. Both PeBic and CoBic have been proven to be exact when we have a positivity constraint. It would be interesting to prove this without the positivity constraint. This should not be a too big challenge.

Furthermore, CoBic converges sometimes to points that do not saturate the sparsity constraint. Is this a saddle point or a local minimum, in which we cannot add a positive element? If this is a saddle point, how can we "restart" the algorithm in an intelligent way to converge towards a minimum?

From the numerical results, we can clearly see that reformulating the ℓ_0-norm is preferable compared to using the ℓ_0-norm. It would be interesting if we could introduce the reformulation into other data fidelity terms, such as the Poisson data fidelity term, and see if we could prove the exactness of the method, especially when SMLM is mostly corrupted by Poisson noise.

MULTIPLICATION: Chapter 4 introduces a multiplicative formulation for sparse optimization. We have seen how this can be translated as $\ell_2 - \mathrm{R}$ minimization using an adaptive regularization parameter. This is an ongoing field of research, and at the moment, there are many perspectives. First, we have not applied the method to SMLM as the minimization is very long. This is because of the step size that goes towards infinity very fast. Thus, a further study of the algorithms adapted to minimize the cost function is essential. Another aspect of further research is the choice of regularization parameters. Is the proposed function ϕ_ϵ the best, or could we find a better one? There are many exciting research paths to follow on this subject.

Part IV

APPENDIX

APPENDIX A

A.1 PRELIMINARY RESULTS FOR LEMMA 2.4

Proposition (Reminder) $\forall x \in \mathbb{R}^N, \exists T_k(x) \in \{1, 2, \dots, k-1, k\}$ such that

$$|x^{\downarrow}_{k-T_k(x)+1}| \leqslant \frac{1}{T_k(x)} \sum_{i=k-T_k(x)+1}^{N} |x^{\downarrow}_i| \leqslant |x^{\downarrow}_{k-T_k(x)}|. \tag{A.1}$$

Proof. First, we suppose that (A.1) is not true for $j \in \{1, 2, \dots, k-1\}$, i.e., either

$$|x^{\downarrow}_{k-j+1}| > \frac{1}{j} \sum_{i=k-j+1}^{N} |x^{\downarrow}_i|, \tag{A.2}$$

or

$$\frac{1}{j} \sum_{i=k-j+1}^{N} |x^{\downarrow}_i| > |x^{\downarrow}_{k-j}|, \tag{A.3}$$

or both. We prove by recurrence that if (A.1) is not true $\forall j \in \{1, 2, \dots, k-1\}$, then (A.2) is false, and (A.3) is true. We investigate the case $j = 1$:

$$\sum_{i=k}^{N} |x^{\downarrow}_i| = |x^{\downarrow}_k| + \sum_{i=k+1}^{N} |x^{\downarrow}_i| \geqslant |x^{\downarrow}_k|. \tag{A.4}$$

The above inequality is obvious, and we can conclude that for $j = 1$, (A.2) is false, and thus (A.3) must be true, i.e.,

$$\sum_{i=k}^{N} |x^{\downarrow}_i| > |x^{\downarrow}_{k-1}|. \tag{A.5}$$

We suppose that for some $j \in \{1, 2, \dots, k-1\}$, (A.2) is false and (A.3) is true, and we investigate $j+1$.

$$\begin{aligned} \frac{1}{j+1} \sum_{i=k-j}^{N} |x^{\downarrow}_i| &= \frac{1}{j+1} \left(|x^{\downarrow}_{k-j}| + \frac{j}{j} \sum_{i=k-j+1}^{N} |x^{\downarrow}_i| \right) \\ &> \frac{1}{j+1} \left(|x^{\downarrow}_{k-j}| + j|x^{\downarrow}_{k-j}| \right) = |x^{\downarrow}_{k-j+1}|. \end{aligned} \tag{A.6}$$

We get (A.6) since we have supposed (A.3) is true for j. Thus, by recurrence, we can conclude that (A.2) is false, and (A.3) is true $\forall j \in \{1, 2, \dots, k-1\}$.

Now, we investigate $j = k$:

$$\frac{1}{k}\sum_{i=1}^{N}|x_i^{\downarrow}| = \frac{1}{k}\left(|x_1^{\downarrow}| + \frac{k-1}{k-1}\sum_{i=2}^{N}|x_i^{\downarrow}|\right)$$
$$> \frac{1}{k}\left(|x_1^{\downarrow}| + (k-1)|x_1^{\downarrow}|\right) = |x_1^{\downarrow}|. \tag{A.7}$$

We use the fact that (A.3) is true for $j = k-1$ to obtain the above inequality. Thus (A.2) is false. By definition $x_0^{\downarrow} = +\infty$, and thus (A.3) is also false. Thus $T_k(x) = k$ verifies the double inequality in (A.1).

To conclude: Either $T_k(x) = k$, or there exists $j \in \{1, 2, \dots, k-1\}$ such that $T_k(x) = j$. □

Definition A.1. *Let* $P^{(x)} \in \mathbb{R}^{N\times N}$ *the permutation matrix such that* $P^{(x)}x = x^{\downarrow}$. *The space* $\mathcal{D}(x)$ *is defined as:*

$$\mathcal{D}(x) = \{b; P^{(x)}b = b^{\downarrow}\}.$$

$z \in \mathcal{D}(x)$ *means* $< z, x >=< z^{\downarrow}, x^{\downarrow} >$.

Remark A.1. $\mathcal{D}(x) = \mathcal{D}(|x|)$, *since we have* $|x^{\downarrow}| = |x|^{\downarrow}$.

Proposition A.1. *Let* $(a, b) \in \mathbb{R}^N_{\geqslant 0} \times \mathbb{R}^N_{\geqslant 0}$. *Then*

$$\sum_i a_i b_i \leqslant \sum_i a_i^{\downarrow} b_i^{\downarrow}$$

and the inequality is strict if $b \notin \mathcal{D}(a)$.

Proof. [Sim05, Lemma 1.8] proves it without proving the strict inequality.

We assume that a is not on the form $a = t(1, 1 \dots, 1)^T$, i.e., there exists $i \neq j$, $a_i \neq a_j$. Moreover, for simplicity, without loss of generality, we suppose $a = a^{\downarrow}$. We write

$$\sum_i^N a_i b_i = a_N \sum_{i=1}^N b_i + (a_{N-1} - a_N)\sum_{i=1}^{N-1} b_i + \cdots + (a_1 - a_2)b_1.$$

As it is obvious that $\forall j = 1, \dots N$

$$\sum_{i=1}^{j} b_i \leqslant \sum_{i=1}^{j} b_i^{\downarrow}, \tag{A.8}$$

and since $a_{j-1} - a_j \geqslant 0 \,\forall j$, we get

$$\sum_{i=1}^{N} a_i b_i \leqslant \sum_{i=1}^{N} a_i b_i^{\downarrow} = \sum_{i=1}^{N} a_i^{\downarrow} b_i^{\downarrow} \tag{A.9}$$

The goal of Proposition A.1 is to show that the inequality in (A.9) is strict if $b \notin \mathcal{D}(a)$.

First, we can remark if $b \notin \mathcal{D}(a)$, then there exists $j_0 \in \{2, 3, \dots, N\}$ such

$$\sum_{i=1}^{j_0-1} b_i < \sum_{i=1}^{j_0-1} b_i^{\downarrow}. \tag{A.10}$$

By contradiction, if (A.10) is not true, we have $\forall j \in \{2, 3, \dots, N\}$

$$\sum_{1=1}^{j-1} b_i^{\downarrow} \leqslant \sum_{1=1}^{j-1} b_i,$$

and with (A.8), we get

$$\sum_{1=1}^{j-1} b_i^{\downarrow} = \sum_{1=1}^{j-1} b_i. \tag{A.11}$$

From (A.11), we easily obtain $\forall j$,

$$b_j = b_j^{\downarrow},$$

which means $b^{\downarrow} = b$, i.e., $b \in \mathcal{D}(a)$, which contradicts the hypothesis $b \notin \mathcal{D}(a)$. So there exists j_0 such that (A.10) is true, and if $a_{j_0-1} \neq a_{j_0}$

$$(a_{j_0-1} - a_{j_0}) \sum_{i=1}^{j_0-1} b_i < (a_{j_0-1} - a_{j_0}) \sum_{i=1}^{j_0-1} b_i^{\downarrow},$$

which, with (A.8), implies

$$\sum_{i=1}^{N} a_i b_i < \sum_{i=1}^{N} a_i b_i^{\downarrow}.$$

It remains to examine the case where $a_{j_0-1} = a_{j_0}$. In this case we claim there exists $j_1 \in \{1, \dots, j_{0-2}\}$ or $\{j_0, \dots, N\}$ such that

$$\sum_{i=1}^{j_1} b_i < \sum_{i=1}^{j_1} b_i^{\downarrow} \text{ or } \sum_{i=j_0}^{j_1} b_i < \sum_{i=j_0}^{j_1} b_i^{\downarrow}. \tag{A.12}$$

If not, with the same proof as before we get

$$b_i^{\downarrow} = b_i \quad i \in \{1, \dots, j_0 - 2\} \cup \{j_0 + 1, \dots, N\},$$

i.e., we have

$$\begin{pmatrix} b_1^{\downarrow} \\ b_2^{\downarrow} \\ \vdots \\ b_{j_0-2}^{\downarrow} \\ x_1^{\downarrow} \\ x_2^{\downarrow} \\ b_{j_0+1}^{\downarrow} \\ \vdots \\ b_N^{\downarrow} \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_{j_0-2} \\ x_1 \\ x_2 \\ b_{j_0+1} \\ \vdots \\ b_N \end{pmatrix}$$

where $(x_1, x_2) = (b_{j_0-1}, b_{j_0})$ or (b_{j_0}, b_{j_0-1}). The order does not matter since $a_{j_0-1} = a_{j_0}$. This implies that $b \in \mathcal{D}(a)$, which contradicts the hypothesis. So (A.12) is true and we get for example

$$(a_{j_1-1} - a_{j_1}) \sum_{i=1}^{j_1-1} b_i < (a_{j_1-1} - a_{j_1}) \sum_{i=1}^{j_1-1} b_i^{\downarrow},$$

and if $a_{j_1-1} - a_{j_1} \neq 0$ we deduce

$$\sum_i a_i b_i < \sum_i a_i b_i^{\downarrow}. \tag{A.13}$$

If $a_{j_1-1} = a_{j_1}$ we repeat the same argument and proof as above, and we are sure to find an index j_w such that $a_{j_w-1} - a_{j_w} \neq 0$ since we have supposed that $a \neq t(1, 1, \ldots, 1)^T$. Therefore (A.13) is always true which concludes the proof. □

Proposition A.2 ([TTG17]). *$g(x) : \mathbb{R}^N \to \mathbb{R}$ defined as $g(x) = \frac{1}{2}\sum_{i=1}^{k} x_i^{\downarrow 2}$, is convex. Furthermore, note that $g(|x|) = g(x)$.*

Lemma A.3. *Let $f_1(z, x) \in \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}$ be defined as*

$$f_1(z, x) := -\frac{1}{2}\sum_{i=1}^{k} z_i^{\downarrow 2} + < z^{\downarrow}, x^{\downarrow} > .$$

Let us consider the concave problem

$$\sup_{z \in \mathbb{R}^N_{\geqslant 0}} f_1(z, |x|). \tag{A.14}$$

Problem (A.14) has the following optimal arguments

$$\arg\sup_{z \in \mathbb{R}^N_{\geqslant 0}} f_1(z, |x|) = \{z; \exists\, P \in \mathbb{R}^{N \times N} \text{ a permutation matrix s.t. } Pz = \hat{z}\}, \tag{A.15}$$

where $\hat{z}$ is defined as

$$\hat{z}_j = \begin{cases} \frac{1}{T_k(x)} \sum_{i=k-T_k(x)+1}^{N} |x_i^{\downarrow}| & \text{if } k \geqslant j \geqslant k - T_k(x) + 1 \\ & \text{or if } j > k \text{ and } x_j^{\downarrow} \neq 0 \\ \left[0, \frac{1}{T_k(x)} \sum_{i=k-T_k(x)+1}^{N} |x_i^{\downarrow}|\right] & \text{if } j > k \text{ and } x_j^{\downarrow} = 0 \\ |x_j^{\downarrow}| & \text{if } j < k - T_k(x) + 1. \end{cases} \tag{A.16}$$

We can remark that $\hat{z} = \hat{z}^{\downarrow}$, and $T_k(x)$ is defined in Proposition 2.2. The value of the supremum problem is

$$\frac{1}{2}\sum_{i=1}^{k-T_k(x)} x_i^{\downarrow 2} + \frac{1}{2T_k(x)} \left(\sum_{i=k-T_k(x)+1}^{N} |x_i^{\downarrow}| \right)^2. \tag{A.17}$$

Proof. Problem (A.14) can be written as

$$\sup_{z\in\mathbb{R}^N_{\geqslant 0}} \sum_{i=1}^{k} |x_i^{\downarrow}| z_i^{\downarrow} - \frac{1}{2}\sum_{i=1}^{k} z_i^{\downarrow 2} + \sum_{i=k+1}^{N} |x_i^{\downarrow}| z_i^{\downarrow}. \tag{A.18}$$

We remark that finding the supremum for $z_i^{\downarrow}, i > k$ reduces to find the supremum of the following term, knowing that $z_i^{\downarrow}$ is upper bounded by $z_{i-1}^{\downarrow}$:

$$\sum_{i=k+1}^{N} |x_i^{\downarrow}| z_i^{\downarrow}. \tag{A.19}$$

Let $z_k^{\downarrow}$ be a constant. The sum in (A.19) is non-negative and increasing with respect to $z_j^{\downarrow}$ and the supremum is obtained when $z_j^{\downarrow}$ reaches its upper bound, i.e $z_j^{\downarrow} = z_{j-1}^{\downarrow}\, \forall j > k$ and $|x_j^{\downarrow}| \neq 0$. By recursion, $z_j^{\downarrow} = z_k^{\downarrow}\, \forall j > k$ and $|x_j^{\downarrow}| \neq 0$. When $\exists\, j > k, |x_j^{\downarrow}| = 0$, we observe that $z_j^{\downarrow}$ is multiplied with zero, and can take on every value between its lower bound and upper bound, which is between 0 and $z_k^{\downarrow}$. Then, obviously, the supremum arguments for (A.19) is

$$z_i^{\downarrow} \begin{cases} = z_k^{\downarrow} \text{ if } |x_i^{\downarrow}| \neq 0 \\ \in [0, z_k^{\downarrow}] \text{ if } |x_i^{\downarrow}| = 0 \end{cases} \tag{A.20}$$

Further, from (A.18), we observe that for $i < k$, the optimal argument is

$$z_i^{\downarrow} = \max(|x_i^{\downarrow}|, z_{i+1}^{\downarrow}). \tag{A.21}$$

By recursion, we can write this as

$$z_i^{\downarrow} = \max(|x_i^{\downarrow}|, z_k^{\downarrow}). \tag{A.22}$$

It remains to find the value of $z_k^{\downarrow}$.

Inserting (A.20) and (A.22) into (A.18), and we obtain:

$$\sup_{z_k^{\downarrow}} \sum_{i=1}^{k} |x_i^{\downarrow}| \max(|x_i^{\downarrow}|, z_k^{\downarrow}) - \frac{1}{2}\sum_{i=1}^{k} \max(|x_i^{\downarrow}|, z_k^{\downarrow})^2 + \sum_{i=k+1}^{N} |x_i^{\downarrow}| z_k^{\downarrow}. \tag{A.23}$$

To treat the term $\max(|x_i^{\downarrow}|, z_k^{\downarrow})$, we introduce $j^*(k) = \sup_j\{j : z_k^{\downarrow} \leqslant |x_j^{\downarrow}|\}$, i.e., $j^*(k)$ is the largest index such that $|x_{j^*(k)}^{\downarrow}| \geqslant z_k^{\downarrow}$, and we define $x_0^{\downarrow} = +\infty$. Therefore, (A.23) rewrites as

$$\sup_{z_k^{\downarrow}} \sum_{i=1}^{j^*(k)} |x_i^{\downarrow}|^2 - \frac{1}{2}\sum_{i=1}^{j^*(k)} |x_i^{\downarrow}|^2 + \sum_{i=j^*(k)+1}^{k} |x_i^{\downarrow}| z_k^{\downarrow} - \frac{1}{2}\sum_{i=j^*(k)+1}^{k} z_k^{\downarrow 2} + \sum_{i=k+1}^{N} |x_i^{\downarrow}| z_k^{\downarrow}. \tag{A.24}$$

(A.24) is a concave problem, and the optimiality conditions yields

$$-\sum_{i=j^*(k)+1}^{k} z_k^{\downarrow} + \sum_{j^*(k)+1}^{N} |x_i^{\downarrow}| = 0. \tag{A.25}$$

We define $\sum_{i=j^*(k)+1}^{k} 1 = S$. Then $j^*(k) = k - S$ and

$$z_k^{\downarrow} = \frac{1}{S} \sum_{k-S+1}^{N} |x_i^{\downarrow}|. \tag{A.26}$$

Furthermore, since $j^*(k) = k - S$ was the largest index such that $|x_{k-S}| \geqslant z_k^{\downarrow} > |x_{k-S+1}|$. This translates to

$$|x_{k-S}^{\downarrow}| \geqslant \frac{1}{S} \sum_{k-S+1}^{N} |x_i^{\downarrow}| > |x_{k-S+1}^{\downarrow}|,$$

which implies $S = T_k(x)$ (see Proposition 2.2). Note that if $j^*(k) = k$ (which is the same to say $T_k(x) = 1$), then the right part of the above inequality is not strict.

Now, assume $|x_{j^*(k)}^{\downarrow}| = z_k^{\downarrow}$. Then, the max function can both take $z_k^{\downarrow}$ or $|x_{j^*(k)}^{\downarrow}|$. If it is the latter, than the expression above is correct. In the former case $\max(|x_{j^*(k)}^{\downarrow}|, z_k^{\downarrow}) = z_k^{\downarrow}$. We obtain

$$z_k^{\downarrow} = \frac{1}{T_k(x)+1} \sum_{k-T_k(x)}^{N} |x_i^{\downarrow}|. \tag{A.27}$$

Furthermore, we use the fact that $|x_{j^*(k)}^{\downarrow}| = z_k^{\downarrow}$ and $j^*(k) = k - T_k(x)$, and develop (A.27) as,

$$z_k^{\downarrow} = \frac{1}{T_k(x)+1} \left(x_{k-T_k(x)} + \sum_{k-T_k(x)+1}^{N} |x_i^{\downarrow}| \right) \tag{A.28}$$

$$(T_k(x)+1) z_k^{\downarrow} = z_k^{\downarrow} + \sum_{k-T_k(x)+1}^{N} |x_i^{\downarrow}| \tag{A.29}$$

$$T_k(x) z_k^{\downarrow} = \sum_{k-T_k(x)+1}^{N} |x_i^{\downarrow}| \tag{A.30}$$

$$z_k^{\downarrow} = (A.26) \tag{A.31}$$

The unique value of $z_k^{\downarrow}$ is given by (A.26).

□

Lemma A.4. *Let* $x \in \mathbb{R}^N$ *and* $f_2(y, x) \in \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}$, *defined as*

$$f_2(y, x) = -\frac{1}{2} \sum_{i=1}^{k} y_i^{\downarrow 2} + < y, x >$$

The following concave supremum problem

$$\sup_{y \in \mathbb{R}^N} f_2(y, x) \tag{A.32}$$

is equivalent to

$$\sup_{z \in \mathbb{R}^N_{\geqslant 0}} f_2(z, |x|). \tag{A.33}$$

The arguments are such that $\hat{y}_i^{\downarrow} = \text{sign}^*(x_i^{\downarrow \hat{z}})\hat{z}_i^{\downarrow}$.

Proof. Let $\hat{z} \in \mathbb{R}^N_{\geqslant 0}$ be the argument of the supremum in (A.33), $\hat{y}$ be such that $\hat{y}_i = \text{sign}(x_i)\,\hat{z}_i$, and note that $f_2(y, x) = -g(y) + < y, x >$ with g defined as in Proposition A.2 in Appendix A.1. First, $f_2(y, x)$ is a concave function in y (see Proposition A.2). Furthermore, $f_2(y, x)$ is such that $-f_2(y, x)$ is coercive in y. Thus a supremum exists. Further note that $g(\hat{y}) = g(|\hat{y}|) = g(\hat{z})$. Then the following sequence of equalities/inequalities completes the proof:

$$\begin{aligned}
(\text{A.33}) &= \sup_{z \in \mathbb{R}^N_{\geqslant 0}} f_2(z, |x|) = -g(\hat{z}) + \sum_{i=1}^{N} \hat{z}_i |x_i| \\
&= -g(\hat{z}) + \sum_{i=1}^{N} \text{sign}(x_i)\hat{z}_i x_i = -g(\hat{y}) + \sum_{i=1}^{N} \hat{y}_i x_i \\
&\leqslant (\text{A.32}) = \sup_{y \in \mathbb{R}^N} f_2(y, x) \underset{< y, x > \leqslant < |y|, |x| >}{\leqslant} \sup_{y \in \mathbb{R}^N} f_2(|y|, |x|) \\
&= \sup_{z \in \mathbb{R}^N_{\geqslant 0}} f_2(z, |x|) = (\text{A.33})
\end{aligned}$$

□

A.2 PROOF OF LEMMA 2.4

Proof. Note that a similar problem has been studied in [ACO17]. They do however work with low-rank approximation, therefore they did not have the problem of how to permute x since they work with matrices. First, let $\mathcal{D}(x)$ be as defined in Definition A.1.

We are interested in

$$\sup_{y \in \mathbb{R}^N} f_2(y, x),$$

and its arguments, with f_2 defined in Lemma A.4. From this lemma, we know that we can rather study

$$\sup_{z \in \mathbb{R}^N_{\geqslant 0}} f_2(z, |x|).$$

Furthermore, from Lemma A.3, we know the expression of $\sup_{z \in \mathbb{R}^N_{\geqslant 0}} f_1(z, |x|)$ and its arguments. We want to show that

$\sup_{z\in\mathbb{R}^N_{\geqslant 0}} f_2(z,|x|) = \sup_{z\in\mathbb{R}^N_{\geqslant 0}} f_1(z,|x|)$, and to find a connection between the arguments of f_2 and f_1.

First, note that

$$\sup_{z\in\mathbb{R}^N_{\geqslant 0}} f_2(z,|x|) \geqslant \sup_{z\in\mathbb{R}^N_{\geqslant 0}\in\mathcal{D}(x)} f_2(z,|x|). \tag{A.34}$$

From [Simo5, Lemma 1.8] and Proposition A.1, we have that $\forall (y,x) \in \mathbb{R}^N_{\geqslant 0}\times\mathbb{R}^N_{\geqslant 0}$:

$$< y, x > \leqslant < y^{\downarrow}, x^{\downarrow} >,$$

and the inequality is strict if $y \notin \mathcal{D}(x)$, and thus

$$\sup_{z\in\mathbb{R}^N_{\geqslant 0}} f_2(z,|x|) \leqslant \sup_{z\in\mathbb{R}^N_{\geqslant 0}} f_1(z,|x|). \tag{A.35}$$

Note that we have $\mathcal{D}(|x|) = \mathcal{D}(x)$, then $\forall z \in \mathcal{D}(x)$, $f_2(z,|x|) = f_1(z,|x|)$ and:

$$\sup_{z\in\mathbb{R}^N_{\geqslant 0}\in\mathcal{D}(x)} f_2(z,|x|) = \sup_{z\in\mathbb{R}^N_{\geqslant 0}} \sum_{i=1}^{N} z_i^{\downarrow}|x_i^{\downarrow}| - \frac{1}{2}\sum_{i=1}^{k} z_i^{\downarrow 2} = \sup_{z\in\mathbb{R}^N_{\geqslant 0}} f_1(z,|x|). \tag{A.36}$$

Using inequalities (A.34) and (A.35) and connecting them to (A.36), we obtain

$$\sup_{z\in\mathbb{R}^N_{\geqslant 0}} f_1(z,|x|) = \sup_{z\in\mathbb{R}^N_{\geqslant 0}\in\mathcal{D}(x)} f_2(z,|x|) \leqslant \sup_{z\in\mathbb{R}^N_{\geqslant 0}} f_2(z,|x|) \leqslant \sup_{z\in\mathbb{R}^N_{\geqslant 0}} f_1(z,|x|).$$

$f_2(z,|x|)$ is upper and lower bounded by the same value, thus we have

$$\sup_{z\in\mathbb{R}^N_{\geqslant 0}} f_2(z,|x|) = \sup_{z\in\mathbb{R}^N_{\geqslant 0}} f_1(z,|x|) \tag{A.37}$$

The $\sup_{z\in\mathbb{R}^N_{\geqslant 0}} f_1(z,|x|)$ is known from Lemma A.3:

$$\sup_{z\in\mathbb{R}^N_{\geqslant 0}} f_1(z,|x|) = \frac{1}{2}\sum_{i=1}^{k-T_k(x)} x_i^{\downarrow 2} + \frac{1}{2T_k(x)}\left(\sum_{i=k-T_k(x)+1}^{N} |x_i^{\downarrow}|\right)^2 \tag{A.38}$$

with the optimal arguments:

$$\arg\sup_{z\in\mathbb{R}^N_{\geqslant 0}} f_1(z,|x|) = \{z; \exists\, P \in \mathbb{R}^{N\times N} \text{ a permutation matrix s.t. } Pz = \hat{z}\}, \tag{A.39}$$

where $\hat{z}$ is such that:

$$\hat{z}_j = \begin{cases} \frac{1}{T_k(x)}\sum_{i=k-T_k(x)+1}^{N} |x_i^{\downarrow}| & \text{if } k \geqslant j \geqslant k - T_k(x) + 1 \\ & \text{or if } j > k \text{ and } x_j^{\downarrow} \neq 0 \\ \left[0, \frac{1}{T_k(x)}\sum_{i=k-T_k(x)+1}^{N} |x_i^{\downarrow}|\right] & \text{if } j > k \text{ and } |x_j^{\downarrow}| = 0 \\ |x_j^{\downarrow}| & \text{if } j < k - T_k(x) + 1. \end{cases} \tag{A.40}$$

Now we are interested in the optimal arguments of f_2. Let $P^{(x)}$ be such that $P^{(x)}x = x^{\downarrow}$. We define $z^* = P^{(x)^{-1}}\hat{z}$. Evidently, $P^{(x)}z^* = \hat{z}$, and since $\hat{z}$ is sorted by its absolute value, $P^{(x)}z^* = z^{*\downarrow}$, and thus $z^* \in \mathcal{D}(x)$. Furthermore, from Lemma A.3, z^* is a optimal argument of f_1.

We have then $f_2(z^*, |x|) = f_1(z^*, |x|) = \sup_{z \in \mathbb{R}^N_{\geqslant 0}} f_1(z, |x|)$. z^* is therefore an optimal argument of f_2 since (A.37) shows the equality between the supremum value of f_1 and f_2.

We have shown that there exists $\hat{z} \in \arg\sup_{z \in \mathbb{R}^N_{\geqslant 0}} f_1(z, |x|)$, from which we can construct $z^* \in \mathcal{D}(x)$, an optimal argument of f_2. Now, by contradiction, we show that all optimal arguments of f_2 are in $\mathcal{D}(x)$. Assume $\hat{z} = \arg\sup_{z \in \mathbb{R}^N_{\geqslant 0}} f_2(z, |x|)$ and that $\hat{z} \notin \mathcal{D}(x)$. We can construct z^*, such that $z^{*\downarrow} = \hat{z}^{\downarrow}$, and $z^* \in \mathcal{D}(x)$. We have then

$$\begin{aligned} f_2(z^*, |x|) - f_2(\hat{z}, |x|) &= -\frac{1}{2}\sum_i^k z_i^{*\downarrow 2} + < z^*, |x| > \\ &\quad + \frac{1}{2}\sum_i^k \hat{z}_i^{\downarrow 2} - < \hat{z}, |x| > \\ &= < z^*, |x| > - < \hat{z}, |x| > \\ &= < z^{*\downarrow}, |x^{\downarrow}| > - < \hat{z}, |x| > > 0. \end{aligned}$$

The last equality is due to $z^* \in \mathcal{D}(x)$, and the last inequality is from Proposition A.1. Thus $\hat{z}$ is not an optimal argument for f_2, and all optimal arguments of f_2 must be in $\mathcal{D}(x)$.

Furthermore, thus it suffices to study $\sup_{z \in \mathbb{R}^N_{\geqslant 0} \in \mathcal{D}(z)} f_2(z, |x|)$, and from (A.36), we can rather study f_1, and construct all supremum arguments of f_2 from f_1.

$$\arg\sup_{z \in \mathbb{R}^N_{\geqslant 0}} f_2(z, |x|) = P^{(x)^{-1}}\hat{z} \tag{A.41}$$

where $\hat{z}$ is defined in (A.40). □

A.3 CALCULATION OF PROXIMAL OPERATOR OF $\zeta(x)$

As preliminary results, we state and prove the following proposition and lemmas 2.9, A.5 and A.6.

Proposition (reminder)
Let $\gamma > 1$ and $z = \operatorname{prox}_{-(\frac{\gamma-1}{\gamma})\sum_{i=k+1}^N (\cdot)^{\downarrow 2}}(y)$. We have

$$\operatorname{prox}_{\frac{Q}{\gamma}}(y) = \frac{\gamma y - z}{\gamma - 1}. \tag{A.42}$$

Proof. By definition we have

$$\operatorname{prox}_{\frac{Q}{\gamma}}(y) = \arg\min_x Q(x) + \frac{\gamma}{2}\|x - y\|^2$$

Inserting the definition of $Q(x)$ from (2.13), we obtain

$$\begin{aligned}\operatorname{prox}_{\frac{Q}{\gamma}}(y) = \arg\min_x & -\frac{1}{2}\|x\|^2 \\ & + \sup_w < x, w > -\frac{1}{2}\sum_{i=1}^{k} w_i^{\downarrow 2} + \frac{\gamma}{2}\|x-y\|^2\end{aligned}$$

This is equal to

$$\begin{aligned}&\arg\min_x \sup_w < x, w > -\frac{1}{2}\sum_{i=1}^{k} w_i^{\downarrow 2} + \frac{\gamma-1}{2}\|x - \frac{\gamma}{\gamma-1}y\|^2 \\ &= \operatorname{prox}_{(\gamma-1)\sup_w <\cdot,w> - \frac{1}{2}\sum_{i=1}^{k} w^{\downarrow 2}}(\frac{\gamma}{\gamma-1}y).\end{aligned}$$

Using now Moreau's decomposition [BC+11, Theorem 14.3(ii)], we have

$$\operatorname{prox}_{\frac{Q}{\gamma}}(y) = \frac{\gamma}{\gamma-1}y - \frac{1}{\gamma-1}\operatorname{prox}_{\left((\gamma-1)\sup_w <\cdot,w> - \frac{1}{2}\sum_{i=1}^{k} w^{\downarrow 2}\right)^*}(\gamma y).$$

First, note that

$$\left((\gamma-1)\sup_w < \cdot, w > -\frac{1}{2}\sum_{i=1}^{k} w^{\downarrow 2}\right)^*(x) = (\gamma-1)\frac{1}{2}\sum_{i=1}^{k} x_i^{\downarrow 2}$$

since $\frac{1}{2}\sum_{i=1}^{k} x_i^{\downarrow 2}$ is convex (See Proposition A.2), and $f^{**} = f$ if f a convex and lower semi continuous function. Thus

$$\operatorname{prox}_{\frac{Q}{\gamma}}(y) = \frac{\gamma}{\gamma-1}y - \frac{1}{\gamma-1}\operatorname{prox}_{\left((\gamma-1)\frac{1}{2}\sum_{i=1}^{k} \cdot_i^{\downarrow 2}\right)}(\gamma y).$$

Using the definition of the proximal, we aim to rewrite the proximal above.

$$\operatorname{prox}_{\left((\gamma-1)\frac{1}{2}\sum_{i=1}^{k} \cdot_i^{\downarrow 2}\right)}(\gamma y) = \arg\min_x \frac{1}{2}\sum_{i=1}^{k} x_i^{\downarrow 2} + \frac{1}{2(\gamma-1)}\|x-\gamma y\|^2.$$

Further expanding and removing the part when y is not influencing yields that

$$\begin{aligned}&\arg\min_x \frac{1}{2}\sum_{i=1}^{k} x_i^{\downarrow 2} + \frac{1}{2(\gamma-1)}\|x\|^2 - \frac{\gamma}{\gamma-1} < x, y > \\ &= \arg\min_x \frac{1}{2}\sum_{i=1}^{k} x_i^{\downarrow 2} + \frac{1}{2(\gamma-1)}\|x\|^2 + \frac{\gamma}{2(\gamma-1)}\|x\|^2 \\ &\quad - \frac{\gamma}{2(\gamma-1)}\|x\|^2 - \frac{\gamma}{\gamma-1} < x, y > \\ &= \arg\min_x -\frac{1}{2}\sum_{i=k+1}^{N} x_i^{\downarrow 2} + \frac{\gamma}{2(\gamma-1)}\|x-y\|^2 \\ &= \operatorname{prox}_{(\frac{\gamma-1}{\gamma})-\frac{1}{2}\sum_{i=k+1}^{N} \cdot_i^{\downarrow 2}}(y).\end{aligned}$$

We have thus

$$\operatorname{prox}_{\frac{Q}{\gamma}}(y) = \frac{\gamma}{\gamma-1}y - \frac{1}{\gamma-1}\operatorname{prox}_{(\frac{\gamma-1}{\gamma})-\frac{1}{2}\sum_{i=k+1}^{N}\cdot_i^{\downarrow 2}}(y). \tag{A.43}$$

□

Lemma A.5. *Let* $j : \mathbb{R} \to \mathbb{R}$ *be a strictly convex and coercive function, let* $w = \arg\min_t j(t)$*, and let us suppose that* j *is symmetric with respect to its minimum, i.e.* $j(w-t) = j(w+t)\ \forall t \in \mathbb{R}$*. The problem*

$$z = \arg\min_{b \leqslant |t| \leqslant a} j(t)$$

with a *and* b *positive, has the following solution*

$$z = \begin{cases} w & \text{if } b \leqslant |w| \leqslant a \\ \operatorname{sign}^*(w)a & \text{if } |w| \geqslant a \\ \operatorname{sign}^*(w)b & \text{if } |w| \leqslant b. \end{cases}$$

Proof. Since j is symmetric with respect to its minimum $j(w+t_1) \leqslant j(w+t_2)\ \forall |t_1| \leqslant |t_2|$. Assume that $0 < w \leqslant b$. We can write $j(b) = j(w+\alpha)$, $\alpha > 0$ and $j(-b) = j(w+\beta), \beta < 0$. Since $w > 0$ then $|\alpha| < |\beta|$ and thus the minimum is reached with $z = b$ on the interval $[b, a]$. Similar reasoning can be used to prove the other cases. □

Lemma A.6. *Let* $g_i : \mathbb{R} \to \mathbb{R}, i \in [1..N]$ *be strictly convex and coercive. Let* $w = (w_1, w_2, \dots w_N)^T = \arg\min_{t_i} \sum g_i(t_i)$*, i.e.,* $w_i = \arg\min_{t_i} g_i(t_i)$*. Assume that* $|w_1| \geqslant |w_2| \geqslant \cdots \geqslant |w_k|$ *and* $|w_{k+1}| \geqslant |w_{k+2}| \geqslant \cdots \geqslant |w_N|$*. Let* g_i *be symmetric with respect to its minimum. Consider the following problem*

$$\arg\min_{|t_1| \geqslant \cdots \geqslant |t_N|} \sum_{i}^{N} g_i(t_i). \tag{A.44}$$

The optimal solution is

$$t_i(\tau) = \begin{cases} \operatorname{sign}^*(w_i)\max(|w_i|, \tau) & \text{if } 1 \leqslant i \leqslant k \\ \operatorname{sign}^*(w_i)\min(|w_i|, \tau) & \text{if } i > k \end{cases} \tag{A.45}$$

where $\tau \in \mathbb{R}$ *is in* $[\min(|w_k|, |w_{k+1}|), \max(|w_k|, |w_{k+1}|)]$ *and is the value that minimizes* $\sum g_i(t_i(\tau))$*.*

Proof. Note that this proof is inspired by [LO16, Theorem 2], with some modifications. First, if $|w_k| \geqslant |w_{k+1}|$, then w satisfies the constraints in Problem (A.44), and thus w is the optimal solution. If $|w_k| < |w_{k+1}|$ we must search a little more. In both cases we can, since each g_i is convex and symmetric with respect to its minimum,

apply Lemma A.5 for t_i, and the choices can be limited to the following choices:

$$t_i = \begin{cases} w_i & \text{if } |t_{i-1}| \geqslant |w_i| \geqslant |t_{i+1}| \\ \text{sign}^*(w_i)|t_{i+1}| & \text{if } |w_i| < |t_{i+1}| \\ \text{sign}^*(w_i)|t_{i-1}| & \text{if } |w_i| > |t_{i-1}| \end{cases} \tag{A.46}$$

This can be rewritten in a shorter form, at first in the case where $i \leqslant k$.

$$t_i = \text{sign}(w_i)^* \max(|w_i|, |t_{i+1}|). \tag{A.47}$$

This can be proved by recursion. In the case of $i = 1$, w_1 is the optimal argument if $|w_1| \geqslant |t_2|$, otherwise $\text{sign}^*(w_1)|t_2|$ is optimal. Therefore $t_1 = \text{sign}^*(w_1)\max(|w_1|, |t_2|)$. Assume that this is true for the i-th index.

$$t_{i+1} = \begin{cases} w_{i+1} & \text{if } |t_i| \geqslant |w_{i+1}| \geqslant |t_{i+2}| \text{ and } i+1 \leqslant k \\ \text{sign}^*(w_{i+1})|t_{i+2}| & \text{if } |w_{i+1}| < |t_{i+2}| \text{ and } i+1 \leqslant k \\ \text{sign}^*(w_{i+1})|t_i| & \text{if } |w_{i+1}| > |t_i| \text{ and } i+1 \leqslant k. \end{cases} \tag{A.48}$$

But $t_i = \text{sign}^*(w_i)\max(|w_i|, |t_{i+1}|)$, which yields $|t_i| \geqslant |w_i| \geqslant |w_{i+1}|$ and thus the third case of (A.48) can be ignored.

Now assume for an $i \leqslant k$ that $t_i \neq w_i$. This implies that

$$|t_i| = |t_{i+1}| > |w_i|.$$

Since w_i is non increasing for $i \leqslant k$, the following inequality $|t_{i+1}| > |w_{i+1}|$ is true. Furthermore, $|t_{i+1}| = \max(|w_{i+1}|, |t_{i+2}|) = |t_{i+2}|$. By recursion, we have

$$|t_i| = |t_{i+1}| = |t_{i+2}| = \cdots = |t_k|.$$

To facilitate the notations, $|t_k| = \tau$. The lemma is proved by inserting τ instead of $|t_{i+1}|$ and $|t_k|$ into equation (A.47)

When $i > k$, a similar proof of recursion gives:

$$t_i = \text{sign}^*(w_i)\min(|t_k|, |w_i|). \tag{A.49}$$

and by adopting the notation τ, we finish the proof. □

Remark A.2. *Note that if w, defined in Lemma A.6 is such that $|w_k| \geqslant |w_{k+1}|$, then w is solution of (A.44).*

Lemma A.7. *Let* $y \in \mathbb{R}^N$. *Define* $\zeta : \mathbb{R}^N \to \mathbb{R}$ *as* $\zeta(x) := -(\frac{\gamma-1}{\gamma}) \sum_{i=k+1}^{N} (x_i)^{\downarrow 2}$. *The proximal operator of* ζ *is such that*

$$prox_{\zeta(\cdot)}(y)^{\downarrow y} = \begin{cases} \text{sign}(y_i^{\downarrow}) \max(|y_i^{\downarrow}|, \tau) & \text{if } i \leqslant k \\ \text{sign}(y_i^{\downarrow}) \min(\tau, |\gamma y_i^{\downarrow}|) & \text{if } i > k. \end{cases} \tag{A.50}$$

If $|y_k^{\downarrow}| < \gamma |y_{k+1}^{\downarrow}|$ *then* τ *is a value in the interval* $[|y_k^{\downarrow}|, \gamma |y_{k+1}^{\downarrow}|]$, *and is defined as*

$$\tau = \frac{\gamma \sum_{i \in n_1} |y_i^{\downarrow}| + \gamma \sum_{i \in n_2} |y_i^{\downarrow}|}{\gamma \# n_1 + \# n_2} \tag{A.51}$$

where n_1 *and* n_2 *are two groups of indices such that* $\forall i \in n_1, y_i^{\downarrow} < \tau$ *and* $\forall i \in n_2, \tau \leqslant \gamma |y_i^{\downarrow}|$ *for an* $\# n_1$ *and* $\# n_2$ *are the sizes of* $n1$ *and* $n2$. *To go from* $prox_{\zeta(\cdot)}(y)^{\downarrow y}$ *to* $prox_{\zeta(\cdot)}(y)$ *we apply the inverse permutation that sort* y *to* $y^{\downarrow}$.

Note that we search

$$\text{prox}_{-(\frac{\gamma-1}{\gamma}) \sum_{i=k+1}^{N} (\cdot)^{\downarrow 2}}(y) = \arg\min_x -\frac{1}{2} \sum_{i=k+1}^{N} x_i^{\downarrow 2} + \frac{\gamma}{2(\gamma-1)} \|x - y\|_2^2$$

We define two functions, $l_1 : \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}$ and $l_2 : \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}$.

$$l_1(z, a) = \frac{\gamma}{2(\gamma-1)} \sum_i^N (z_i - a_i)^2 - \frac{1}{2} \sum_{i=k+1}^{N} z_i^{\downarrow 2} \tag{A.52}$$

$$l_2(z, a) = \frac{\gamma}{2(\gamma-1)} \sum_i^N (z_i^{\downarrow} - |a_i^{\downarrow}|)^2 - \frac{1}{2} \sum_{i=k+1}^{N} z_i^{\downarrow 2}. \tag{A.53}$$

As in Lemma 2.4, we can create relations between $l_1(z, |a|)$ and $l_2(z, |z|)$, where l_2 can be solved using Lemma A.6.
We omit the proof as it is similar to the one of Lemma 2.4.

A.4 DETAILED ALGORITHM

Algorithm 5 : Nonmonotone APG

Initialization :

$z^{(1)} = x^{(1)} = x^{(0)},\ t^{(1)} = 1, t^{(0)} = 0, \eta \in [0,1), \delta > 0, c^{(1)} = F(x^{(1)}), q^{(1)} = 1, \alpha_x < \frac{1}{L}, \alpha_y < \frac{1}{L}$

Repeat :

$$y^{(p)} = x^{(p)} + \frac{t^{(p-1)}}{t^{(p)}}(z^{(p)} - x^{(p)}) + \frac{t^{(p-1)} - 1}{t^{(p)}}(x^{(p)} - x^{(p-1)})$$

$$z^{(p+1)} = \text{prox}_{\alpha_x g}(y^{(p)} - \alpha_y \nabla f(y^{(p)}))$$

if $F(z^{(p+1)}) \leqslant c^{(p)} - \delta \|z^{(p+1)} - y^{(p)}\|^2$ **then:**

$$x^{(p+1)} = z^{(p+1)}$$

else:

$$v^{(p+1)} = \text{prox}_{\alpha_x g}(x^{(p)} - \alpha_y \nabla f(x^{(p)}))$$

$$x^{(p+1)} = \begin{cases} z^{(p+1)} \text{ if } F(z^{(p+1)}) \leqslant F(v^{(p+1)}) \\ v^{(p+1)} \text{ otherwise} \end{cases}$$

end if.

$$t^{(p+1)} = \frac{\sqrt{4(t^{(p)})^2 + 1} + 1}{2}$$

$$q^{(p+1)} = \eta q^{(p)} + 1$$

$$c^{(p+1)} = \frac{\eta q^{(p)} c^{(p)} + F(x^{(p+1)})}{q^{(p+1)}}$$

Until : Convergence

The convergence of the algorithm is ensured for a non convex function F which satisfies the K-Łcondition, see [LL15].

B

APPENDIX B

B.1 ADDITIONAL PROOFS FOR THE BICONVEX REFORMULATION

Lemma B.1. *Let* $B \in \mathbb{R}^{N \times l}$ *be a semi-orthogonal matrix, that is, a non-square matrix composed of orthonormal columns. Then,* $B^T B$ *is the identity matrix in* $\mathbb{R}^{l \times l}$.

Lemma B.2. *Let* $A \in \mathbb{R}^{P \times N}$, *let* a_i *denote the* i*th column of* A. *Defining* ω *to be a set of indices,* $\omega \subseteq \{1, \ldots, N\}$. *Let the restriction of* A *to the columns indexed by the elements of* ω *be denoted as* $A_\omega = (a_{\omega[1]}, \ldots, a_{\omega[\#\omega]}) \in \mathbb{R}^{P \times \#\omega}$. *Then* $\|A_\omega\| \leqslant \|A\|$.

Proof. A_ω can be written as the product of matrix A and a matrix B. We define the vector $e_i \in \mathbb{R}^P$, the unitary vector which has zeros everywhere except for the i-th place. The matrix $B \in \mathbb{R}^{N \times \#\omega}$ can be constructed with $e_i \; \forall i \in \omega$. The matrix B is thus a semi-orthonormal matrix. The spectral norm of the matrix B is 1, as $B^T B$ is the identity matrix (from Lemma B.1). We overbound A_ω as

$$\|A_\omega\| = \|AB\| \leqslant \|A\|\|B\| = \|A\| \tag{B.1}$$

□

Lemma B.3. *[Pshenichnyi-Rockafellar lemma][Zalo2, Theorem 2.9.1] Assume* g *is a proper lower semi-continuous convex function. Let* C *be a convex set, such that* $\mathrm{int}(C) \cap \mathrm{dom}(g) \neq \emptyset$. *Then*

$$\hat{x} = \arg\min_{x \in C} g(x) \Leftrightarrow \boldsymbol{o} \in \partial g(\hat{x}) + N_C(\hat{x})$$

where N_C *is the normal cone of the convex set* C.

Lemma B.4. *Given the minimization problem*

$$\arg\min_x \frac{1}{2}\|Ax - d\|^2 + < w, |x| > \tag{B.2}$$

where $A \in \mathbb{R}^{P \times N}$ *is a full rank matrix and* w *a non-negative vector.* $|x|$ *is a vector which contains the absolute value of each component of* x. *Let* $\hat{x}$ *be a solution of problem (B.2). Then* $\|A\hat{x} - d\|_2$ *is bounded independently of* w *and*

$$\|A\hat{x} - d\| \leqslant \|d\| \tag{B.3}$$

Proof. Let $\hat{x}$ be the solution of $\arg\min_x \frac{1}{2}\|Ax-d\|^2 + < w, |x| >$, then we have $\forall x \in \mathbb{R}^N$

$$\frac{1}{2}\|A\hat{x}-d\|^2 + < w, |\hat{x}| > \leqslant \frac{1}{2}\|Ax-d\|^2 + < w, |x| > . \tag{B.4}$$

In particular, by choosing $x = 0$ we have:

$$\frac{1}{2}\|A\hat{x}-d\|^2 + < w, |\hat{x}| > \leqslant \frac{1}{2}\|d\|^2. \tag{B.5}$$

Since w is a non-negative vector, the term $< w, |\hat{x}| >$ is always non-negative; therefore we have

$$\frac{1}{2}\|A\hat{x}-d\|^2 \leqslant \frac{1}{2}\|d\|^2$$

and so

$$\|A\hat{x}-d\| \leqslant \|d\|.$$

□

Lemma B.5. *Let* $f(x) = \frac{1}{2}\|Ax-d\|_2^2 + < w, |x| > + \chi_{\cdot \geqslant 0}(x)$, *A be a full rank matrix and w is a non-negative vector. We have the following result: If* $w_i > \sigma(A)\|d\|_2$ *then the optimal solution of the following optimization problem:*

$$\hat{x} = \arg\min_x f(x) \tag{B.6}$$

is achieved with $\hat{x}_i = 0$.

Proof. We start by proving that $\sigma(A)\|d\|_2 \geqslant \left|\left(A^T(A\hat{x}-d)\right)_i\right|$. Remark that Lemma B.4 is valid for problem (B.6), from which we have

$$\begin{aligned}
\sigma(A)\|d\|_2 &\geqslant \sigma(A)\|A\hat{x}-d\|_2 \\
&\geqslant \|A^T\|\|A\hat{x}-d\|_2 \\
&\geqslant \|A^T(A\hat{x}-d)\|_2 \\
&\geqslant \|A^T(A\hat{x}-d)\|_\infty \\
&\geqslant |\left(A^T(A\hat{x}-d)\right)_i| \quad \forall i \in \{1,\dots,N\}
\end{aligned}$$

Then, by choosing, for all $i \in [1..N]$, $w_i > \sigma(A)\|d\|_2$, we are sure that $w_i > \left|\left(A^T(A\hat{x}-d)\right)_i\right|$.

From the Pshenichnyi-Rockafellar lemma, a necessary and sufficient condition for $\hat{x}$ is a minimizer of f on C is that

$$\mathbf{o} \in \partial f(\hat{x}) + N_C(\hat{x})$$

In our case C is the $\mathbb{R}^N_{\geqslant 0}$ and $f(x) = \frac{1}{2}\|Ax-d\|^2 + < w, |x| >$. We have that $\partial f(x) = \partial(\frac{1}{2}\|Ax-d\|^2) + \partial(< w, |x| >)$ as $f(x)$ is a sum of two convex functions, where the intersection of the domains is non empty

(see [HBLC17, Corollary 16.38]).

The optimal condition is therefore

$$\mathbf{o} \in A^T(A\hat{x} - d) + \partial < w, |\hat{x}| > + N_{\mathbb{R}^N_{\geqslant 0}}(\hat{x})$$

where

$$(\partial < w, |\hat{x}| >)_i \begin{cases} = w_i \text{ if } \hat{x}_i > 0 \\ = -w_i \text{ if } \hat{x}_i < 0 \\ \in [-w_i, w_i] \text{ if } \hat{x}_i = 0 \end{cases}$$

and

$$(N_{\mathbb{R}^N_{\geqslant 0}}(\hat{x}))_i \begin{cases} = 0 \text{ if } \hat{x}_i > 0 \\ \in (-\infty, 0] \text{ if } \hat{x}_i = 0 \end{cases}$$

For $\hat{x}_i$ we have the following optimal condition

$$-A^T(A\hat{x} - d)_i \begin{cases} = w_i \text{ if } \hat{x}_i > 0 \\ \in [-w_i, w_i] + \ (-\infty, 0] \text{ if } \hat{x}_i = 0 \end{cases}$$

If $w_i > \sigma(A)\|d\|_2$, then $|A^T(A\hat{x} - d)_i| < w_i$ and $\hat{x}_i$ cannot be strictly positive, furthermore $\hat{x}_i$ cannot be strictly negative since we work in the non-negative space. Therefore $\hat{x}_i = 0$.

□

Lemma B.6. *Let (x_ρ, u_ρ) be a local minimizer of G_ρ defined in (3.9), with I on the constrained form, that is, defined as in (3.6). Let $G_{x_\rho}(u) = \frac{1}{2}\|Ax_\rho - d\|^2 + I(u) + \rho(\|x_\rho\|_1 - < x_\rho, u >)$. We denote O the indexes of the k largest values of $\{i = 1...N, |(x_\rho)_i|\}$. $Q := \{i|(x_\rho)_i > 0\}$, and $S := \{j|(x_\rho)_j < 0\}$. Moreover, define $D := O \cap Q$, $L := O \cap S$ and $W := \{1, 2..., N\}\backslash\{D \cup L\}$. If $\#(D \cup L) = k$, that is, $\|x_\rho\|_0 \geqslant k$, then the minimum of $G_{x_\rho}(u)$ will be reached with u_ρ such that*

$$(u_\rho)_i \begin{cases} = 1 \text{ if } i \in D \\ = -1 \text{ if } i \in L \\ = 0 \text{ if } i \in W \end{cases} \tag{B.7}$$

If $\#(D \cup L) < k$, that is, $\|x_\rho\|_0 < k$, then

$$(u_\rho)_i \begin{cases} = 1 \text{ if } i \in D \\ = -1 \text{ if } i \in L \\ \in [-1, 1] \text{ if } i \in W \end{cases} \tag{B.8}$$

such that $\sum_{i \in W} |u_i| \leqslant k - \#(D \cup L)$.

Proof. The minimization of $G_{x_\rho}(u)$ can be viewed as a problem of minimizing $- < x_\rho, u > + \chi_{-1\leqslant\cdot\leqslant 1}(u) + \chi_{\|\cdot\|_1\leqslant k}(u)$ by using the definition of $I(u)$. The results are obvious. □

Lemma B.7. *Let (x_ρ, u_ρ) be a local minimizer of G_ρ defined in (3.9), with I on the penalized form defined as in (3.7). Let $G_{x_\rho}(u) = \frac{1}{2}\|Ax_\rho - d\|^2 + I(u) + \rho(\|x_\rho\|_1 - < x_\rho, u >)$. The minimum of $G_{x_\rho}(u)$ will be reached with a u_ρ such that*

$$(u_\rho)_i \begin{cases} = 1 \text{ iff } (x_\rho)_i \in [\frac{\lambda}{\rho}, +\infty) \\ = 0 \text{ iff } (x_\rho)_i \in \frac{\lambda}{\rho}[-1, 1] \\ = -1 \text{ iff } (x_\rho)_i \in (-\infty, -\frac{\lambda}{\rho}] \\ \in (0, 1) \text{ iff } (x_\rho)_i = \frac{\lambda}{\rho} \\ \in (-1, 0) \text{ iff } (x_\rho)_i = -\frac{\lambda}{\rho} \end{cases} \tag{B.9}$$

Proof. The proof of the necessary condition:
We start by writing the optimal conditions of $G_{x_\rho}(u)$.

$$\mathbf{0} \in -\rho x_\rho + N_{-1\leqslant\cdot\leqslant 1}(u_\rho) + \begin{cases} \lambda \text{ if } (u_\rho)_i > 0 \\ -\lambda \text{ if } (u_\rho)_i < 0 \\ [-\lambda, \lambda] \text{ if } (u_\rho)_i = 0 \end{cases} \tag{B.10}$$

We split the study of (B.10) in five cases.

- If $(u_\rho)_i = 1$

$$0 \in -\rho(x_\rho)_i + N_{-1\leqslant\cdot\leqslant 1}((u_\rho)_i) + \lambda \Leftrightarrow (x_\rho)_i \in \frac{[0, \infty) + \lambda}{\rho}$$

  Thus, $(u_\rho)_i = 1 \Rightarrow (x_\rho)_i \in [\frac{\lambda}{\rho}, +\infty)$

- If $0 < (u_\rho)_i < 1$

$$0 \in -\rho(x_\rho)_i + N_{-1\leqslant\cdot\leqslant 1}((u_\rho)_i) + \lambda \Leftrightarrow (x_\rho)_i = \frac{\lambda}{\rho}$$

  Thus $0 < (u_\rho)_i < 1 \Rightarrow (x_\rho)_i = \frac{\lambda}{\rho}$

- If $(u_\rho)_i = 0$

$$0 \in -\rho(x_\rho)_i + N_{-1\leqslant\cdot\leqslant 1}((u_\rho)_i) + [-\lambda, \lambda] \Leftrightarrow (x_\rho)_i \in \frac{\lambda}{\rho}[-1, 1]$$

  Thus $(u_\rho)_i = 0 \Rightarrow (x_\rho)_i \in \frac{\lambda}{\rho}[-1, 1]$

- If $-1 < (u_\rho)_i < 0$

$$0 \in -\rho(x_\rho)_i + N_{-1\leqslant\cdot\leqslant 1}((u_\rho)_i) - \lambda \Leftrightarrow (x_\rho)_i = -\frac{\lambda}{\rho}$$

  Thus $-1 < (u_\rho)_i < 0 \Rightarrow (x_\rho)_i = -\lambda\rho$

- If $(u_\rho)_i = -1$

$$0 \in -\rho(x_\rho)_i + N_{-1\leqslant\cdot\leqslant 1}((u_\rho)_i) - \lambda \Leftrightarrow (x_\rho)_i \in \frac{(-\infty, 0] - \lambda}{\rho}$$

Thus, $u_\rho = -1 \Rightarrow (x_\rho)_i \in (-\infty, -\frac{\lambda}{\rho}]$

The proof of sufficient condition:
We can prove the reverse statement. Rewrite $(x_\rho)_i = \frac{\beta}{\rho}$, for some $\beta \in \mathbb{R}$. We have then, from the optimal conditions (B.10) that

$$\mathbf{0} \in -\rho\frac{\beta}{\rho} + N_{-1\leqslant\cdot\leqslant 1}(u_\rho) + \begin{cases} \lambda \text{ if } (u_\rho)_i > 0 \\ -\lambda \text{ if } (u_\rho)_i < 0 \\ [-\lambda, \lambda] \text{ if } (u_\rho)_i = 0 \end{cases} \tag{B.11}$$

$$0 \in [-\beta + \lambda, +\infty) \text{ if } (u_\rho)_i = 1 \tag{B.12}$$
$$0 \in -\beta + \lambda \text{ if } 0 < (u_\rho)_i < 1 \tag{B.13}$$
$$0 \in [-\lambda - \beta, \lambda - \beta] \text{ if } (u_\rho)_i = 0 \tag{B.14}$$
$$0 \in -\beta - \lambda \text{ if } -1 < (u_\rho)_i < 0 \tag{B.15}$$
$$0 \in (-\infty, -(\beta + \lambda)] \text{ if } (u_\rho)_i = -1 \tag{B.16}$$

Assuming $\beta > \lambda$, then only (B.12) is possible. If $\beta = \lambda$, then (B.12), (B.13) (B.14) are possible. If $0 \leqslant \beta < \lambda$, then only (B.14) is possible. If $-\lambda < \beta < 0$, then only (B.14) is possible. If $\beta = -\lambda$, then (B.14), (B.15) and (B.16) are possible. If $\beta < -\lambda$, then only (B.16) is possible.

This finishes the proof.

□

Lemma B.8. *[YG16, Lemma 1] For any* $x \in \mathbb{R}^N$

$$\|x\|_0 = \min_{-\mathbf{1}\leqslant u\leqslant \mathbf{1}} \|u\|_1 \; s.t \; \|x\|_1 =< u, x > \tag{B.17}$$

Proof. We consider first the problem

$$\min_{-\mathbf{1}\leqslant u\leqslant \mathbf{1}} \|u\|_1 \; s.t. \; |x_i| = u_i x_i \quad \forall i \tag{B.18}$$

The equality constraint $|x_i| = u_i x_i$ and $-1 \leqslant u_i \leqslant 1$ yields that $\hat{u}$, solution of (B.18), is

$$\hat{u}_i \begin{cases} = 1 \text{ if } x_i > 0 \\ = -1 \text{ if } x_i < 0 \\ \in [-1, 1] \text{ if } x_i = 0. \end{cases} \tag{B.19}$$

As we minimize $\|u\|_1$, if $x_i = 0$ then $\hat{u}_i = 0$. Thus, we have that $\|\hat{u}\|_1 = \|x\|_0$. Furthermore, since $u \in [-1, 1]$, we have $|x_i| - u_i x_i \geqslant 0 \, \forall i$. So the constraint $|x_i| = x_i u_i \, \forall i$ is equivalent to $\sum_i |x_i| = \sum_i x_i u_i$ which is exactly $\|x\|_1 =< x, u >$. □

Proposition B.9. *The solution* $u^{(s+1)}$ *of*

$$\arg\min_u \lambda\|u\|_1 + \frac{1}{2b^{(s)}}\|u-z\|^2 + \chi_{-1\leqslant\cdot\leqslant 1}(u) \tag{B.20}$$

is reached for

$$(u^{(s+1)})_i = \begin{cases} 1 \textit{ if } z_i \in [1+\lambda b^{(s)}, \infty) \\ z_i - \lambda b^{(s)} \textit{ if } z_i \in (\lambda b^{(s)}, 1+\lambda b^{(s)}) \\ 0 \textit{ if } z_i \in \lambda b^{(s)}[-1,1] \\ z_i + \lambda b^{(s)} \textit{ if } z_i \in (-1-\lambda b^{(s)}, -\lambda b^{(s)}) \\ -1 \textit{ if } z_i \in (-\infty, -1-\lambda b^{(s)}] \end{cases}$$

Proof. A closed form expression can be found by calculating the subgradient for the problem (B.20) with respect to u. The subgradient of the box constraint $\chi_{-1\leqslant\cdot\leqslant 1}$ is 0 if $|u_i| < 1$, $[0,\infty)$ if $u_i = 1$ and $(-\infty, 0]$ if $u_i = -1$. We obtain the following optimal conditions:

$$0 \in \begin{cases} \lambda + [0,\infty) + \frac{1}{b^{(s)}}(u_i^{(s+1)} - z_i) \text{ if } u_i^{(s+1)} = 1 \\ \lambda + \frac{1}{b^{(s)}}(u_i^{(s+1)} - z_i) \text{ if } 1 > u_i^{(s+1)} > 0 \\ \lambda[-1,1] - \frac{1}{b^{(s)}}(z_i) \text{ if } u_i^{(s+1)} = 0 \\ -\lambda + \frac{1}{b^{(s)}}(u_i^{(s+1)} - z_i) \text{ if } -1 < u_i^{(s+1)} < 0 \\ -\lambda + (-\infty, 0] + \frac{1}{b^{(s)}}(u_i^{(s+1)} - z_i) \text{ if } u_i^{(s+1)} = -1 \end{cases}$$

and the optimal solution u_ρ is

$$(u_\rho^{(s+1)})_i = \begin{cases} 1 \text{ if } z_i \in [1+\lambda b^{(s)}, \infty) \\ z_i - \lambda b^{(s)} \text{ if } z_i \in (\lambda b^{(s)}, 1+\lambda b^{(s)}) \\ 0 \text{ if } z_i \in \lambda b^{(s)}[-1,1] \\ z_i + \lambda b^{(s)} \text{ if } z_i \in (-1-\lambda b^{(s)}, -\lambda b^{(s)}) \\ -1 \text{ if } z_i \in (-\infty, -1-\lambda b^{(s)}]. \end{cases}$$

□

APPENDIX C

Table 6: Jaccard index for CoBic with L=8 and L=4 for acquisition 1, 200 and 361, with 99 non-zero pixels. Note that the Jaccard index is higher for L=4 then in the results presented in Table 4 when considering only these samples.

Tolerance	50	100	150	200
Jaccard index L=8	30.2	47.3	51.3	52.7
Jaccard index L=4	33.5	53.2	57.4	58.0

BIBLIOGRAPHY

[Abu+04] Aria Abubakar, Peter Berg, Tarek Habashy, and Henning Braunisch. "A Multiplicative Regularization Approach for Deblurring Problems." In: *Image Processing, IEEE Transactions on* 13 (Dec. 2004), pp. 1524 –1532. DOI: 10.1109/TIP.2004.836172.

[AV94] Robert Acar and Curtis R Vogel. "Analysis of bounded variation penalty methods for ill-posed problems." In: *Inverse problems* 10.6 (1994), p. 1217.

[ACO17] Fredrik Andersson, Marcus Carlsson, and Carl Olsson. "Convex envelopes for fixed rank approximation." In: *Optimization Letters* 11.8 (2017), pp. 1783–1795.

[Asp03] Alexandre d' Aspremont. "A semidefinite representation for some minimum cardinality problems." In: *42nd IEEE International Conference on Decision and Control (IEEE Cat. No. 03CH37475)*. Vol. 5. IEEE. 2003, pp. 4985–4990.

[Att+10] Hédy Attouch, Jérôme Bolte, Patrick Redont, and Antoine Soubeyran. "Proximal alternating minimization and projection methods for nonconvex problems: An approach based on the Kurdyka-Łojasiewicz inequality." In: *Mathematics of Operations Research* 35.2 (2010), pp. 438–457.

[ABS13] Hedy Attouch, Jérôme Bolte, and Benar Fux Svaiter. "Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward–backward splitting, and regularized Gauss–Seidel methods." en. In: *Mathematical Programming* 137.1-2 (Feb. 2013), pp. 91–129. ISSN: 0025-5610, 1436-4646. DOI: 10.1007/s10107-011-0484-9.

[ADS17] M Aucejo and Olivier De Smet. "A multiplicative regularization for force reconstruction." In: *Mechanical Systems and Signal Processing* 85 (2017), pp. 730–745.

[Bab+13] Hazen P Babcock, Jeffrey R Moffitt, Yunlong Cao, and Xiaowei Zhuang. "Fast compressed sensing analysis for super-resolution imaging using L1-homotopy." In: *Optics express* 21.23 (2013), pp. 28583–28596.

[BSZ12] Hazen Babcock, Yaron M Sigal, and Xiaowei Zhuang. "A high-density 3D localization algorithm for stochastic optical reconstruction microscopy." In: *Optical Nanoscopy* 1.1 (2012), p. 6.

[BC+11] Heinz H Bauschke, Patrick L Combettes, et al. *Convex analysis and monotone operator theory in Hilbert spaces*. Vol. 408. Springer, 2011.

[BBFA20a] Arne Bechensteen, Laure Blanc-Féraud, and Gilles Aubert. "A continuous relaxation of the constrained L_2- L_0 problem." In: *To appear in Journal of Mathematical Imaging and Vision* (2020).

[BBFA20b] Arne Bechensteen, Laure Blanc-Féraud, and Gilles Aubert. "New $L_2 - L_0$ algorithm for single-molecule localization microscopy." In: *Biomedical Optics Express* 11.2 (2020), pp. 1153–1174.

[BT09] A. Beck and M. Teboulle. "A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems." In: *SIAM Journal on Imaging Sciences* 2.1 (Jan. 2009), pp. 183–202. DOI: 10.1137/080716542.

[Ber60] Arthur Berry. *A short history of astronomy*. Dover Publications, 1960.

[Bet+06] Eric Betzig, George H. Patterson, Rachid Sougrat, O. Wolf Lindwasser, Scott Olenych, Juan S. Bonifacino, Michael W. Davidson, Jennifer Lippincott-Schwartz, and Harald F. Hess. "Imaging Intracellular Fluorescent Proteins at Nanometer Resolution." en. In: *Science* 313.5793 (Sept. 2006), pp. 1642–1645. ISSN: 0036-8075, 1095-9203. DOI: 10.1126/science.1127344.

[BLP14] Shujun Bi, Xiaolan Liu, and Shaohua Pan. "Exact penalty decomposition method for zero-norm minimization based on MPEC formulation." In: *SIAM Journal on Scientific Computing* 36.4 (2014), A1451–A1477.

[BD08] Thomas Blumensath and Mike E Davies. "Iterative thresholding for sparse approximations." In: *Journal of Fourier analysis and Applications* 14.5-6 (2008), pp. 629–654.

[BST14] Jérôme Bolte, Shoham Sabach, and Marc Teboulle. "Proximal alternating linearized minimization for nonconvex and nonsmooth problems." In: *Mathematical Programming* 146.1 (Aug. 2014), pp. 459–494. ISSN: 1436-4646. DOI: 10.1007/s10107-013-0701-9.

[Bon+17] Silvia Bonettini, Ignace Loris, Federica Porta, Marco Prato, and Simone Rebegoldi. "On the convergence of a linesearch based proximal-gradient method for nonconvex optimization." In: *Inverse Problems* 33.5 (2017), p. 055005.

[Bou+16] Sébastien Bourguignon, Jordan Ninin, Hervé Carfantan, and Marcel Mongeau. "Exact sparse approximation problems via mixed-integer programming: Formulations and computational performance." In: *IEEE Transactions on Signal Processing* 64.6 (2016), pp. 1405–1419.

[Boy+18] Nicholas Boyd, Eric Jonas, Hazen Babcock, and Benjamin Recht. "Deeploco: Fast 3D localization microscopy using neural networks." In: *BioRxiv* (2018), p. 267096.

[BSR17] Nicholas Boyd, Geoffrey Schiebinger, and Benjamin Recht. "The alternating descent conditional gradient method for sparse inverse problems." In: *SIAM Journal on Optimization* 27.2 (2017), pp. 616–639.

[Boy+11] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. "Distributed optimization and statistical learning via the alternating direction method of multipliers." In: *Foundations and Trends in Machine learning* 3.1 (2011), pp. 1–122.

[Bur+18] James V Burke, Frank E Curtis, Adrian S Lewis, Michael L Overton, and Lucas EA Simões. "Gradient sampling methods for nonsmooth optimization." In: *arXiv preprint arXiv:1804.11003* (2018).

[CWB08] Emmanuel J Candes, Michael B Wakin, and Stephen P Boyd. "Enhancing sparsity by reweighted ℓ_1 minimization." In: *Journal of Fourier analysis and applications* 14.5-6 (2008), pp. 877–905.

[Can+08] Emmanuel J Candes et al. "The restricted isometry property and its implications for compressed sensing." In: *Comptes rendus mathematique* 346.9-10 (2008), pp. 589–592.

[Car16] Marcus Carlsson. "On convexification/optimization of functionals including an l2-misfit term." In: *arXiv:1609.09378 [math]* (Sept. 2016). arXiv: 1609.09378.

[Car19] Marcus Carlsson. "On convex envelopes and regularization of non-convex functionals without moving global minima." In: *Journal of Optimization Theory and Applications* 183.1 (2019), pp. 66–84.

[Cha14] Makhlad Chahid. "Echantillonnage compressif appliqué à la microscopie de fluorescence et à la microscopie de super résolution." PhD thesis. Bordeaux, 2014.

[Cha07] Rick Chartrand. "Exact reconstruction of sparse signals via nonconvex minimization." In: *IEEE Signal Processing Letters* 14.10 (2007), pp. 707–710.

[CBL89] S. Chen, S. A. Billings, and W. LUO. "Orthogonal least squares methods and their application to non-linear system identification." In: *International Journal of Control* 50.5 (Nov. 1989), pp. 1873–1896. ISSN: 0020-7179. DOI: 10.1080/00207178908953472.

[CDS01] Scott Shaobing Chen, David L Donoho, and Michael A Saunders. "Atomic decomposition by basis pursuit." In: *SIAM review* 43.1 (2001), pp. 129–159.

[CD94] Shaobing Chen and David Donoho. "Basis pursuit." In: *Proceedings of 1994 28th Asilomar Conference on Signals, Systems and Computers*. Vol. 1. IEEE. 1994, pp. 41–44.

[CXY10] Xiaojun Chen, Fengmin Xu, and Yinyu Ye. "Lower bound theory of nonzero entries in solutions of \ell_2-\ell_p minimization." In: *SIAM Journal on Scientific Computing* 32.5 (2010), pp. 2832–2852.

[Cho+13] Emilie Chouzenoux, Anna Jezierska, Jean-Christophe Pesquet, and Hugues Talbot. "A majorize-minimize subspace approach for $\ell_2 - \ell_0$ image regularization." In: *SIAM Journal on Imaging Sciences* 6.1 (2013), pp. 563–591.

[Cla90] Frank H Clarke. *Optimization and nonsmooth analysis*. Vol. 5. Siam, 1990.

[CDD09] Albert Cohen, Wolfgang Dahmen, and Ronald DeVore. "Compressed sensing and best k-term approximation." In: *Journal of the American mathematical society* 22.1 (2009), pp. 211–231.

[CP11] Patrick L Combettes and Jean-Christophe Pesquet. "Proximal splitting methods in signal processing." In: *Fixed-point algorithms for inverse problems in science and engineering*. Springer, 2011, pp. 185–212.

[CW05] Patrick L Combettes and Valérie R Wajs. "Signal recovery by proximal forward-backward splitting." In: *Multiscale Modeling & Simulation* 4.4 (2005), pp. 1168–1200.

[Com14] Wikimedia Commons. *Airy disk spacing near Rayleigh criterion.png*. 2014. URL: https://en.wikipedia.org/wiki/File:Airy_disk_spacing_near_Rayleigh_criterion.png.

[Dav+12] Mark A Davenport, Marco F Duarte, Yonina C Eldar, and Gitta Kutyniok. *Introduction to compressed sensing*. 2012.

[DE03] David L Donoho and Michael Elad. "Optimally sparse representation in general (nonorthogonal) dictionaries via ℓ_1 minimization." In: *Proceedings of the National Academy of Sciences* 100.5 (2003), pp. 2197–2202.

[EA06] Michael Elad and Michal Aharon. "Image denoising via sparse and redundant representations over learned dictionaries." In: *IEEE Transactions on Image processing* 15.12 (2006), pp. 3736–3745.

[FL01] Jianqing Fan and Runze Li. "Variable selection via nonconcave penalized likelihood and its oracle properties." In: *Journal of the American statistical Association* 96.456 (2001), pp. 1348–1360.

[FBD10] Mário AT Figueiredo and José M Bioucas-Dias. "Restoration of Poissonian images using alternating direction optimization." In: *IEEE transactions on Image Processing* 19.12 (2010), pp. 3133–3145.

[GSBF17] Simon Gazagnes, Emmanuel Soubies, and Laure Blanc-Féraud. "High density molecule localization for super-resolution microscopy using CELo based sparse approximation." In: *Biomedical Imaging (ISBI 2017), 2017 IEEE 14th International Symposium on*. IEEE. 2017, pp. 28–31.

[GTT18] Jun-ya Gotoh, Akiko Takeda, and Katsuya Tono. "DC formulations and algorithms for sparse optimization problems." In: *Mathematical Programming* 169.1 (2018), pp. 141–176.

[Gu+18] Bin Gu, De Wang, Zhouyuan Huo, and Heng Huang. "Inexact proximal gradient methods for non-convex and non-smooth optimization." In: *Thirty-Second AAAI Conference on Artificial Intelligence*. 2018.

[Gus00] Mats GL Gustafsson. "Surpassing the lateral resolution limit by a factor of two using structured illumination microscopy." In: *Journal of microscopy* 198.2 (2000), pp. 82–87.

[Gus+16] Nils Gustafsson, Siân Culley, George Ashdown, Dylan M Owen, Pedro Matos Pereira, and Ricardo Henriques. "Fast live-cell conventional fluorophore nanoscopy with ImageJ through super-resolution radial fluctuations." eng. In: *Nature communications* 7.1 (2016), pp. 12471–12471. ISSN: 2041-1723.

[HBLC17] Heinz H. Bauschke and Patrick L. Combettes. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Jan. 2017.

[Haa10] Alfred Haar. "On the theory of orthogonal function systems." In: *Mathematische Annalen* 69.3 (1910), pp. 331–371.

[Hei20] Hannah Heil. "Sharpening super-resolution by single molecule localization microscopy in front of a tuned mirror." PhD thesis. Jan. 2020.

[Hen+10] Ricardo Henriques, Mickael Lelek, Eugenio F. Fornasiero, Flavia Valtorta, Christophe Zimmer, and Musa M. Mhlanga. "QuickPALM: 3D real-time photoactivation nanoscopy image processing in ImageJ." eng. In: *Nature Methods* 7.5 (May 2010), pp. 339–340. ISSN: 1548-7105. DOI: 10.1038/nmeth0510-339.

[HGM06] Samuel T. Hess, Thanu P. K. Girirajan, and Michael D. Mason. "Ultra-High Resolution Imaging by Fluorescence Photoactivation Localization Microscopy." In: *Biophysical Journal* 91.11 (Dec. 2006), pp. 4258–4272. ISSN: 0006-3495. DOI: 10.1529/biophysj.106.091116.

[Hug+17] S Hugelier, PHC Eilers, O Devos, and Cyril Ruckebusch. "Improved superresolution microscopy imaging by sparse deconvolution with an interframe penalty." In: *Journal of Chemometrics* 31.4 (2017), e2847.

[KH99] Thomas A Klar and Stefan W Hell. "Subdiffraction resolution in far-field fluorescence microscopy." In: *Optics letters* 24.14 (1999), pp. 954–956.

[LO16] Viktor Larsson and Carl Olsson. "Convex Low Rank Approximation." en. In: *International Journal of Computer Vision* 120.2 (Nov. 2016), pp. 194–214. ISSN: 0920-5691, 1573-1405. DOI: 10.1007/s11263-016-0904-7.

[LT+15] Hoai An Le Thi, T Pham Dinh, Hoai Minh Le, and Xuan Thanh Vo. "DC approximation approaches for sparse optimization." In: *European Journal of Operational Research* 244.1 (2015), pp. 26–46.

[LL15] Huan Li and Zhouchen Lin. "Accelerated proximal gradient methods for nonconvex programming." In: *Advances in neural information processing systems*. 2015, pp. 379–387.

[LLL10] Ariel Lipson, Stephen G. Lipson, and Henry Lipson. *Optical Physics*. 4th ed. Cambridge University Press, 2010. DOI: 10.1017/CBO9780511763120.

[LBP18] Yulan Liu, Shujun Bi, and Shaohua Pan. "Equivalent Lipschitz surrogates for zero-norm and rank optimization problems." In: *Journal of Global Optimization* 72.4 (2018), pp. 679–704.

[Luc94] LB Lucy. "Astronomical Inverse Problems." In: *Reviews in modern astronomy*. Vol. 7. 1994, pp. 31–50.

[MZ93] S. G. Mallat and Zhifeng Zhang. "Matching pursuits with time-frequency dictionaries." In: *IEEE Transactions on Signal Processing* 41.12 (Dec. 1993), pp. 3397–3415. ISSN: 1053-587X. DOI: 10.1109/78.258082.

[Mar+18] Elaine Crespo Marques, Nilson Maciel, Lirida Naviner, Hao Cai, and Jun Yang. "A review of sparse recovery algorithms." In: *IEEE Access* 7 (2018), pp. 1300–1322.

[Men13] Jerry M Mendel. *Optimal seismic deconvolution: an estimation-based approach*. Elsevier, 2013.

[Mic16] OpenStax Microbiology. *Microbiology ID: e42bd376-624b-4cof-972f-eoc57998e765@4.4*. 2016. URL: https://courses.lumenlearning.com/biology1/chapter/comparing-prokaryotic-and-eukaryotic-cells/.

[Min+14] Junhong Min, Cédric Vonesch, Hagai Kirshner, Lina Carlini, Nicolas Olivier, Seamus Holden, Suliana Manley, Jong Chul Ye, and Michael Unser. "FALCON: fast and unbiased reconstruction of high-density super-resolution microscopy data." In: *Scientific reports* 4 (2014), p. 4577.

[ML19] AS Mishin and KA Lukyanov. "Live-Cell Super-resolution Fluorescence Microscopy." In: *Biochemistry (Moscow)* 84.1 (2019), pp. 19–31.

[Mor06] Boris S Mordukhovich. *Variational analysis and generalized differentiation I: Basic theory*. Vol. 330. Springer Science & Business Media, 2006.

[MN13] Boris S Mordukhovich and Nguyen Mau Nam. "An easy path to convex analysis and applications." In: *Synthesis Lectures on Mathematics and Statistics* 6.2 (2013), pp. 1–218.

[Neh+18] Elias Nehme, Lucien E Weiss, Tomer Michaeli, and Yoav Shechtman. "Deep-STORM: super-resolution single-molecule microscopy by deep learning." In: *Optica* 5.4 (2018), pp. 458–464.

[Ngu+19] Thanh Thi Nguyen, Jérôme Idier, Charles Soussen, and El-Hadi Djermoune. "Non-negative orthogonal greedy algorithms." In: *IEEE Transactions on Signal Processing* 67.21 (2019), pp. 5643–5658.

[Nik16] Mila Nikolova. "Relationship between the optimal solutions of least squares regularized with ℓ_0-norm and constrained by k-sparsity." In: *Applied and Computational Harmonic Analysis*. Sparse Representations with Applications in Imaging Science, Data Analysis and Beyond 41.1 (July 2016), pp. 237–265. ISSN: 1063-5203. DOI: 10.1016/j.acha.2015.10.010.

[OSK94] Michael S O'Brien, Anthony N Sinclair, and Stuart M Kramer. "Recovery of a sparse spike time series by l/sub 1/norm deconvolution." In: *IEEE Transactions on Signal Processing* 42.12 (1994), pp. 3353–3365.

[Och+15] Peter Ochs, Alexey Dosovitskiy, Thomas Brox, and Thomas Pock. "On iteratively reweighted algorithms for nonsmooth nonconvex optimization in computer vision." In: *SIAM Journal on Imaging Sciences* 8.1 (2015), pp. 331–372.

[Pad+20] Anantachai Padcharoen, Duangkamon Kitkuan, Poom Kumam, Jewaidu Rilwan, and Wiyada Kumam. "Accelerated alternating minimization algorithm for poisson noisy image recovery." In: *Inverse Problems in Science and Engineering* (2020), pp. 1–26.

[PRK93] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad. "Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition." In: *Proceedings of 27th Asilomar Conference on Signals, Systems and Computers*. Nov. 1993, 40–44 vol.1. DOI: 10.1109/ACSSC.1993.342465.

[RW09] R Tyrrell Rockafellar and Roger J-B Wets. *Variational analysis*. Vol. 317. Springer Science & Business Media, 2009.

[RBZ06] Michael J. Rust, Mark Bates, and Xiaowei Zhuang. "Sub-diffraction-limit imaging by stochastic optical reconstruction microscopy (STORM)." en. In: *Nature Methods* 3.10 (Oct. 2006), pp. 793–796. ISSN: 1548-7091. DOI: 10.1038/nmeth929.

[Sag+15] Daniel Sage, Hagai Kirshner, Thomas Pengo, Nico Stuurman, Junhong Min, Suliana Manley, and Michael Unser. "Quantitative evaluation of software packages for single-molecule localization microscopy." In: *Nature methods* 12.8 (2015), p. 717.

[Sag+19] Daniel Sage, Thanh-An Pham, Hazen Babcock, Tomas Lukes, Thomas Pengo, Jerry Chao, Ramraj Velmurugan, Alex Herbert, Anurag Agrawal, Silvia Colabrese, et al. "Super-resolution fight club: assessment of 2D and 3D single-molecule localization microscopy software." In: *Nature methods* 16.5 (2019), pp. 387–395.

[Sch+19] Lothar Schermelleh, Alexia Ferrand, Thomas Huser, Christian Eggeling, Markus Sauer, Oliver Biehlmaier, and Gregor PC Drummen. "Super-resolution microscopy demystified." In: *Nature cell biology* 21.1 (2019), pp. 72–84.

[Sel17] Ivan Selesnick. "Sparse regularization via convex analysis." In: *IEEE Transactions on Signal Processing* 65.17 (2017), pp. 4481–4494.

[SST10] Simon Setzer, Gabriele Steidl, and Tanja Teuber. "Deblurring Poissonian images by split Bregman techniques." In: *Journal of Visual Communication and Image Representation* 21.3 (2010), pp. 193–199.

[She18] Yoav Shechtman. *Software Nano-bio-optics lab*. https://nanobiooptics.net.technion.ac.il/software/. [Online; accessed 2018-10-09]. 2018.

[She+09] Bin Shen, Wei Hu, Yimin Zhang, and Yu-Jin Zhang. "Image inpainting via sparse representation." In: *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE. 2009, pp. 697–700.

[Sim05] Barry Simon. *Trace ideals and their applications*. 120. American Mathematical Soc., 2005.

[SBFA15] Emmanuel Soubies, Laure Blanc-Féraud, and Gilles Aubert. "A Continuous Exact ℓ_0 Penalty (CEL0) for Least Squares Regularized Problem." In: *SIAM Journal on Imaging Sciences* 8.3 (2015), pp. 1607–1639.

[SBFA17] Emmanuel Soubies, Laure Blanc-Féraud, and Gilles Aubert. "A Unified View of Exact Continuous Penalties for $\ell_2 - \ell\_0$ Minimization." In: *SIAM Journal on Optimization* 27.3 (2017), pp. 2034–2060.

[Sou16] Emmnanuel Soubies. "Sur quelques problèmes de reconstruction en imagerie MA-TIRF et en optimisation parcimonieuse par relaxation continue exacte de critères pénalisés en norme-lo." fr. PhD thesis. Université Côte d'Azur, Oct. 2016.

[Sou+11] Charles Soussen, Jérôme Idier, David Brie, and Junbo Duan. "From Bernoulli–Gaussian deconvolution to sparse signal restoration." In: *IEEE Transactions on Signal Processing* 59.10 (2011), pp. 4572–4584.

[Sou+15] Charles Soussen, Jérome Idier, Junbo Duan, and David Brie. "Homotopy Based Algorithms for ℓ_0-Regularized Least-Squares." In: *IEEE Transactions on Signal Processing* 63.13 (2015), pp. 3301–3316.

[STM19] Artur Speiser, Srinivas C Turaga, and Jakob H Macke. "Teaching deep neural networks to localize sources in super-resolution microscopy by combining simulation-based learning and unsupervised learning." In: *arXiv preprint arXiv:1907.00770* (2019).

[Ste04] Stefan M. Stefanov. "Convex quadratic minimization subject to a Linear constraint and box constraints." In: *Applied Mathematics Research eXpress* 2004.1 (2004), pp. 17–42. DOI: 10.1155/S168712000402009X.

[Tak+18] T Takeshima, T Takahashi, J Yamashita, Y Okada, and S Watanabe. "A multi-emitter fitting algorithm for potential live cell super-resolution imaging over a wide range of molecular densities." In: *Journal of microscopy* 271.3 (2018), pp. 266–281.

[Tib96] Robert Tibshirani. "Regression shrinkage and selection via the lasso." In: *Journal of the Royal Statistical Society: Series B (Methodological)* 58.1 (1996), pp. 267–288.

[TTG17] Katsuya Tono, Akiko Takeda, and Jun-ya Gotoh. "Efficient DC algorithm for constrained sparse optimization." In: *arXiv preprint arXiv:1701.08498* (2017).

[Vek00] Olga Veksler. "Efficient graph-based energy minimization methods in computer vision." PhD thesis. Cornell University, 2000.

[Wan+19] Chunyan Wang, Qiaolin Ye, Peng Luo, Ning Ye, and Liyong Fu. "Robust capped L1-norm twin support vector machine." In: *Neural Networks* 114 (2019), pp. 47–59.

[Wil09] David P Wilson. "Mathematics is applied by everyone except applied mathematicians." In: *Applied mathematics letters* 22.5 (2009), pp. 636–637.

[Wu+20] Yu-Le Wu, Aline Tschanz, Leonard Krupnik, and Jonas Ries. "Quantitative Data Analysis in Single-Molecule Localization Microscopy." In: *Trends in Cell Biology* (2020).

[Xu+17] Yong Xu, Zhengming Li, Jian Yang, and David Zhang. "A survey of dictionary learning algorithms for face recognition." In: *IEEE access* 5 (2017), pp. 8502–8514.

[YG16] Ganzhao Yuan and Bernard Ghanem. "Sparsity Constrained Minimization via Mathematical Programming with Equilibrium Constraints." In: *arXiv:1608.04430* (Aug. 2016).

[Zal02] Constantin Zalinescu. *Convex analysis in general vector spaces*. World scientific, 2002.

[Zha+10] Cun-Hui Zhang et al. "Nearly unbiased variable selection under minimax concave penalty." In: *The Annals of statistics* 38.2 (2010), pp. 894–942.

[ZK10] Yingsong Zhang and Nick Kingsbury. "Restoration of images and 3D data to higher resolution by deconvolution with sparsity regularization." In: *2010 IEEE International Conference on Image Processing*. IEEE. 2010, pp. 1685–1688.

[Zha+15] Zheng Zhang, Yong Xu, Jian Yang, Xuelong Li, and David Zhang. "A survey of sparse representation: algorithms and applications." In: *IEEE access* 3 (2015), pp. 490–530.

[Zou06] Hui Zou. "The adaptive lasso and its oracle properties." In: *Journal of the American statistical association* 101.476 (2006), pp. 1418–1429.